



## 目次

---

- 1. 改訂情報
- 2. はじめに
  - 2.1. 本書の内容
  - 2.2. 対象読者
  - 2.3. その他
- 3. 概要
  - 3.1. GadgetBuilderとは?
  - 3.2. メリット
- 4. 基本
  - 4.1. GadgetBuilder定義
    - 4.1.1. 概要
    - 4.1.2. GadgetBuilder定義
    - 4.1.3. Sample
  - 4.2. 設定の流れ（申請書ガジェット）
    - 4.2.1. テーブル定義
    - 4.2.2. Service、Entity生成
    - 4.2.3. GadgetBuilder定義作成
    - 4.2.4. SQL作成
    - 4.2.5. 動作確認
- 5. リファレンス
  - 5.1. フィールドタイプ
    - 5.1.1. intra-mart Accel Kaiden!標準のフィールドタイプ
  - 5.2. バリデート処理
    - 5.2.1. intra-mart Accel Kaiden!標準のバリデート処理
  - 5.3. 補完処理
    - 5.3.1. intra-mart Accel Kaiden!標準の補完処理
  - 5.4. 通信処理
    - 5.4.1. intra-mart Accel Kaiden!標準の通信処理
  - 5.5. イベント処理
    - 5.5.1. イベントキー
    - 5.5.2. intra-mart Accel Kaiden!標準のイベント処理
  - 5.6. インクルードJSP
    - 5.6.1. intra-mart Accel Kaiden!標準のインクルードJSP
- 6. カスタマイズ手順
  - 6.1. 手順①：GadgetBuilder定義コピー
  - 6.2. 手順②：ガジェットクラス、ガジェットバリエーション編集
  - 6.3. 手順③：ガジェットインポート
  - 6.4. 手順④：ガジェット設定
  - 6.5. 手順⑤：GadgetBuilder定義編集
  - 6.6. 手順⑥-1：GadgetBuilder定義反映(申請書ガジェットマスタ更新)
  - 6.7. 手順⑥-2：GadgetBuilder定義反映(申請書ガジェットマスタ更新不要)

## 改訂情報

変更年月日	変更内容
2017-10-01	初版
2018-04-01	第2版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ ドキュメントの構成や記載内容を見直しました。</li></ul>
2019-04-01	第3版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ ドキュメントの構成や記載内容を見直しました。</li></ul>
2019-08-01	第4版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ ドキュメントの構成や記載内容を見直しました。</li></ul>
2019-12-01	第5版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ diconファイルに記述されている DTDファイルの参照先を http から https に変更しました。<ul style="list-style-type: none"><li>■ 「<a href="#">設定の流れ (申請書ガジェット)</a>」 の kaiden_product_workflow.dicon</li></ul></li></ul>
2020-04-01	第6版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ ドキュメントの記載内容を見直しました。</li></ul>
2020-08-01	第7版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ ドキュメントの記載内容を見直しました。</li></ul>
2020-12-01	第8版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ ドキュメントの記載内容を見直しました。</li></ul>
2022-03-31	第9版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ ドキュメントの記載内容を見直しました。</li></ul>
2022-09-30	第10版 以下の追加・変更を行いました。 <ul style="list-style-type: none"><li>■ ドキュメントの記載内容を見直しました。</li></ul>

## はじめに

---

### 本書の内容

---

本書では、次の様な構成でGadgetBuilderを使用したガジェット関連機能の構築方法や拡張方法を記載しています。

- [概要](#)  
GadgetBuilderの概要やコンセプトを記載しています。
- [基本](#)  
GadgetBuilderを使用した設定方法を記載しています。
- [リファレンス](#)  
本書のリファレンスを記載しています。



#### コラム

本書では、特に記載のない限り次の製品（バージョン）を前提に記載しています。

- intra-mart Accel Platform 2017 Spring (Portland)
- intra-mart Accel Kaiden! Core Module 2.0.6

### 対象読者

---

本書では次の読者を対象としています。

- intra-mart Accel Kaiden! の開発者

### その他

---

- 本書内で記載されている外部URLは、2022年6月現在のものとなります。

## 概要

本項では、GadgetBuilderの概要を説明します。

- [GadgetBuilderとは？](#)
- [メリット](#)

## GadgetBuilderとは？

GadgetBuilderとは、intra-mart Accel Kaiden!のガジェット機能を「設定ファイル」で実現するための機能です。  
(現時点では、申請書向けのガジェットのみを対象としています。)

「設定ファイル」以外にも各所に拡張ポイントを設け、カスタマイズ性も考慮しています。  
次項以降では、基本的な設定方法やカスタマイズポイント等を解説します。



### コラム

ガジェットについては『[intra-mart Accel Kaiden! プログラミングガイド](#)』の「基本」→「仕様解説」→「Gadget」を参照してください。

## メリット

GadgetBuilderを利用することで次の様なメリットが生まれます。

- **構築工数の低下**  
GadgetBuilderでは、「設定ファイル」でガジェットを実現するため、ガジェットを個別に作成する場合に比べ、大幅に工数が削減できます。  
また、カスタマイズを行った場合でもカスタマイズ範囲を局所化可能なため、カスタマイズ工数の削減が見込めます。
- **要件確定後の仕様変更抑止**  
従来ExcelやHTMLで作成していたモックアップと同様に簡単に画面構築が可能のため、要件定義時に実際に動作する画面を確認することができ、後続フェーズでの仕様変更を抑止することができます。
- **仕様変更への柔軟な対応**  
構築後に仕様変更があった場合でも、設定により対応が行えるため、再構築時による手戻りを低減できます。

## 基本

本項では、GadgetBuilderの基本的な設定方法を説明します。

## GadgetBuilder定義

本項では、GadgetBuilderの設定ファイル（以降、GadgetBuilder定義）の作成方法を説明します。

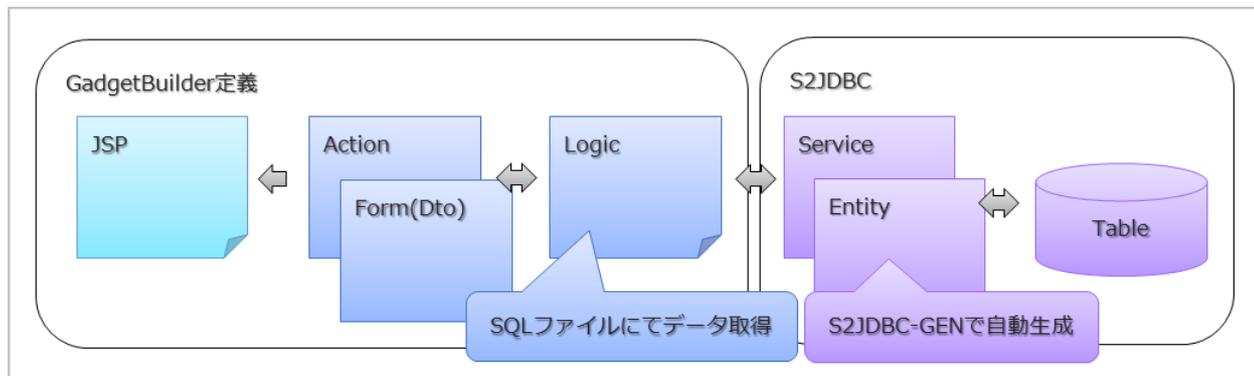
- 概要
  - GadgetBuilder定義とは？
  - GadgetBuilder定義の配置先
- GadgetBuilder定義
  - ガジェット概要設定
  - ガジェットブロック設定
    - フィールドグループ設定
    - フィールド設定
    - 補完設定
    - 通信設定
    - イベント設定
    - インクルード設定
  - ガジェットサーフィス設定
  - ガジェットタグ設定
- Sample

### 概要

#### GadgetBuilder定義とは？

GadgetBuilder定義は、XML形式で設定を行います。  
本項では、経費明細を例に解説を行います。

GadgetBuilder定義は、JSP、Action、Form(Dto)、Logicの設定を行います。



#### **i** コラム

GadgetBuilder定義は、ガジェットマスタの設定階層に準拠しています。  
ガジェットマスタについては『intra-mart Accel Kaiden! プログラミングガイド』の「応用」→「ガジェット（経費旅費）」→「機能説明」→「申請書ガジェットマスタデータについて」を参照してください。  
GadgetBuilder定義の全体イメージは「[Sample](#)」を参照してください。

#### **i** コラム

GadgetBuilder定義の標準設定で対応できない場合は、独自のJSPやLogicを作成し、設定することで拡張することができます。  
GadgetBuilder定義の標準設定で対応できる場合は、拡張設定を行う必要はありません。

## i コラム

申請書ガジェットとIM-Workflowコンテンツの関係性は、下記の図のようになっています。

■ IM-Workflowコンテンツ

■ 申請書ガジェット

**経費旅費情報**

支出目的 \* 支出目的 **経費旅費情報ガジェット**

備考 備考

---

**精算情報**

精算手段 \* 銀行振込

精算内訳 **精算情報ガジェット**

費用合計額	会社負担額	精算予定額	精算額	再集計
0	0	0	0	0

再集計

---

**経費明細**

発生日 *	経費内容 *	組織	プロジェクト
経費区分 *	支払先	金額 *	
発生日 <input type="text" value="2021"/>	経費内容 <input type="text" value="経費内容"/>		
1	経費区分 <input type="text" value="経費区分"/>	<input checked="" type="checkbox"/> サンプル課 1 1	<input checked="" type="checkbox"/> 情報システム株式会...
立替	支払先 <input type="text" value="支払先"/>		金額 <input type="text" value="金額"/>

**経費明細ガジェット**

申請 一時保存

## GadgetBuilder定義の配置先

GadgetBuilder定義は次のパスに配置します。

「%ストレージパス%/kaiden/generic/gadget/%ガジェットクラス%\_%ガジェットバージョン%.xml」

ストレージはシステムストレージ、パブリックストレージのどちらでも配置可能で、パブリックストレージが優先的に読み込まれます。

intra-mart Accel Kaiden!のGadgetBuilder定義は、システムストレージに配置されます。

上記の読み込み順を使用して、独自のGadgetBuilder定義を優先的に読み込ませることが可能です。

(テナント別にGadgetBuilder定義を作成したい場合も、パブリックストレージを使用してください。)

## GadgetBuilder定義

設定

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">
  <gadgetClass>expenseDetail</gadgetClass>
  <gadgetVariation>v01</gadgetVariation>
  <jspPath>/WEB-INF/view/kaiden2/extension_imw/generic/gadget/generic_v01.jsp</jspPath>
  <javaPath>jp.co.slcs.kaiden2.extension_imw.feature.logic.kaiden.gadget.builder.GadgetBuilderLogic</javaPath>
  <sortKey>51001</sortKey>
  <surfaceKey>chapterTitle</surfaceKey>
  <descriptions>

  <!-- 略 -->

</descriptions>
<blocks>

  <!-- 略 -->

</blocks>
<surfaces>

  <!-- 略 -->

</surfaces>
<tags>

  <!-- 略 -->

</tags>
</gadgetBuilderSettings>

```

### コラム

GadgetBuilder定義は、申請書ガジェットマスタへ登録されます。

申請書ガジェットマスタ	設定値
ガジェットクラス	GadgetBuilder定義のガジェットクラス
ガジェットバリエーション	GadgetBuilder定義のガジェットバリエーション
JSPパス	GadgetBuilder定義のJSPパス
Javaパス	GadgetBuilder定義のJavaパス
ソートキー	GadgetBuilder定義のソートキー
サーフィスキー	GadgetBuilder定義のサーフィスキー

### 解説

- **ガジェットクラス (gadgetBuilderSettings / gadgetClass) : 必須**  
ガジェットクラスを設定します。
- **ガジェットバリエーション (gadgetBuilderSettings / gadgetVariation) : 必須**  
ガジェットバリエーションを設定します。
- **JSPパス (gadgetBuilderSettings / jspPath) : 必須**  
ガジェットのJSPパスを定義します。

### コラム

標準のJSPパスを使用する場合、「/WEB-INF/view/kaiden2/extension\_imw/generic/gadget/generic\_v01.jsp」を設定します。  
検索画面のガジェットを作成する場合、「/WEB-INF/view/kaiden2/extension\_imw/generic/gadget/search\_v01.jsp」を設定します。  
標準のJSPパスから変更する場合、独自に作成したJSPパスを設定してください。

- **Javaパス (gadgetBuilderSettings / javaPath) : 必須**  
ガジェットのJavaクラスパス (Logicクラスのクラスパス) を設定します。

### コラム

標準のLogicクラスを使用する場合、  
「jp.co.slcs.kaiden2.extension\_imw.feature.logic.kaiden.gadget.builder.GadgetBuilderLogic」を設定します。  
標準のLogicクラスから変更する場合、独自に作成したLogicクラスのパスを設定してください。

**i** コラム

Logicクラスを独自に作成する場合、  
「jp.co.slcs.kaiden2.extension\_imw.feature.logic.kaiden.gadget.builder.AbstractGadgetBuilderLogic」を継承してください。

- **ソートキー (gadgetBuilderSettings / sortKey) : 必須**  
ガジェットの表示順等で使用するソートキーを設定します。
- **サーフィスキー (gadgetBuilderSettings / surfaceKey) : 必須**  
ガジェットのサーフィスキーを設定します。
- **ガジェットタイトル表示有無 (gadgetBuilderSettings / displayTitle) : 任意**  
JSP上でガジェットのタイトルを表示するか否かを設定します。  
省略時は「true:表示する」が設定されます。
- **ガジェット概要設定 (gadgetBuilderSettings / descriptions) : 必須**  
ガジェット概要を設定します。  
後述の「[ガジェット概要設定](#)」を参照してください。
- **ガジェットブロック設定 (gadgetBuilderSettings / blocks) : 任意**  
ガジェットブロックを設定します。  
後述の「[ガジェットブロック設定](#)」を参照してください。
- **ガジェットサーフィス設定 (gadgetBuilderSettings / surfaces) : 任意**  
ガジェットサーフィスを設定します。  
後述の「[ガジェットサーフィス設定](#)」を参照してください。
- **ガジェットタグ設定 (gadgetBuilderSettings / tags) : 任意**  
ガジェットタグを設定します。  
後述の「[ガジェットタグ設定](#)」を参照してください。
- **ガジェット引数設定 (gadgetBuilderSettings / args) : 任意**  
ガジェット引数を設定します。
- **拡張設定 (gadgetBuilderSettings / extension) : 任意**  
標準機能を拡張 (カスタマイズ) する場合の拡張先の設定を行います。
- **拡張設定 (Logic) (gadgetBuilderSettings / extension / logic) : 任意**  
標準のLogicクラスを変更する場合に設定します。

**i** コラム

標準のLogicクラス「jp.co.slcs.kaiden2.base.feature.logic.kaiden.generic.impl.GadgetLogic」を変更する場合に設定します。

GadgetBuilder定義で拡張できるLogicクラスの役割は次の通りです。

<b>Javaパス</b>	ワークフローイベントの申請・承認・再申請・一時保存などを行います。
<b>拡張設定 (Logic)</b>	構成情報の返却やデータの登録・更新・削除・検索などを行います。 「拡張設定 (Logic)」で設定したクラスは、「Javaパス」に設定したクラスから使用されません。

- **拡張設定 (URL) (gadgetBuilderSettings / extension / url) : 任意**  
標準のURLを変更する場合に設定します。

**i** コラム

標準のURL「kaiden/gadget/gadgetBuilder」を変更する場合に設定します。  
標準のURLを変更することで、Actionクラスの拡張が行えます。

## ガジェット概要設定

ガジェット概要設定では、申請書マスタメンテナンス機能などで表示される、各ガジェットの概要を設定します。

ガジェット概要設定は、「gadgetBuilderSettings / descriptions / description」階層に設定を行います。

「gadgetBuilderSettings / descriptions / description」階層は、1件以上の複数設定 (ロケール別) が可能です。

設定

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

  <!-- 略 -->

  <descriptions>
    <description localeId="en" gadgetName="Expenses Particulars" description="Enter Expenses Particulars.&#xD;Only Japanese Yen is available."/>
    <description localeId="ja" gadgetName="経費明細" description="経費明細の入力を行います。&#xD;日本円のみ利用可能です。"/>
    <description localeId="zh_CN" gadgetName="明細" description="入明細。&#xD;只可使用日元。"/>
  </descriptions>

  <!-- 略 -->

</gadgetBuilderSettings>
```

## コラム

ガジェット概要設定は、申請書ガジェット概要マスタへ登録されます。

### 申請書ガジェット概要マスタ 設定値

ロケールID	ガジェット概要設定のロケールID
ガジェット名	ガジェット概要設定のガジェット名
ガジェット概要	ガジェット概要設定のガジェット概要

## 解説

- **ロケールID (gadgetBuilderSettings / descriptions / description @ localeId)** : 必須  
ロケールIDを設定します。
- **ガジェット名 (gadgetBuilderSettings / descriptions / description @ gadgetName)** : 必須  
ガジェット名を設定します。
- **ガジェット概要 (gadgetBuilderSettings / descriptions / description @ description)** : 必須  
ガジェット概要を設定します。

## ガジェットブロック設定

ガジェットブロック設定では、ガジェットブロックの定義を設定します。

ガジェットブロック設定は、「gadgetBuilderSettings / blocks / block」階層に設定を行います。  
「gadgetBuilderSettings / blocks / block」階層は、複数設定が可能です。

## 設定

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">
  <!-- 部 -->
  <blocks>
    <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">
      <service basic="k30tExpenseDetailService" archive="ka30tExpenseDetailService">
        <sqls>
          <sql id="init" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="copy" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="merge" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="archive" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="select" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
        </sqls>
      </service>
      <groups>
        <!-- 部 -->
      </groups>
      <table rowNumberTarget="rowNumber" type="HORIZONTAL">
        <colgroups>
          <colgroup style="width:50px; overflow:hidden"/>
          <colgroup style="overflow:hidden"/>
          <colgroup style="width:23%; overflow:hidden"/>
          <colgroup style="width:23%; overflow:hidden"/>
          <colgroup style="width:23%; overflow:hidden"/>
        </colgroups>
        <rows>
          <row>
            <cols>
              <col groupId="rowNumber" rowspan="3"/>
              <col groupId="occurDate"/>
              <col groupId="summary" colspan="3"/>
            </cols>
          </row>
          <row>
            <cols>
              <col groupId="expType" colspan="2"/>
              <col groupId="department"/>
              <col groupId="project"/>
            </cols>
          </row>
          <row>
            <cols>
              <col groupId="payType"/>
              <col groupId="payee" colspan="2"/>
              <col groupId="transAmount"/>
            </cols>
          </row>
        </rows>
      </table>
    </block>
  </blocks>
  <!-- 部 -->
  </gadgetBuilderSettings>

```

■ 表示状態

## コラム

ガジェットブロック設定は、申請書ガジェットブロックマスタへ登録されます。

### 申請書ガジェットブロックマスタ 設定値

ブロックID	ガジェットブロック設定のブロックID
ブロックタイプ	ガジェットブロック設定のブロックタイプ
ソートキー	ガジェットブロック設定のソートキー
サーフィスキー	ガジェットブロック設定のサーフィスキー

## 解説

- **ブロックID (gadgetBuilderSettings / blocks / block @ id)** : 必須  
ブロックIDを設定します。
- **ブロックタイプ (gadgetBuilderSettings / blocks / block @ type)** : 必須  
ガジェットブロックの種類を設定します。

**i** コラム

「NORMAL」「DETAIL」のいずれかを設定します。

- 明細行を保持しない場合は「NORMAL」を設定します。
- 明細行を保持する場合は「DETAIL」を設定します。

- **モード (gadgetBuilderSettings / blocks / block @ mode) : 任意**  
ガジェットブロックのモードを設定します。  
ブロックタイプが「DETAIL」の場合、「card」を設定可能です。  
「card」を設定すると、明細行の表示形式がカード方式になります。  
標準ガジェットはモードを設定した状態で検証していません。  
モードを設定する場合、十分な検証が必要です。
- **ソートキー (gadgetBuilderSettings / blocks / block @ sortKey) : 任意**  
ソートキーを設定します。  
未設定の場合はガジェットブロック設定順に連番となります。
- **サーフィスキー (gadgetBuilderSettings / blocks / block @ surfaceKey) : 必須**  
ガジェットブロックのサーフィスキーを設定します。
- **サービス設定 (gadgetBuilderSettings / blocks / block / service) : 任意**  
サービスやSQLパスなどを設定します。
- **基本サービスクラス (gadgetBuilderSettings / blocks / block / service @ basic) : 任意**  
基本テーブルのサービスクラス（コンポーネント名）を設定します。

**i** コラム

基本テーブルとは、申請時のデータを保存するトランザクションテーブルです。

- **アーカイブサービスクラス (gadgetBuilderSettings / blocks / block / service @ archive) : 任意**  
アーカイブテーブルのサービスクラス（コンポーネント名）を設定します。

**i** コラム

アーカイブテーブルとは、案件退避時に基本テーブルのデータを保存するトランザクションテーブルです。

- **SQL設定 (gadgetBuilderSettings / blocks / block / service / sqls / sql) : 任意**  
検索などで使用するSQLや検索用クラスを設定します。
- **SQLID (gadgetBuilderSettings / blocks / block / service / sqls / sql @ id) : 必須**  
SQL設定のIDを設定します。

**i** コラム

SQL設定を使用した検索を行う場合、検索条件にSQLIDも含まれます。  
SQLIDと主な用途は次の通りです。

ID	主な用途
INIT	初期表示時のデータ検索（申請、処理、一時保存、各詳細画面などで使用します）
INIT_ARCHIVE	アーカイブデータ初期表示時のデータ検索（過去案件詳細画面で使用します）
COPY	コピー時のコピー元データ検索（申請書コピーで使用します）
MERGE	マージ元データ検索（承認時や再申請時にリクエストデータで不足しているデータを基本テーブルから補完する用途で使用します）
ARCHIVE	アーカイブ時のアーカイブ元データ検索（案件退避処理で使用します）
SELECT	対象検索（事前申請書検索による申請書紐付け時、検索画面ガジェットの初期検索などで使用します）
SEARCH	一覧検索（検索画面ガジェットで使用します）

- **SQLパス (gadgetBuilderSettings / blocks / block / service / sqls / sql @ path) : 任意**  
SQLパスを設定します。
- **代替クラス (gadgetBuilderSettings / blocks / block / service / sqls / sql @ component) : 任意**  
検索を代替するクラス（コンポーネント名）を設定します。  
API等を使用して検索を行う場合に使用します。

**i** コラム

設定クラスは次のクラスを継承したクラスである必要があります。

- `jp.co.slcs.kaiden2.base.foundation.gadget.builder.select.AbstractGadgetSelect`

**i** コラム

検索を行わない場合（初期表示しない、コピーしないなど）、次の様に設定することで検索を抑止できます。  
（「notSelect」クラス（コンポーネント名）は検索を行わないクラスです。）

```
<sql id="copy" component="notSelect"/>
```

- **フィールドグループ設定（`gadgetBuilderSettings / blocks / block / groups`）**：任意  
フィールドグループを設定します。  
後述の「[フィールドグループ設定](#)」を参照してください。
- **構成設定（`gadgetBuilderSettings / blocks / block / table`）**：任意  
PC用画面のガジェット構成を設定します。  
未設定の場合、ガジェット構成時にフィールドグループ設定順で構成されます。

**i** コラム

スマートフォン用画面のガジェット構成は、フィールドグループ設定順で構成されます。  
ブロック引数設定（`gadgetBuilderSettings / blocks / block / args`）に「キー：tableFormatSp、値：true」を指定することで、tableタグの設定順に従って表示することが可能です。

- **構成タイプ（`gadgetBuilderSettings / blocks / block / table @ type`）**：必須  
ガジェット構成のタイプを設定します。  
「HORIZONTAL」「VERTICAL」のいずれかを設定します。

**i** コラム

「HORIZONTAL」（横配置）を設定した場合、後述の「行 - 列設定」を使用します。  
画面構成は次のようになります。

<input type="checkbox"/>	発生日*	経費内容*	組織	プロジェクト	連携
	経費区分*	支払分類*	支払先	金額*	
1	<input type="text"/>				
	立替	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

**i** コラム

「VERTICAL」（縦配置）を設定した場合、後述の「列 - 行設定」を使用します。  
画面構成は次のようになります。

出張期間*	<input type="text"/>	<input type="text"/>
出張目的*	<input type="text"/>	<input type="text"/>
出張先*	<input type="text"/>	<input type="text"/>
出張先住所	<input type="text"/>	<input type="text"/>
備考	<input type="text"/>	<input type="text"/>

- **行番号フィールド（`gadgetBuilderSettings / blocks / block / table @ rowNumberTarget`）**：任意  
ブロックタイプ「DETAIL」を設定した場合の行番号フィールドを設定します。  
TmTableタグの「rowNumberTarget」属性に設定されます。

**i** コラム

TmTableについては『[intra-mart Accel Kaiden! プログラミングガイド](#)』の「基本」→「仕様解説」→「TmTable」を参照してください。

- **カラムグループ設定（`gadgetBuilderSettings / blocks / block / table / colgroups / colgroup`）**：任意  
ガジェット構成時のカラムグループを設定します。
- **カラムグループスタイル（`gadgetBuilderSettings / blocks / block / table / colgroups / colgroup @ style`）**：任意  
ガジェット構成時のカラムグループのスタイルを設定します。
- **行 - 列設定（`gadgetBuilderSettings / blocks / block / table / rows / row / cols / col`）**：任意

ガジェット構成時の行 - 列を設定します。

### コラム

構成タイプで「HORIZONTAL」を設定した場合に使用します。

- フィールドグループID (`gadgetBuilderSettings / blocks / block / table / rows / row / cols / col @ groupId`) : 任意  
当該構成を行うフィールドグループのIDを設定します。
- 列結合数 (`gadgetBuilderSettings / blocks / block / table / rows / row / cols / col @ colspan`) : 任意  
当該構成の列結合数を設定します。
- 行結合数 (`gadgetBuilderSettings / blocks / block / table / rows / row / cols / col @ rowspan`) : 任意  
当該構成の行結合数を設定します。
- 列 - 行設定 (`gadgetBuilderSettings / blocks / block / table / cols / col / rows / row`) : 任意  
ガジェット構成時の列 - 行を設定します。

### コラム

構成タイプで「VERTICAL」を設定した場合に使用します。

- フィールドグループID (`gadgetBuilderSettings / blocks / block / table / cols / col / rows / row @ groupId`) : 任意  
当該構成を行うフィールドグループのIDを設定します。
- 列結合数 (`gadgetBuilderSettings / blocks / block / table / cols / col / rows / row @ colspan`) : 任意  
当該構成の列結合数を設定します。
- 行結合数 (`gadgetBuilderSettings / blocks / block / table / cols / col / rows / row @ rowspan`) : 任意  
当該構成の行結合数を設定します。
- 補完設定 (`gadgetBuilderSettings / blocks / block / completions`) : 任意  
補完処理を設定します。  
後述の「[補完設定](#)」を参照してください。
- 通信設定 (`gadgetBuilderSettings / blocks / block / claims`) : 任意  
ガジェット間通信を設定します。  
後述の「[通信設定](#)」を参照してください。
- イベント設定 (`gadgetBuilderSettings / blocks / block / events`) : 任意  
ガジェットの各イベントを設定します。  
後述の「[イベント設定](#)」を参照してください。
- インクルード設定 (`gadgetBuilderSettings / blocks / block / includes`) : 任意  
JSPへのインクルード先を設定します。  
後述の「[インクルード設定](#)」を参照してください。
- ブロック引数設定 (`gadgetBuilderSettings / blocks / block / args`) : 任意  
ブロック引数を設定します。  
主に、「class」属性や「style」属性の値となります。

### コラム

ブロック引数設定に、申請データの初期表示や申請書コピーの際にSQLで取得したデータの内、設定不要なデータを除外することができます。  
「key」に「loadDataExclude」及び「loadData4CopyExclude」を設定し、「value」に除外する項目（予備項目等）をカンマ区切りで設定してください。

#### フィールドグループ設定

フィールドグループ設定では、単一または複数のフィールドをまとめたグループの定義を設定します。

フィールドグループ設定は、「`gadgetBuilderSettings / blocks / block / groups / group`」階層に設定を行います。  
「`gadgetBuilderSettings / blocks / block / groups / group`」階層は、複数設定が可能です。

設定

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

  <!-- 略 -->

  <blocks>
    <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">

      <!-- 略 -->

      <groups>
        <group id="rowLumber" surfaceKey="rowLumber" requiredType="ALWAYS_ANY" inputType="LABEL">
          <args>
            <arg key="style_pc" value="text-align:center;"/>
          </args>
          <fields>

            <!-- 略 -->

          </fields>
        </group>

        <!-- 略 -->

        <group id="expType" surfaceKey="expType" requiredType="ALWAYS_REQ" inputType="CONFIG">
          <fields>

            <!-- 略 -->

          </fields>
        </group>

        <!-- 略 -->

      </groups>
    </block>

    <!-- 略 -->

  </blocks>

  <!-- 略 -->
        
```

■表示状態

## コラム

フィールドグループ設定は、申請書ガジェットフィールドグループマスタへ登録されます。

### 申請書ガジェットフィールドグループマスタ 設定値

ブロックID	ガジェットブロックマスタのブロックID
フィールドグループID	フィールドグループ設定のフィールドグループID
サーフィスキー	フィールドグループ設定のサーフィスキー
必須タイプ	フィールドグループ設定の必須タイプ 以下、必須タイプによる設定値 「0:SETTING_ANY」 「1:SETTING_REQ」 「a:ALWAYS_ANY」 「b:ALWAYS_REQ」
入力タイプ	フィールドグループ設定の入力タイプ 以下、入力タイプによる設定値 「0:CONFIG」 「a:LABEL」 「b:TEXT」 「c:TEXT_LABEL」
ソートキー	フィールドグループ設定のソートキー

## 解説

- **フィールドグループID (gadgetBuilderSettings / blocks / block / groups / group @ id)** : 必須  
フィールドグループIDを設定します。
- **サーフィスキー (gadgetBuilderSettings / blocks / block / groups / group @ surfaceKey)** : 必須  
フィールドグループのサーフィスキーを設定します。
- **必須タイプ (gadgetBuilderSettings / blocks / block / groups / group @ requiredType)** : 任意  
フィールドグループの必須タイプを設定します。  
未設定の場合は「SETTING\_ANY」(変更可能(デフォルト任意))となります。  
選択可能な必須タイプは次の通りです。

設定値	説明
SETTING_ANY	変更可能(デフォルト任意)
SETTING_REQ	変更可能(デフォルト必須)
ALWAYS_ANY	常時任意

**i** コラム

任意の場合、フィールドは次のようになります。

- PC

サンプル

- SP(スマートフォン)

サンプル

**i** コラム

必須の場合、フィールドは次のようになります。

- PC

サンプル\*

- SP(スマートフォン)

\* サンプル

- 入カタイプ (**gadgetBuilderSettings / blocks / block / groups / group @ inputType**) : 任意  
フィールドグループの入カタイプを設定します。  
未設定の場合は「CONFIG」(設定変更可)となります。  
選択可能な入カタイプは次の通りです。

設定値	説明
CONFIG	設定変更可
LABEL	ラベル(変更不可)
TEXT	テキスト(変更不可)
TEXT_LABEL	テキスト(処理時入力不可・変更不可)

**i** コラム

ラベルの場合、フィールドは次の様に入力が不可能になります。

- PC

サンプル

- SP(スマートフォン)

サンプル

**i** コラム

テキストの場合、フィールドは次の様に入力が可能になります。

- PC

サンプル

- SP(スマートフォン)

サンプル

- ソートキー (**gadgetBuilderSettings / blocks / block / groups / group @ sortKey**) : 任意

ソートキーを設定します。

未設定の場合はフィールドグループ設定順に連番となります。

- 必須マーク (**gadgetBuilderSettings / blocks / block / groups / group @ requiredMark**) : 任意  
フィールドグループの必須マーク (true / false) を設定します。

## コラム

申請書用のガジェットの場合は、申請書マスタメンテナンスでの設定で上書きされます。

- フィールド設定 (**gadgetBuilderSettings / blocks / block / groups / group / fields**) : 任意  
内包するフィールドを設定します。  
後述の「[フィールド設定](#)」を参照してください。
- 引数設定 (**gadgetBuilderSettings / blocks / block / groups / group / args**) : 任意  
フィールドグループへの引数を設定します。  
主に、「class」属性や「style」属性の値となります。

## フィールド設定

フィールド設定では、フィールドの構成やバリデーションなどを設定します。

フィールド設定は、「gadgetBuilderSettings / blocks / block / groups / group / fields / field」階層に設定を行います。  
「gadgetBuilderSettings / blocks / block / groups / group / fields / field」階層は、複数設定が可能です。

## 設定

The image displays the configuration for a date field in GadgetBuilder. On the left, the XML configuration is shown, highlighting the field definition and its arguments and validation rules. On the right, the application UI is shown with a date field and associated error messages.

```

<?xml version="1.0" encoding="UTF-8" standalone="yes">
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">
  <!-- 略 -->
  <blocks>
    <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">
      <!-- 略 -->
      <groups>
        <group id="rowNumber" surfaceKey="rowNumber" requiredType="ALWAYS_ANY" inputType="LABEL">
          <!-- 略 -->
          <fields>
            <field id="rowNumber" surFaceKey="rowNumber" requiredType="INHERIT" inputType="INHERIT" type="PANEL"/>
            <field id="imaApplyBaseDate" surfaceKey="imaApplyBaseDate" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DIO"/>
            <field id="searchCriteriaCompany" surfaceKey="searchCriteriaCompany" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DIO"/>
            <field id="payMethodCd" surfaceKey="payMethod" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
          </fields>
          <group id="occurDate" surfaceKey="occurDate" requiredType="ALWAYS_REQ" inputType="CONFIG">
            <fields>
              <field id="occurDate" surFaceKey="occurDate" requiredType="INHERIT" inputType="INHERIT" type="TEXT_DATE">
                <args>
                  <arg key="style_pc" value="max-width:70px; width:calc(100% - 40px);"/>
                  <arg key="title" value="occurDate"/>
                  <arg key="placeholder" value="occurDate"/>
                  <arg key="autoFormat" value="true"/>
                </args>
                <validates>
                  <validate event="default" config="{ 'id': 'date' }"/>
                </validates>
              </field>
            </fields>
          </group>
        </group>
      </groups>
    </block>
  </blocks>
  <!-- 略 -->
</gadgetBuilderSettings>
    
```

The UI on the right shows a form with a date field. A red box highlights the field and its error messages:

- クライアントバリデート**: 入力エラーが見つかりました。 経費明細 - 1 - 発生日
- サーババリデート**: 申請に失敗しました。 サーバリデートエラーが発生しました。 "19000101" - "29991231"の範囲で入力してください。 経費明細 - 1 - 発生日

**i** コラム

フィールド設定は、申請書ガジェットフィールドマスタへ登録されます。

## 申請書ガジェットフィールドマスタ 設定値

ブロックID	ガジェットブロックマスタのブロックID
フィールドキー	フィールド設定のフィールドID
フィールドグループID	ガジェットフィールドマスタのフィールドグループID
サーフィスキー	フィールド設定のサーフィスキー
必須タイプ	フィールド設定の必須タイプ 以下、必須タイプによる設定値 「0:SETTING_ANY」 「1:SETTING_REQ」 「a:ALWAYS_ANY」 「b:ALWAYS_REQ」 「z:INHERIT」
入力タイプ	フィールド設定の入力タイプ 以下、入力タイプによる設定値 「0:CONFIG」 「a:LABEL」 「b:TEXT」 「c:TEXT_LABEL」 「z:INHERIT」
ソートキー	フィールド設定のソートキー

## 解説

- **フィールドID (gadgetBuilderSettings / blocks / block / groups / group / fields / field @ id)** : 必須  
フィールドIDを設定します。
- **サーフィスキー (gadgetBuilderSettings / blocks / block / groups / group / fields / field @ surfaceKey)** : 必須  
フィールドのサーフィスキーを設定します。
- **必須タイプ (gadgetBuilderSettings / blocks / block / groups / group / fields / field @ requiredType)** : 任意  
フィールドの必須タイプを設定します。  
未設定の場合は「INHERIT」（フィールドグループの設定を引き継ぐ）となります。  
選択可能な必須タイプは次の通りです。

設定値	説明
SETTING_ANY	変更可能(デフォルト任意)
SETTING_REQ	変更可能(デフォルト必須)
ALWAYS_ANY	常時任意
ALWAYS_REQ	常時必須
INHERIT	フィールドグループの設定を引き継ぐ

**i** コラム

任意の場合、フィールドは次のようになります。

- PC

サンプル

- SP(スマートフォン)

サンプル

**i** コラム

必須の場合、フィールドは次のようになります。

- PC

サンプル

- SP(スマートフォン)

\* サンプル

- 入力タイプ (**gadgetBuilderSettings / blocks / block / groups / group / fields / field @ inputType**) : 任意  
フィールドの入力タイプを設定します。  
未設定の場合は「INHERIT」（フィールドグループの設定を引き継ぐ）となります。  
選択可能な入力タイプは次の通りです。

設定値	説明
CONFIG	設定変更可
LABEL	ラベル(変更不可)
TEXT	テキスト(変更不可)
TEXT_LABEL	テキスト(処理時入力不可・変更不可)
INHERIT	フィールドグループの設定を引き継ぐ

**i** コラム

ラベルの場合、フィールドは次の様に入力が不可能になります。

- PC

サンプル

- SP(スマートフォン)

サンプル

**i** コラム

テキストの場合、フィールドは次の様に入力が可能になります。

- PC

サンプル

- SP(スマートフォン)

サンプル

- ソートキー (**gadgetBuilderSettings / blocks / block / groups / group / fields / field @ sortKey**) : 任意  
ソートキーを設定します。  
未設定の場合はフィールド設定順に連番となります。
- フィールドタイプ (**gadgetBuilderSettings / blocks / block / groups / group / fields / field @ type**) : 任意  
フィールドタイプを設定します。  
設定可能なフィールドタイプは「[フィールドタイプ](#)」を参照してください。
- 引数設定 (**gadgetBuilderSettings / blocks / block / groups / group / fields / field / args / arg**) : 任意  
フィールドへの引数を設定します。  
設定可能な引数（属性）はフィールドタイプに依存します。  
詳細は「[フィールドタイプ](#)」を参照してください。
- バリデート設定 (**gadgetBuilderSettings / blocks / block / groups / group / fields / field / validates / validate**) : 任意  
フィールドへのバリデートを設定します。
- バリデート実行イベント (**gadgetBuilderSettings / blocks / block / groups / group / fields / field / validates / validate @ event**) : 必須

フィールドバリデートの実行イベントを設定します。

主なイベントは次の通りです。

設定値	説明
default	全てのイベントで実行
apply	申請イベント(PC版)のみ実行
applysp	申請イベント(SP(スマートフォン)版)のみ実行
tempsave	一時保存イベント(PC版)のみ実行
tempsavesp	一時保存イベント(SP(スマートフォン)版)のみ実行
reapply	再申請イベント(PC版)のみ実行
reapplysp	再申請イベント(SP(スマートフォン)版)のみ実行
process	処理イベント(PC版)のみ実行
processsp	処理イベント(SP(スマートフォン)版)のみ実行
warning	アラート処理を実行 (情報明細ガジェットにコメント必須)
information	アラート処理を実行 (情報明細ガジェットのコメントは任意)

### コラム

バリデート実行イベントに、「apply」、「tempsave」、「reapply」、「process」のいずれかを設定した場合、各イベントのPC版のクライアントサイドチェック、PC版とSP(スマートフォン)版のサーバーサイドチェックを実施します。SP(スマートフォン)版のクライアントサイドチェックを実施したい場合は「applysp」、「tempsavesp」、「reapplysp」、「processsp」をあわせて設定してください。  
「applysp」、「tempsavesp」、「reapplysp」、「processsp」のバリデート実行対象を「ALWAYS」に設定してもサーバーサイドチェックは実施されません。

### コラム

「warning」、「information」を設定したガジェットは、情報明細ガジェットを同じ申請書に組み込んでください。  
「warning」、「information」を設定した場合、チェックがNGの際も申請可能です。  
アラート処理の詳細は『[intra-mart Accel Kaiden! 経費旅費 / 管理者操作ガイド](#)』を参照してください。

- バリデート実行対象 (`gadgetBuilderSettings / blocks / block / groups / group / fields / field / validates / validate @ target`) : 任意  
フィールドバリデートの実行対象を設定します。  
選択可能な実行対象は次の通りです。(省略時は「ALWAYS」)

設定値	説明
CLIENT	クライアントサイドのみ実行
SERVER	サーバーサイドのみ実行
ALWAYS	クライアントサイド、サーバーサイドの両方で実行

- バリデート内容 (`gadgetBuilderSettings / blocks / block / groups / group / fields / field / validates / validate @ config`) : 任意  
フィールドバリデートの内容を設定します。  
詳細は「[バリデート処理](#)」を参照してください。

## 補完設定

補完設定では、各フィールドに対する補完処理を設定します。

補完設定は、「`gadgetBuilderSettings / blocks / block / completions / completion`」階層に設定を行います。

「`gadgetBuilderSettings / blocks / block / completions / completion`」階層は、複数設定が可能です。

## 設定

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

  <!-- 略 -->

  <blocks>
    <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">

      <!-- 略 -->

      <completions>
        <completion name="departmentSetCd" position="1" type="claimValueOnInit" args="{ 'claim':'claimDefaultDepartmentSetCd',
'target':'1' }" target="CLIENT" mode="EMPTY"/>
        <completion name="departmentCd" position="1" type="claimValueOnInit" args="{ 'claim':'claimDefaultDepartmentCd',
'target':'1' }" target="CLIENT" mode="EMPTY"/>
        <completion name="projectCd" position="1" type="claimValueOnInit" args="{ 'claim':'claimDefaultProjectCd', 'target':'1' }"
target="CLIENT" mode="EMPTY"/>
        <completion name="payMethodCd" position="1" type="claimValue" args="{ 'claim':'claimPayMethod' }"
target="ALWAYS" mode="ALWAYS"/>
        <completion name="transCurrencyCd" position="2" type="variable" args="{ 'field':'companyCurrencyCd' }"
target="ALWAYS" mode="ALWAYS"/>
        <completion name="companyRate" position="3" type="fixed" args="{ 'value':'1' }"
target="ALWAYS" mode="EMPTY"/>
        <completion name="companyAmount" position="4" type="variable" args="{ 'field':'transAmount' }"
target="ALWAYS" mode="ALWAYS"/>
        <completion name="quantity" position="5" type="fixed" args="{ 'value':'1' }"
target="ALWAYS" mode="EMPTY"/>
        <completion name="companyAmountSum" position="6" type="calcMul" args="
{ 'field1':'companyAmount', 'field2':'quantity', 'round':'4', 'digits':'0' }" target="ALWAYS" mode="ALWAYS"/>
        <completion name="companyAmountTax" position="7" type="calcTax" args="
{ 'amount':'companyAmountSum', 'taxRate':'taxRate', 'round':'4', 'digits':'0' }" target="ALWAYS" mode="ALWAYS"/>
        <completion name="companyAmountExcTax" position="8" type="calcSub" args="
{ 'field1':'companyAmountSum', 'field2':'companyAmountTax', 'round':'4', 'digits':'0' }" target="ALWAYS" mode="ALWAYS"/>
      </completions>

      <!-- 略 -->

    </block>
  </blocks>

  <!-- 略 -->

</gadgetBuilderSettings>

```

## 解説

- 補完フィールド名 (**gadgetBuilderSettings / blocks / block / completions / completion @ name**) : 必須  
補完結果を設定するフィールド名を設定します。
- 補完順 (**gadgetBuilderSettings / blocks / block / completions / completion @ position**) : 必須  
補完処理の実行順を設定します。  
本設定に従って補完処理の実行順を制御します。
- 補完対象 (**gadgetBuilderSettings / blocks / block / completions / completion @ target**) : 任意  
次の補完処理対象を設定します。(未設定の場合は「ALWAYS」)

設定値	説明
CLIENT	クライアントサイドのみ補完
SERVER	サーバーサイドのみ補完
ALWAYS	クライアントサイド、サーバーサイドの両方で補完

- 補完モード (**gadgetBuilderSettings / blocks / block / completions / completion @ mode**) : 任意  
次の補完処理モードを設定します。(未設定の場合は「ALWAYS」)

設定値	説明
EMPTY	補完フィールドに値がない場合補完
NOT_EMPTY	補完フィールドに値がある場合補完
ALWAYS	常に補完

- 補完タイプ (**gadgetBuilderSettings / blocks / block / completions / completion @ type**) : 必須

補完処理を行うタイプを設定します。

詳細は「[補完処理](#)」を参照してください。

- 引数 (**gadgetBuilderSettings / blocks / block / completions / completion / args**) : 任意

補完処理の引数を設定します。

詳細は「[補完処理](#)」を参照してください。

## **i** コラム

補完処理実行タイミング

- 初期表示時
- 申請時
- 一時保存時
- 処理時
- 再申請時

## **i** コラム

フィールド設定の引数設定に、イベントを設定している場合

フィールドに設定できるイベントはフィールドタイプによって異なるため「[フィールドタイプ](#)」を参照してください。

- フィールドに「click」イベントを設定している場合、そのフィールドのクリック時にも補完処理が実行されます。

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="BUTTON">
  <args>
    <arg key="event" value="click"/>
  </args>
</field>
```

- フィールドに「change」イベントを設定している場合、そのフィールドの変更時にも補完処理が実行されます。

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT_NUMERIC">
  <args>
    <arg key="event" value="change"/>
  </args>
</field>
```

## 通信設定

通信設定では、各ブロックのガジェット間（ブロック間）通信を設定します。

通信設定は、「gadgetBuilderSettings / blocks / block / claims / claim」階層に設定を行います。

「gadgetBuilderSettings / blocks / block / claims / claim」階層は、複数設定が可能です。

## 設定

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

  <!-- 略 -->

  <blocks>
    <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">

      <!-- 略 -->

      <claims>
        <claim name="claimExpenseAmount" type="claimAmountSummary" args="{ 'amount':'companyAmountSum' }"/>
        <claim name="claimTemporaryPayAmount" type="claimAmountSummary" args="
        { 'amount':'companyAmountSum', 'paymentFlag':'paymentFlag', 'payment':'1' }"/>
        <claim name="claimCompanyPayAmount" type="claimAmountSummary" args="
        { 'amount':'companyAmountSum', 'paymentFlag':'paymentFlag', 'payment':'0' }"/>
        <claim name="claimDateDetail" type="claimColumnValuesList" args="{ 'field1':'occurDate', 'field2':'occurDate' }"/>
        <claim name="claimExpenseDetail" type="claimColumnValuesList" args="{ 'field1':'companyAmountSum' }"/>
        <claim name="claimExpenccCount" type="claimRowCount" args=""/>
        <claim name="loadData4Settle" type="claimSearchData" args="{ 'systemMatterId':'imwSystemMatterId',
        'userDataId':'imwUserDataId', 'gadgetClass':'gadgetClass' }" target="CLIENT"/>
      </claims>

      <!-- 略 -->
    </block>
  </blocks>

  <!-- 略 -->
</gadgetBuilderSettings>

```

## 解説

- 通信名 (**gadgetBuilderSettings / blocks / block / claims / claim @ name**) : 必須  
通信処理の名称を設定します。
- 通信対象 (**gadgetBuilderSettings / blocks / block / claims / claim @ target**) : 任意  
次の通信処理対象を設定します。(未設定の場合は「ALWAYS」)

設定値	説明
CLIENT	クライアントサイドのみ通信
SERVER	サーバーサイドのみ通信
ALWAYS	クライアントサイド、サーバーサイドの両方で通信

- 通信タイプ (**gadgetBuilderSettings / blocks / block / claims / claim @ type**) : 必須  
通信処理を行うタイプを設定します。  
詳細は「[通信処理](#)」を参照してください。
- 引数 (**gadgetBuilderSettings / blocks / block / claims / claim / args**) : 任意  
通信処理の引数を設定します。  
詳細は「[通信処理](#)」を参照してください。



## コラム

### 通信処理実行タイミング

- 初期表示時
- 申請時
- 一時保存時
- 処理時
- 再申請時

## イベント設定

イベント設定では、各ブロックのイベント処理を設定します。

イベント設定は、「gadgetBuilderSettings / blocks / block / events / event」階層に設定を行います。  
「gadgetBuilderSettings / blocks / block / events / event」階層は、複数設定が可能です。

## 設定

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

<!-- 略 -->

<blocks>
  <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">

<!-- 略 -->

    <events>
      <event target="matter_end" name="journalSourceDetailEntryForSqlEvent">
        <args>
          <arg key="sqlPath" value="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/journal/ExpenseDetail.sql"/>
        </args>
      </event>
    </events>

<!-- 略 -->

  </block>
</blocks>

<!-- 略 -->

</gadgetBuilderSettings>

```

## 解説

- 対象イベント (**gadgetBuilderSettings / blocks / block / events / event @ target**) : 必須  
対象のイベントキーを設定します。  
詳細は「[イベント処理](#)」を参照してください。
- イベント名 (**gadgetBuilderSettings / blocks / block / events / event @ name**) : 必須  
イベント名 (コンポーネント名) を設定します。  
詳細は「[イベント処理](#)」を参照してください。
- イベント引数 (**gadgetBuilderSettings / blocks / block / events / event / args**) : 任意  
イベント処理の引数を設定します。  
詳細は「[イベント処理](#)」を参照してください。

## インクルード設定

インクルード設定では、各ブロックのクライアントサイドでインクルードを行うJSPを設定します。

インクルード設定は、「gadgetBuilderSettings / blocks / block / includes / include」階層に設定を行います。  
「gadgetBuilderSettings / blocks / block / includes / include」階層は、複数設定が可能です。

## 設定

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

<!-- 略 -->

<blocks>
  <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">

<!-- 略 -->

    <includes>
      <include path="/WEB-INF/view/kaiden2/product_workflow/generic/gadget/include/defaultDepartmentProject.jsp" />
    </includes>

<!-- 略 -->

  </block>
</blocks>

<!-- 略 -->

</gadgetBuilderSettings>

```

## 解説

- インクルードパス (**gadgetBuilderSettings / blocks / block / includes / include @ path**) : 必須  
インクルードを行うJSPのパスを設定します。  
詳細は「[インクルードJSP](#)」を参照してください。
- インクルード引数 (**gadgetBuilderSettings / blocks / block / includes / include / args**) : 必須  
インクルードJSPに対する引数を設定します。  
詳細は「[インクルードJSP](#)」を参照してください。

**i** コラム

インクルードJSPは、設定順でインクルードされます。  
設定順によって想定処理が行われない場合があるので、優先順位を確認して設定してください。

## ガジェットサーフィス設定

ガジェットサーフィス設定では、ガジェットが使用するサーフィスを設定します。

ガジェットサーフィス設定は、「gadgetBuilderSettings / surfaces / surface」階層に設定を行います。  
「gadgetBuilderSettings / surfaces / surface」階層は、複数設定が可能です。

## 設定

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

  <!-- 略 -->

  <surfaces>
    <surface surfaceKey="chapterTitle">
      <name localeId="en" surface="Expenses Particulars"/>
      <name localeId="ja" surface="経費明細"/>
      <name localeId="zh_CN" surface="[]明[]"/>
    </surface>

    <surface surfaceKey="rowNumber">
      <name localeId="en" surface="No"/>
      <name localeId="ja" surface="No"/>
      <name localeId="zh_CN" surface="No"/>
    </surface>

  <!-- 略 -->

</surfaces>

<!-- 略 -->

</gadgetBuilderSettings>
```

**i** コラム

ガジェットサーフィス設定は、申請書ガジェットサーフィスマスタへ登録されます。

## 申請書ガジェットサーフィスマスタ 設定値

サーフィスキー	ガジェットサーフィス設定のサーフィスキー
ロケールID	ガジェットサーフィス設定のロケールID
サーフィス	ガジェットサーフィス設定のサーフィス

## 解説

- サーフィスキー (**gadgetBuilderSettings / surfaces / surface @ surfaceKey**) : 必須  
サーフィスキーを設定します。
- ロケールID (**gadgetBuilderSettings / surfaces / surface / name @ localeId**) : 必須  
ロケールIDを設定します。
- サーフィス (**gadgetBuilderSettings / surfaces / surface / name @ surface**) : 必須  
サーフィスを設定します。



### コラム

ロケールIDの設定では、システムとして使用していないロケールは設定不要です。

## ガジェットタグ設定

ガジェットタグ設定では、ガジェットの属性を設定します。

ガジェット概要設定は、「gadgetBuilderSettings / tags / tag」階層に設定を行います。  
「gadgetBuilderSettings / tags / tag」階層は、複数設定が可能です。

### 設定

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

<!-- 略 -->

<tags>
  <tag value="appNormal"/>
  <tag value="appAdvance"/>
  <tag value="appSettle"/>
  <tag value="spSupport"/>
</tags>
</gadgetBuilderSettings>
```



### コラム

ガジェットタグ設定は、申請書ガジェットタグマスタへ登録されます。

#### 申請書ガジェットタグマスタ 設定値

タグ	ガジェットタグ設定のタグ

### 解説

- タグ (**gadgetBuilderSettings / tags / tag @ value**) : 必須  
ガジェットの属性を判定するタグを設定します。

## i コラム

次のタグが存在し、設定することでガジェットに属性が付与され、ガジェット属性ごとの振る舞いを定義できます。  
intra-mart Accel Kaiden!標準のガジェットには次のタグ以外も設定されていますが、将来拡張用のタグです。

タグ分類	タグ名	説明
制約	constraintDuplicationSelf	申請書内で自身と重複を禁止します。
制約	constraintDuplicationClass	申請書内で同ガジェットクラスと重複を禁止します。
申請書タイプ	appAdvance	対象申請書タイプ「事前申請」を設定します。 申請書メンテナンスで申請書タイプ「事前申請書」を選択した場合、選択できるガジェットです。
申請書タイプ	appSettle	対象申請書タイプ「精算申請」を設定します。 申請書メンテナンスで申請書タイプ「精算申請書」を選択した場合、選択できるガジェットです。
申請書タイプ	appNormal	対象申請書タイプ「一般」を設定します。 申請書メンテナンスで申請書タイプ「一般」を選択した場合、選択できるガジェットです。
申請書タイプ	appBill	対象申請書タイプ「請求書払申請」を設定します。 申請書メンテナンスで申請書タイプ「請求書払申請書」を選択した場合、選択できるガジェットです。
SPサポート	spSupport	SP(スマートフォン)画面をサポートする場合に設定します。
SPサポート対象外	spNoSupport	SP(スマートフォン)画面をサポートしない場合に設定します。
チェックルール	checkRule	申請書メンテナンスのチェックルール画面で入力チェックの設定を可能にする場合設定します。 申請書メンテナンスの詳細は、『 <a href="#">intra-mart Accel Kaiden! 経費旅費 / 管理者操作ガイド</a> 』を参照してください。

## Sample

次のGadgetBuilder定義は経費明細ガジェットの設定を行った場合のサンプルです。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">
  <gadgetClass>expenseDetail</gadgetClass>
  <gadgetVariation>v01</gadgetVariation>
  <jspPath>/WEB-INF/view/kaiden2/extension_imw/generic/gadget/generic_v01.jsp</jspPath>
  <javaPath>jp.co.slcs.kaiden2.extension_imw.feature.logic.kaiden.gadget.builder.GadgetBuilderLogic</javaPath>
  <sortKey>51001</sortKey>
  <surfaceKey>chapterTitle</surfaceKey>
  <descriptions>
    <description localeId="en" gadgetName="Expenses Particulars" description="Enter Expenses Particulars.&#xD;Only Japanese Yen is available."/>
    <description localeId="ja" gadgetName="経費明細" description="経費明細の入力を行います。&#xD;日本円のみ利用可能です。"/>
    <description localeId="zh_CN" gadgetName="明細" description="输入明细。&#xD;只可使用日元。"/>
  </descriptions>
  <blocks>
    <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">
      <service basic="k30tExpenseDetailsService" archive="ka30tExpenseDetailsService">
        <sqls>
          <sql id="init" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="copy" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="merge" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="archive" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="select" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
        </sqls>
      </service>
    </block>
  </blocks>
  <groups>
    <group id="rowNumber" surfaceKey="rowNumber" requiredType="ALWAYS_ANY" inputType="LABEL">
      <args>
        <arg key="style_pc" value="text-align:center;"/>
      </args>
      <fields>
        <field id="rowNumber" surfaceKey="rowNumber" requiredType="INHERIT" inputType="INHERIT" type="PANEL"/>
        <field id="imwApplyBaseDate" surfaceKey="imwApplyBaseDate" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DTO"/>
        <field id="searchCriteriaCompany" surfaceKey="searchCriteriaCompany" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DTO"/>
      </fields>
    </group>
  </groups>
</gadgetBuilderSettings>
```

```

<field id="payMethodCd" surfaceKey="payMethod" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
</fields>
</group>
<group id="occurDate" surfaceKey="occurDate" requiredType="ALWAYS_REQ" inputType="CONFIG">
<fields>
<field id="occurDate" surfaceKey="occurDate" requiredType="INHERIT" inputType="INHERIT" type="TEXT_DATE">
<args>
<arg key="style_pc" value="max-width:70px; width:calc(100% - 40px);"/>
<arg key="title" value="occurDate"/>
<arg key="placeholder" value="occurDate"/>
<arg key="autoFormat" value="true"/>
</args>
<validates>
<validate event="default" config="{ 'id':'date' }"/>
</validates>
</field>
</fields>
</group>
<group id="summary" surfaceKey="summary" requiredType="SETTING_REQ" inputType="CONFIG">
<fields>
<field id="summary" surfaceKey="summary" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
<args>
<arg key="maxLength" value="250"/>
<arg key="style_pc" value="width:calc(100% - 10px);"/>
<arg key="title" value="summary"/>
<arg key="placeholder" value="summary"/>
</args>
<validates>
<validate event="default" config="{ 'id':'maxLength', 'args':'250' }"/>
</validates>
</field>
</fields>
</group>
<group id="expType" surfaceKey="expType" requiredType="ALWAYS_REQ" inputType="CONFIG">
<fields>
<field id="expTypeCd" surfaceKey="expType" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
<args>
<arg key="masterId" value="exp_type_filtering"/>
<arg key="extensionId" value="expenseDetail"/>
<arg key="searchType" value="search"/>
<arg key="viewName" value="expTypeName"/>
<arg key="criteria" value="
{ 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany', 'expTypeCd':'expTypeCd', 'filteringType':'gadget,con

<arg key="resultChain" value="
{ 'expTypeCd':'expTypeCd', 'expTypeName':'expTypeName', 'accountCd':'accountCd', 'accountName':'accountName', 'taxTypeCd':'taxTypeCd', 'taxT

<arg key="resultValue" value="expTypeName"/>
<arg key="primaryTarget" value="expTypeCd"/>
<arg key="tmtableId" value="expenseDetail"/>
<arg key="refreshTarget" value="imwApplyBaseDate,searchCriteriaCompany"/>
<arg key="style_pc" value="width:calc(40% - 35px);"/>
<arg key="title" value="expType"/>
<arg key="placeholder" value="expType"/>
</args>
<validates>
<validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'exp_type_filtering',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'expTypeCd':'expTypeCd', 'filteringType':'gadget,contents', 'filterGadgetClass':'expenseDetail', 'filterGadgetVariation':'v01', 'filterContentsId':'imwC

</validates>
</field>
<field id="imwContentsId" surfaceKey="imwContentsId" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DTO"/>
<field id="accountCd" surfaceKey="account" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN">
<validates>
<validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'account',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'accountCd':'accountCd' } }"/>
</validates>
</field>
<field id="accountName" surfaceKey="account" requiredType="ALWAYS_ANY" inputType="LABEL" type="TEXT">
<args>
<arg key="class_pc" value="kaiden-ui-dummey-text label-overflow"/>
<arg key="style_pc" value="width:calc(30% - 15px);"/>
<arg key="title" value="account"/>
</args>
</field>
<field id="taxTypeCd" surfaceKey="taxType" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN">

```

```

<validates>
  <validate event="default" target="SERVER" config="{ 'id': 'masterBuilderExists', 'args': { 'masterId': 'tax_type',
'extensionId': 'expenseDetail', 'searchCriteriaDate': 'imwApplyBaseDate', 'searchCriteriaCompany': 'searchCriteriaCompany',
'taxTypeCd': 'taxTypeCd' } }"/>
</validates>
</field>
<field id="taxTypeName" surfaceKey="taxType" requiredType="ALWAYS_ANY" inputType="LABEL" type="TEXT">
<args>
  <arg key="class_pc" value="kaiden-ui-dummey-text label-overflow"/>
  <arg key="style_pc" value="width:calc(30% - 15px);"/>
  <arg key="title" value="taxType"/>
</args>
</field>
<field id="taxRate" surfaceKey="taxRate" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
</fields>
</group>
<group id="department" surfaceKey="department" requiredType="ALWAYS_ANY" inputType="CONFIG">
<fields>
  <field id="departmentSetCd" surfaceKey="departmentSet" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
  <field id="departmentCd" surfaceKey="department" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
<args>
  <arg key="masterId" value="department"/>
  <arg key="extensionId" value="expenseDetail"/>
  <arg key="searchType" value="search"/>
  <arg key="viewName" value="departmentName"/>
  <arg key="criteria" value="
{ 'searchCriteriaDate': 'imwApplyBaseDate', 'searchCriteriaCompany': 'searchCriteriaCompany', 'departmentSetCd': 'searchCriteriaCompany', 'depart
  <arg key="resultChain" value="
{ 'departmentCd': 'departmentCd', 'departmentName': 'departmentName', 'departmentSetCd': 'departmentSetCd' }"/>
  <arg key="resultValue" value="departmentName"/>
  <arg key="primaryTarget" value="departmentSetCd, departmentCd"/>
  <arg key="tmtableId" value="expenseDetail"/>
  <arg key="refreshTarget" value="imwApplyBaseDate, searchCriteriaCompany"/>
  <arg key="style_pc" value="width:calc(100% - 35px);"/>
  <arg key="title" value="department"/>
  <arg key="placeholder" value="department"/>
</args>
</validates>
<validate event="apply" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="reapply" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="process" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="applysp" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="reapplysp" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="processsp" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="default" target="SERVER" config="{ 'id': 'masterBuilderExists', 'args': { 'masterId': 'department',
'extensionId': 'expenseDetail', 'searchCriteriaDate': 'imwApplyBaseDate', 'searchCriteriaCompany': 'searchCriteriaCompany',
'departmentSetCd': 'searchCriteriaCompany', 'departmentCd': 'departmentCd' } }"/>
</validates>
</field>
</fields>
</group>
<group id="project" surfaceKey="project" requiredType="ALWAYS_ANY" inputType="CONFIG">
<fields>
  <field id="projectCd" surfaceKey="project" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
<args>
  <arg key="masterId" value="project_workflow"/>
  <arg key="extensionId" value="expenseDetail"/>
  <arg key="searchType" value="search"/>
  <arg key="viewName" value="projectName"/>
  <arg key="criteria" value="
{ 'searchCriteriaDate': 'imwApplyBaseDate', 'searchCriteriaCompany': 'searchCriteriaCompany', 'projectCd': 'projectCd', 'includeBlank': 'true' }"/>
  <arg key="resultChain" value="{ 'projectCd': 'projectCd', 'projectName': 'projectName' }"/>
  <arg key="resultValue" value="projectName"/>
  <arg key="primaryTarget" value="projectCd"/>
  <arg key="tmtableId" value="expenseDetail"/>
  <arg key="refreshTarget" value="imwApplyBaseDate, searchCriteriaCompany"/>
  <arg key="style_pc" value="width:calc(100% - 35px);"/>
  <arg key="title" value="project"/>
  <arg key="placeholder" value="project"/>
</args>
</validates>
<validate event="apply" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="reapply" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="process" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="applysp" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="reapplysp" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>
<validate event="processsp" target="ALWAYS" config="{ 'id': 'requiredAny', 'args': [ 'departmentCd', 'projectCd' ] }"/>

```

```

<validate event="processop" target="ALWAYS_REQ" config="{ 'id':'requiredAny', 'args':{'departmentCd':'projectCd'}}"/>
<validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'project_workflow',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'projectCd':'projectCd'}}"/>
</validates>
</field>
</fields>
</group>
<group id="payType" surfaceKey="payType" requiredType="ALWAYS_REQ" inputType="CONFIG">
<fields>
<field id="payTypeCd" surfaceKey="payType" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
<args>
<arg key="masterId" value="pay_type"/>
<arg key="extensionId" value="expenseDetail"/>
<arg key="searchType" value="select"/>
<arg key="viewName" value="payTypeName"/>
<arg key="criteria" value=""
{ 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany', 'payTypeCd':'payTypeCd', 'includeBlank':'false' }"/>

<arg key="resultChain" value="{ 'payTypeCd':'payTypeCd', 'payTypeName':'payTypeName', 'paymentFlag':'paymentFlag' }"/>
<arg key="resultValue" value="payTypeName"/>
<arg key="primaryTarget" value="payTypeCd"/>
<arg key="tntableId" value="expenseDetail"/>
<arg key="refreshTarget" value="imwApplyBaseDate,searchCriteriaCompany"/>
<arg key="style_pc" value="width:calc(100% - 10px);"/>
<arg key="title" value="payType"/>
<arg key="placeholder" value="payType"/>
</args>
<validates>
<validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'pay_type',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'payTypeCd':'payTypeCd'}}"/>
</validates>
</field>
<field id="paymentFlag" surfaceKey="paymentFlag" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
</fields>
</group>
<group id="payee" surfaceKey="payee" requiredType="SETTING_ANY" inputType="CONFIG">
<fields>
<field id="payeeName" surfaceKey="payee" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
<args>
<arg key="maxLength" value="250"/>
<arg key="style_pc" value="width:calc(100% - 10px);"/>
<arg key="title" value="payee"/>
<arg key="placeholder" value="payee"/>
</args>
<validates>
<validate event="default" config="{ 'id':'maxLength', 'args':{'250'}}"/>
</validates>
</field>
</fields>
</group>
<group id="transAmount" surfaceKey="transAmount" requiredType="ALWAYS_REQ" inputType="CONFIG">
<fields>
<field id="transAmount" surfaceKey="transAmount" requiredType="INHERIT" inputType="INHERIT" type="TEXT_NUMERIC">
<args>
<arg key="style_pc" value="width:calc(100% - 10px);"/>
<arg key="title" value="transAmount"/>
<arg key="placeholder" value="transAmount"/>
<arg key="amount" value="true"/>
<arg key="event" value="change"/>
</args>
<validates>
<validate event="default" config="{ 'id':'currencyAmnt2', 'args':{'','searchCriteriaCompany'}}"/>
<validate event="default" config="{ 'id':'amountMin', 'args':{'0'}}"/>
</validates>
</field>
<field id="transCurrencyCd" surfaceKey="transCurrency" requiredType="ALWAYS_ANY" inputType="INHERIT"
type="TEXT_HIDDEN"/>
<field id="companyRate" surfaceKey="companyRate" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
<field id="companyAmount" surfaceKey="companyAmount" requiredType="ALWAYS_ANY" inputType="INHERIT"
type="TEXT_HIDDEN"/>
<field id="companyCurrencyCd" surfaceKey="companyCurrency" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DTO"/>
<field id="quantity" surfaceKey="quantity" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
<field id="companyAmountSum" surfaceKey="companyAmountSum" requiredType="INHERIT" inputType="INHERIT"
type="TEXT_HIDDEN"/>
<field id="companyAmountExcTax" surfaceKey="companyAmountExcTax" requiredType="INHERIT" inputType="INHERIT"
type="TEXT_HIDDEN"/>

```

```

<field id="companyAmountTax" surfaceKey="companyAmountTax" requiredType="INHERIT" inputType="INHERIT"
type="TEXT_HIDDEN"/>
</fields>
</group>
</groups>
<table rowNumberTarget="rowNumber" type="HORIZONTAL">
<colgroups>
<colgroup style="width:50px; overflow:hidden"/>
<colgroup style="overflow:hidden"/>
<colgroup style="width:23%; overflow:hidden"/>
<colgroup style="width:23%; overflow:hidden"/>
<colgroup style="width:23%; overflow:hidden"/>
</colgroups>
<rows>
<row>
<cols>
<col groupId="rowNumber" rowspan="3"/>
<col groupId="occurDate"/>
<col groupId="summary" colspan="3"/>
</cols>
</row>
<row>
<cols>
<col groupId="expType" colspan="2"/>
<col groupId="department"/>
<col groupId="project"/>
</cols>
</row>
<row>
<cols>
<col groupId="payType"/>
<col groupId="payee" colspan="2"/>
<col groupId="transAmount"/>
</cols>
</row>
</rows>
</table>
<completions>
<completion name="departmentSetCd" position="1" type="claimValueOnInit" args="{ 'claim':'claimDefaultDepartmentSetCd',
'target':'1'}" target="CLIENT" mode="EMPTY"/>
<completion name="departmentCd" position="1" type="claimValueOnInit" args="{ 'claim':'claimDefaultDepartmentCd',
'target':'1'}" target="CLIENT" mode="EMPTY"/>
<completion name="projectCd" position="1" type="claimValueOnInit" args="{ 'claim':'claimDefaultProjectCd', 'target':'1'}"
target="CLIENT" mode="EMPTY"/>
<completion name="departmentSetCd" position="1" type="claimValueAddRow" args="{ 'claim':'claimDefaultDepartmentSetCd',
'target':'1'}" target="CLIENT" mode="EMPTY"/>
<completion name="departmentCd" position="1" type="claimValueAddRow" args="{ 'claim':'claimDefaultDepartmentCd',
'target':'1'}" target="CLIENT" mode="EMPTY"/>
<completion name="projectCd" position="1" type="claimValueAddRow" args="{ 'claim':'claimDefaultProjectCd', 'target':'1'}"
target="CLIENT" mode="EMPTY"/>
<completion name="payMethodCd" position="1" type="claimValue" args="{ 'claim':'claimPayMethod'}"
target="ALWAYS" mode="ALWAYS"/>
<completion name="transCurrencyCd" position="2" type="variable" args="{ 'field':'companyCurrencyCd'}"
target="ALWAYS" mode="ALWAYS"/>
<completion name="companyRate" position="3" type="fixed" args="{ 'value':'1'}"
target="ALWAYS" mode="EMPTY"/>
<completion name="companyAmount" position="4" type="variable" args="{ 'field':'transAmount'}"
target="ALWAYS" mode="ALWAYS"/>
<completion name="quantity" position="5" type="fixed" args="{ 'value':'1'}"
target="ALWAYS" mode="EMPTY"/>
<completion name="companyAmountSum" position="6" type="calcMul" args="
{'field1':'companyAmount','field2':'quantity','round':'4','digits':'0'}" target="ALWAYS" mode="ALWAYS"/>
<completion name="companyAmountTax" position="7" type="calcTax" args="
{'amount':'companyAmountSum','taxRate':'taxRate','round':'4','digits':'0'}" target="ALWAYS" mode="ALWAYS"/>
<completion name="companyAmountExcTax" position="8" type="calcSub" args="
{'field1':'companyAmountSum','field2':'companyAmountTax','round':'4','digits':'0'}" target="ALWAYS" mode="ALWAYS"/>
</completions>
<claims>
<claim name="claimExpenseAmount" type="claimAmountSummary" args="{ 'amount':'companyAmountSum'}"/>
<claim name="claimTemporaryPayAmount" type="claimAmountSummary" args="
{'amount':'companyAmountSum','paymentFlag':'paymentFlag','payment':'1'}"/>
<claim name="claimCompanyPayAmount" type="claimAmountSummary" args="
{'amount':'companyAmountSum','paymentFlag':'paymentFlag','payment':'0'}"/>
<claim name="claimDateDetail" type="claimColumnValuesList" args="{ 'field1':'occurDate','field2':'occurDate'}"/>
<claim name="claimExpenseDetail" type="claimColumnValuesList" args="{ 'field1':'companyAmountSum'}"/>
<claim name="claimExpenseCount" type="claimRowCount" args=""/>
<claim name="loadData4Settle" type="claimSearchData" args="{ 'systemMatterId':'imwSystemMatterId',
'userDatald':'imwUserDatald','gadgetClass':'gadgetClass'}" target="CLIENT"/>

```

```

</claims>
<events>
  <event target="matter_end" name="journalSourceDetailEntryForSqlEvent">
    <args>
      <arg key="sqlPath" value="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/journal/ExpenseDetail.sql"/>
    </args>
  </event>
</events>
<includes>
  <include path="/WEB-INF/view/kaiden2/product_workflow/generic/gadget/include/defaultDepartmentProject.jsp" />
</includes>
</block>
</blocks>
<surfaces>
  <surface surfaceKey="chapterTitle">
    <name localeId="en" surface="Expenses Particulars"/>
    <name localeId="ja" surface="経費明細"/>
    <name localeId="zh_CN" surface="□□明□"/>
  </surface>

  <surface surfaceKey="rowNumber">
    <name localeId="en" surface="No"/>
    <name localeId="ja" surface="No"/>
    <name localeId="zh_CN" surface="No"/>
  </surface>

  <surface surfaceKey="imwApplyBaseDate">
    <name localeId="en" surface="Apply Reference Date"/>
    <name localeId="ja" surface="申請基準日"/>
    <name localeId="zh_CN" surface="申□基準日"/>
  </surface>

  <surface surfaceKey="searchCriteriaCompany">
    <name localeId="en" surface="Search Reference Company"/>
    <name localeId="ja" surface="検索基準会社"/>
    <name localeId="zh_CN" surface="□索基准公司"/>
  </surface>

  <surface surfaceKey="payMethod">
    <name localeId="en" surface="Payment Method"/>
    <name localeId="ja" surface="支払方法"/>
    <name localeId="zh_CN" surface="支付方法"/>
  </surface>

  <surface surfaceKey="occurDate">
    <name localeId="en" surface="Occurrence Date"/>
    <name localeId="ja" surface="発生日"/>
    <name localeId="zh_CN" surface="□生日"/>
  </surface>

  <surface surfaceKey="summary">
    <name localeId="en" surface="Description of Expense"/>
    <name localeId="ja" surface="経費内容"/>
    <name localeId="zh_CN" surface="□□内容"/>
  </surface>

  <surface surfaceKey="expType">
    <name localeId="en" surface="Expense Type"/>
    <name localeId="ja" surface="経費区分"/>
    <name localeId="zh_CN" surface="□□分□"/>
  </surface>

  <surface surfaceKey="account">
    <name localeId="en" surface="Account"/>
    <name localeId="ja" surface="勘定科目"/>
    <name localeId="zh_CN" surface="□□科目"/>
  </surface>

  <surface surfaceKey="taxType">
    <name localeId="en" surface="Tax Category"/>
    <name localeId="ja" surface="税区分"/>
    <name localeId="zh_CN" surface="税分□"/>
  </surface>

  <surface surfaceKey="taxRate">
    <name localeId="en" surface="Tax Rate"/>
    <name localeId="ja" surface="税率"/>
    <name localeId="zh_CN" surface="税率"/>
  </surface>

  <surface surfaceKey="departmentSet">
    <name localeId="en" surface="Organization Set"/>
    <name localeId="ja" surface="組織セット"/>
    <name localeId="zh_CN" surface="□□集合"/>
  </surface>

  <surface surfaceKey="department">
    <name localeId="en" surface="Organization"/>
    <name localeId="ja" surface="組織"/>

```

```

<name localeId="ja" surface="社職"/>
<name localeId="zh_CN" surface=""/>
</surface>
<surface surfaceKey="project">
<name localeId="en" surface="Project"/>
<name localeId="ja" surface="プロジェクト"/>
<name localeId="zh_CN" surface="项目"/>
</surface>
<surface surfaceKey="payType">
<name localeId="en" surface="Payment Classification"/>
<name localeId="ja" surface="支払分類"/>
<name localeId="zh_CN" surface="支付分"/>
</surface>
<surface surfaceKey="paymentFlag">
<name localeId="en" surface="Reimbursement Flag"/>
<name localeId="ja" surface="立替フラグ"/>
<name localeId="zh_CN" surface="付志"/>
</surface>
<surface surfaceKey="payee">
<name localeId="en" surface="Payee"/>
<name localeId="ja" surface="支払先"/>
<name localeId="zh_CN" surface="收款方"/>
</surface>
<surface surfaceKey="transAmount">
<name localeId="en" surface="Amount"/>
<name localeId="ja" surface="金額"/>
<name localeId="zh_CN" surface="金"/>
</surface>
<surface surfaceKey="transCurrency">
<name localeId="en" surface="Currency"/>
<name localeId="ja" surface="通貨"/>
<name localeId="zh_CN" surface=""/>
</surface>
<surface surfaceKey="companyRate">
<name localeId="en" surface="rate"/>
<name localeId="ja" surface="レート"/>
<name localeId="zh_CN" surface="率"/>
</surface>
<surface surfaceKey="companyAmount">
<name localeId="en" surface="Converted"/>
<name localeId="ja" surface="換算額"/>
<name localeId="zh_CN" surface="算"/>
</surface>
<surface surfaceKey="companyCurrency">
<name localeId="en" surface="Company Currency"/>
<name localeId="ja" surface="会社通貨"/>
<name localeId="zh_CN" surface="公司"/>
</surface>
<surface surfaceKey="quantity">
<name localeId="en" surface="Quantity"/>
<name localeId="ja" surface="数量"/>
<name localeId="zh_CN" surface="数量"/>
</surface>
<surface surfaceKey="companyAmountSum">
<name localeId="en" surface="Total amount"/>
<name localeId="ja" surface="合計額"/>
<name localeId="zh_CN" surface="量"/>
</surface>
<surface surfaceKey="companyAmountExcTax">
<name localeId="en" surface="Tax excluded amount"/>
<name localeId="ja" surface="税抜金額"/>
<name localeId="zh_CN" surface="税抜金量"/>
</surface>
<surface surfaceKey="companyAmountTax">
<name localeId="en" surface="Tax amount"/>
<name localeId="ja" surface="税額"/>
<name localeId="zh_CN" surface="税"/>
</surface>
</surfaces>
<tags>
<tag value="appNormal"/>
<tag value="appAdvance"/>
<tag value="appSettle"/>
<tag value="spSupport"/>
<tag value="checkRule"/>
</tags>
</gadgetBuilderSettings>

```

## 設定の流れ（申請書ガジェット）

本項では、GadgetBuilderの設定の流れを説明します。  
（経費明細を例にします。）

- テーブル定義
  - 必須カラム
    - 基本テーブル
    - 退避テーブル
    - 定義サンプル
- Service、Entity生成
  - S2JDBC-Genのセットアップ
  - Service、Entityの作成
  - Service、Entityの配置とコンポーネント設定
- GadgetBuilder定義作成
  - GadgetBuilder定義ファイル作成
- SQL作成
- 動作確認
  - GadgetBuilder定義インポート
  - 申請書マスタメンテナンス
  - 申請画面

### テーブル定義

はじめに、対象となるテーブルの定義を行います。  
要件に合わせて次の各カラムを組み合わせることで定義を行ってください。

#### 必須カラム

GadgetBuilderで申請書ガジェットを作成する場合のテーブルの定義です。

#### 基本テーブル

基本テーブルの場合は次のカラムは必ず設定してください。（全て「not null」制約）

論理名	物理名	PK	PostgreSQL	Oracle	SqlServer
システム案件ID	system_matter_id	○	varchar(20)	varchar2(20)	nvarchar(20)
ユーザデータID	user_data_id	○	varchar(20)	varchar2(20)	nvarchar(20)
ガジェットID	gadget_id		varchar(220)	varchar2(220)	nvarchar(220)
ガジェットクラス	gadget_class	○	varchar(100)	varchar2(100)	nvarchar(100)
ガジェットバリエーション	gadget_variation		varchar(100)	varchar2(100)	nvarchar(100)
ガジェットインスタンス	gadget_instance	○	numeric(18,0)	numeric(18,0)	numeric(18,0)
タプルID	tuple_id		numeric(18,0)	numeric(18,0)	numeric(18,0)
行番号	row_number		numeric(18,0)	numeric(18,0)	numeric(18,0)
登録日時	entry_ts		timestamp without time zone	timestamp	datetime2
登録ユーザ	entry_user_cd		varchar(100)	varchar2(100)	nvarchar(100)
更新カウント	renew_cnt		numeric(18,0)	numeric(18,0)	numeric(18,0)
更新日時	renew_ts		timestamp without time zone	timestamp	datetime2
更新ユーザ	renew_user_cd		varchar(100)	varchar2(100)	nvarchar(100)

#### 退避テーブル

退避（アーカイブ）テーブルの場合は次のカラムは必ず設定してください。（全て「not null」制約）

論理名	物理名	PK	PostgreSQL	Oracle	SqlServer
システム案件ID	system_matter_id	○	varchar(20)	varchar2(20)	nvarchar(20)
ユーザデータID	user_data_id	○	varchar(20)	varchar2(20)	nvarchar(20)
ガジェットID	gadget_id		varchar(220)	varchar2(220)	nvarchar(220)
ガジェットクラス	gadget_class	○	varchar(100)	varchar2(100)	nvarchar(100)
ガジェットバリエーション	gadget_variation		varchar(100)	varchar2(100)	nvarchar(100)
ガジェットインスタンス	gadget_instance	○	numeric(18,0)	numeric(18,0)	numeric(18,0)
ロケールID	locale_id	○	varchar(50)	varchar2(50)	nvarchar(50)
アーカイブ月	archive_month		varchar(6)	varchar2(6)	nvarchar(6)
タプルID	tuple_id		numeric(18,0)	numeric(18,0)	numeric(18,0)
行番号	row_number		numeric(18,0)	numeric(18,0)	numeric(18,0)
登録日時	entry_ts		timestamp without time zone	timestamp	datetime2
登録ユーザ	entry_user_cd		varchar(100)	varchar2(100)	nvarchar(100)
更新カウント	renew_cnt		numeric(18,0)	numeric(18,0)	numeric(18,0)
更新日時	renew_ts		timestamp without time zone	timestamp	datetime2
更新ユーザ	renew_user_cd		varchar(100)	varchar2(100)	nvarchar(100)

### コラム

タプルID、行番号は明細系ブロックの登録先の場合のみ設定してください。

### 定義サンプル

次項以降では、次の定義を利用した場合の設定の流れを説明します。

#### ■ PostgreSQL

```
create table k30t_expense_detail
(
  system_matter_id      varchar(20) not null,
  user_data_id          varchar(20) not null,
  gadget_class          varchar(100) not null,
  gadget_instance       numeric(18,0) not null,
  tuple_id              numeric(18,0) not null,
  gadget_id             varchar(220) not null,
  gadget_variation      varchar(100) not null,
  row_number            numeric(18,0) not null,
  occur_date            timestamp without time zone,
  summary               varchar(4000),
  exp_type_cd           varchar(100),
  account_cd            varchar(100),
  tax_type_cd           varchar(100),
  tax_rate              numeric(28,10),
  quantity              numeric(28,10),
  trans_currency_cd     varchar(100),
  trans_amount          numeric(28,10),
  company_currency_cd   varchar(100),
  company_rate          numeric(28,10),
  company_amount        numeric(28,10),
  company_amount_exc_tax numeric(28,10),
  company_amount_tax    numeric(28,10),
  company_amount_sum    numeric(28,10),
  pay_type_cd           varchar(100),
  pay_method_cd         varchar(100),
  payee_cd              varchar(100),
  payee_name            varchar(1000),
  department_set_cd     varchar(100),
  department_cd         varchar(100),
  project_cd            varchar(100),
```

```

extension_a      varchar(1000),
extension_b      varchar(1000),
extension_c      varchar(1000),
extension_d      varchar(1000),
extension_e      varchar(1000),
extension_f      varchar(1000),
extension_g      varchar(1000),
extension_h      varchar(1000),
extension_i      varchar(1000),
extension_j      varchar(1000),
entry_ts        timestamp without time zone not null,
entry_user_cd   varchar(100) not null,
renew_cnt       numeric(18,0) not null,
renew_ts        timestamp without time zone not null,
renew_user_cd   varchar(100) not null
);
alter table k30t_expense_detail add primary key (system_matter_id, user_data_id, tuple_id, gadget_class, gadget_instance);
alter table k30t_expense_detail add check (gadget_instance >= 0);
alter table k30t_expense_detail add check (tuple_id >= 0);
alter table k30t_expense_detail add check (row_number >= 0);
alter table k30t_expense_detail add check (tax_rate >= 0);

create table ka30t_expense_detail
(
  system_matter_id      varchar(20) not null,
  user_data_id          varchar(20) not null,
  gadget_class          varchar(100) not null,
  gadget_instance       numeric(18,0) not null,
  tuple_id              numeric(18,0) not null,
  locale_id             varchar(50) not null,
  archive_month         varchar(6) not null,
  gadget_id             varchar(220) not null,
  gadget_variation      varchar(100) not null,
  row_number            numeric(18,0) not null,
  occur_date           timestamp without time zone,
  summary               varchar(4000),
  exp_type_cd          varchar(100),
  account_cd           varchar(100),
  tax_type_cd          varchar(100),
  tax_rate              numeric(28,10),
  quantity              numeric(28,10),
  trans_currency_cd    varchar(100),
  trans_amount          numeric(28,10),
  company_currency_cd  varchar(100),
  company_rate          numeric(28,10),
  company_amount       numeric(28,10),
  company_amount_exc_tax numeric(28,10),
  company_amount_tax   numeric(28,10),
  company_amount_sum   numeric(28,10),
  pay_type_cd          varchar(100),
  pay_method_cd        varchar(100),
  payee_cd             varchar(100),
  payee_name           varchar(1000),
  department_set_cd    varchar(100),
  department_cd        varchar(100),
  project_cd           varchar(100),
  exp_type_name        varchar(1000),
  account_name         varchar(1000),
  tax_type_name        varchar(1000),
  trans_currency_name  varchar(1000),
  company_currency_name varchar(1000),
  pay_type_name        varchar(1000),
  payment_flag         varchar(1),
  pay_method_name     varchar(1000),
  department_name      varchar(1000),
  project_name         varchar(1000),
  extension_a          varchar(1000),
  extension_b          varchar(1000),
  extension_c          varchar(1000),
  extension_d          varchar(1000),
  extension_e          varchar(1000),
  extension_f          varchar(1000),
  extension_g          varchar(1000),
  extension_h          varchar(1000),
  extension_i          varchar(1000),
  extension_j          varchar(1000),
  entry_ts             timestamp without time zone not null,
  entrv user cd        varchar(100) not null.

```

```

renew_cnt          numeric(18,0) not null,
renew_ts           timestamp without time zone not null,
renew_user_cd      varchar(100) not null
);
alter table ka30t_expense_detail add primary key (system_matter_id, user_data_id, gadget_class, gadget_instance, tuple_id, locale_id);

create index idx_ka30t_expense_detail_1 on ka30t_expense_detail (archive_month);

```

- Oracle

```

create table k30t_expense_detail
(
  system_matter_id      varchar2(20) not null,
  user_data_id          varchar2(20) not null,
  gadget_class          varchar2(100) not null,
  gadget_instance      numeric(18,0) not null,
  tuple_id              numeric(18,0) not null,
  gadget_id             varchar2(220) not null,
  gadget_variation     varchar2(100) not null,
  row_number            numeric(18,0) not null,
  occur_date           timestamp,
  summary              varchar2(4000),
  exp_type_cd          varchar2(100),
  account_cd           varchar2(100),
  tax_type_cd          varchar2(100),
  tax_rate              numeric(28,10),
  quantity             numeric(28,10),
  trans_currency_cd    varchar2(100),
  trans_amount         numeric(28,10),
  company_currency_cd  varchar2(100),
  company_rate         numeric(28,10),
  company_amount       numeric(28,10),
  company_amount_exc_tax numeric(28,10),
  company_amount_tax   numeric(28,10),
  company_amount_sum   numeric(28,10),
  pay_type_cd          varchar2(100),
  pay_method_cd        varchar2(100),
  payee_cd             varchar2(100),
  payee_name           varchar2(1000),
  department_set_cd    varchar2(100),
  department_cd        varchar2(100),
  project_cd           varchar2(100),
  extension_a          varchar2(1000),
  extension_b          varchar2(1000),
  extension_c          varchar2(1000),
  extension_d          varchar2(1000),
  extension_e          varchar2(1000),
  extension_f          varchar2(1000),
  extension_g          varchar2(1000),
  extension_h          varchar2(1000),
  extension_i          varchar2(1000),
  extension_j          varchar2(1000),
  entry_ts             timestamp not null,
  entry_user_cd        varchar2(100) not null,
  renew_cnt            numeric(18,0) not null,
  renew_ts             timestamp not null,
  renew_user_cd        varchar2(100) not null
);
alter table k30t_expense_detail add primary key (system_matter_id, user_data_id, tuple_id, gadget_class, gadget_instance);
alter table k30t_expense_detail add check (gadget_instance >= 0);
alter table k30t_expense_detail add check (tuple_id >= 0);
alter table k30t_expense_detail add check (row_number >= 0);
alter table k30t_expense_detail add check (tax_rate >= 0);

create table ka30t_expense_detail
(
  system_matter_id      varchar2(20) not null,
  user_data_id          varchar2(20) not null,
  gadget_class          varchar2(100) not null,
  gadget_instance      numeric(18,0) not null,
  tuple_id              numeric(18,0) not null,
  locale_id            varchar2(50) not null,
  archive_month         varchar2(6) not null,
  gadget_id             varchar2(220) not null,
  gadget_variation     varchar2(100) not null,
  row_number            numeric(18,0) not null

```

```

row_number          numeric(18,0) not null,
occur_date          timestamp,
summary             varchar2(4000),
exp_type_cd         varchar2(100),
account_cd          varchar2(100),
tax_type_cd         varchar2(100),
tax_rate            numeric(28,10),
quantity            numeric(28,10),
trans_currency_cd   varchar2(100),
trans_amount        numeric(28,10),
company_currency_cd varchar2(100),
company_rate        numeric(28,10),
company_amount      numeric(28,10),
company_amount_exc_tax numeric(28,10),
company_amount_tax  numeric(28,10),
company_amount_sum  numeric(28,10),
pay_type_cd         varchar2(100),
pay_method_cd       varchar2(100),
payee_cd            varchar2(100),
payee_name          varchar2(1000),
department_set_cd   varchar2(100),
department_cd       varchar2(100),
project_cd          varchar2(100),
exp_type_name       varchar2(1000),
account_name        varchar2(1000),
tax_type_name       varchar2(1000),
trans_currency_name varchar2(1000),
company_currency_name varchar2(1000),
pay_type_name       varchar2(1000),
payment_flag        varchar2(1),
pay_method_name     varchar2(1000),
department_name     varchar2(1000),
project_name        varchar2(1000),
extension_a         varchar2(1000),
extension_b         varchar2(1000),
extension_c         varchar2(1000),
extension_d         varchar2(1000),
extension_e         varchar2(1000),
extension_f         varchar2(1000),
extension_g         varchar2(1000),
extension_h         varchar2(1000),
extension_i         varchar2(1000),
extension_j         varchar2(1000),
entry_ts            timestamp not null,
entry_user_cd       varchar2(100) not null,
renew_cnt           numeric(18,0) not null,
renew_ts            timestamp not null,
renew_user_cd       varchar2(100) not null
);
alter table ka30t_expense_detail add primary key (system_matter_id, user_data_id, gadget_class, gadget_instance, tuple_id, locale_id);

create index idx_ka30t_expense_detail_1 on ka30t_expense_detail (archive_month);

```

## ■ SqlServer

```

create table k30t_expense_detail
(
system_matter_id      nvarchar(20) not null,
user_data_id         nvarchar(20) not null,
gadget_class         nvarchar(100) not null,
gadget_instance      numeric(18,0) not null,
tuple_id             numeric(18,0) not null,
gadget_id            nvarchar(220) not null,
gadget_variation     nvarchar(100) not null,
row_number           numeric(18,0) not null,
occur_date           datetime2,
summary              nvarchar(4000),
exp_type_cd          nvarchar(100),
account_cd           nvarchar(100),
tax_type_cd          nvarchar(100),
tax_rate             numeric(28,10),
quantity             numeric(28,10),
trans_currency_cd    nvarchar(100),
trans_amount         numeric(28,10),
company_currency_cd  nvarchar(100),
company_rate         numeric(28,10),
company_amount       numeric(28,10)
);

```

```

company_amount      numeric(28,10),
company_amount_exc_tax  numeric(28,10),
company_amount_tax   numeric(28,10),
company_amount_sum   numeric(28,10),
pay_type_cd         nvarchar(100),
pay_method_cd       nvarchar(100),
payee_cd            nvarchar(100),
payee_name          nvarchar(1000),
department_set_cd   nvarchar(100),
department_cd       nvarchar(100),
project_cd          nvarchar(100),
extension_a         nvarchar(1000),
extension_b         nvarchar(1000),
extension_c         nvarchar(1000),
extension_d         nvarchar(1000),
extension_e         nvarchar(1000),
extension_f         nvarchar(1000),
extension_g         nvarchar(1000),
extension_h         nvarchar(1000),
extension_i         nvarchar(1000),
extension_j         nvarchar(1000),
entry_ts            datetime2 not null,
entry_user_cd       nvarchar(100) not null,
renew_cnt           numeric(18,0) not null,
renew_ts            datetime2 not null,
renew_user_cd       nvarchar(100) not null
);
alter table k30t_expense_detail add primary key (system_matter_id, user_data_id, tuple_id, gadget_class, gadget_instance);
alter table k30t_expense_detail add check (gadget_instance >= 0);
alter table k30t_expense_detail add check (tuple_id >= 0);
alter table k30t_expense_detail add check (row_number >= 0);
alter table k30t_expense_detail add check (tax_rate >= 0);

create table ka30t_expense_detail
(
  system_matter_id      nvarchar(20) not null,
  user_data_id          nvarchar(20) not null,
  gadget_class          nvarchar(100) not null,
  gadget_instance       numeric(18,0) not null,
  tuple_id              numeric(18,0) not null,
  locale_id             nvarchar(50) not null,
  archive_month         nvarchar(6) not null,
  gadget_id             nvarchar(220) not null,
  gadget_variation      nvarchar(100) not null,
  row_number            numeric(18,0) not null,
  occur_date            datetime2,
  summary              nvarchar(4000),
  exp_type_cd          nvarchar(100),
  account_cd           nvarchar(100),
  tax_type_cd          nvarchar(100),
  tax_rate              numeric(28,10),
  quantity              numeric(28,10),
  trans_currency_cd    nvarchar(100),
  trans_amount         numeric(28,10),
  company_currency_cd  nvarchar(100),
  company_rate         numeric(28,10),
  company_amount       numeric(28,10),
  company_amount_exc_tax  numeric(28,10),
  company_amount_tax   numeric(28,10),
  company_amount_sum   numeric(28,10),
  pay_type_cd         nvarchar(100),
  pay_method_cd       nvarchar(100),
  payee_cd            nvarchar(100),
  payee_name          nvarchar(1000),
  department_set_cd   nvarchar(100),
  department_cd       nvarchar(100),
  project_cd          nvarchar(100),
  exp_type_name        nvarchar(1000),
  account_name        nvarchar(1000),
  tax_type_name        nvarchar(1000),
  trans_currency_name  nvarchar(1000),
  company_currency_name nvarchar(1000),
  pay_type_name        nvarchar(1000),
  payment_flag         nvarchar(1),
  pay_method_name     nvarchar(1000),
  department_name     nvarchar(1000),
  project_name         nvarchar(1000),
  extension_a         nvarchar(1000),

```

```

extension_b      nvarchar(1000),
extension_c      nvarchar(1000),
extension_d      nvarchar(1000),
extension_e      nvarchar(1000),
extension_f      nvarchar(1000),
extension_g      nvarchar(1000),
extension_h      nvarchar(1000),
extension_i      nvarchar(1000),
extension_j      nvarchar(1000),
entry_ts        datetime2 not null,
entry_user_cd   nvarchar(100) not null,
renew_cnt       numeric(18,0) not null,
renew_ts        datetime2 not null,
renew_user_cd   nvarchar(100) not null
);
alter table ka30t_expense_detail add primary key (system_matter_id, user_data_id, gadget_class, gadget_instance, tuple_id, locale_id);

create index idx_ka30t_expense_detail_1 on ka30t_expense_detail (archive_month);

```

## Service、Entity生成

次に「S2JDBC-Gen」を使用して、Service、Entityの作成を行います。



### 注意

Service、Entityの作成前に必ず、「[テーブル定義](#)」で定義したテーブルを作成してください。

## S2JDBC-Genのセットアップ

Service、Entityを作成するために、S2JDBC-Genをセットアップします。

セットアップ方法の詳細は『[intra-mart Accel Kaiden! プログラミングガイド](#)』の「基本 - 事前準備 - S2JDBC」を参照してください。

## Service、Entityの作成

Service、Entityを作成します。

作成方法の詳細は『[intra-mart Accel Kaiden! プログラミングガイド](#)』の「応用 - マスタメンテナンス - チュートリアル - S2JDBC」を参照してください。

## Service、Entityの配置とコンポーネント設定

作成したService、Entityをプロジェクトに配置します。

配置先のパス、およびパッケージは必要に応じて変更してください。

また、作成したServiceのコンポーネント追加先も使用プロジェクトのDiconファイルに追加してください。

(本説明では、intra-mart Accel Kaiden! ワークフローモジュールに配置した想定で作成しています。)

- **K30tExpenseDetail.java**

%ModuleProject%/src/main/generated/jp/co/slcs/kaiden2/product\_workflow/foundation/model/entity/K30tExpenseDetail.java

```

package jp.co.slcs.kaiden2.product_workflow.foundation.model.entity;

import java.io.Serializable;
import java.math.BigDecimal;
import java.sql.Timestamp;
import javax.annotation.Generated;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.Lob;
import javax.persistence.Table;
import jp.co.slcs.kaiden2.base.foundation.model.entity.GenerateEntity;

/**
 * 経費明細テーブル.
 *
 * @author Sumitomo Life Information Systems Co.,Ltd.
 */
@Entity
@Table(name = "k30t_expense_detail")
@Generated(value = {"S2JDBC-Gen 2.4.46", "org.seasar.extension.jdbc.gen.internal.model.EntityModelFactoryImpl"})
public final class K30tExpenseDetail extends GenerateEntity implements Serializable {

```

```

/** シリアルバージョンUID. */
private static final long serialVersionUID = 1L;

/** システム案件id. */
@Id
@Column(name = "system_matter_id", columnDefinition = "varchar2(20)", nullable = false, unique = false)
public String systemMatterId;

/** ユーザーデータid. */
@Id
@Column(name = "user_data_id", columnDefinition = "varchar2(20)", nullable = false, unique = false)
public String userDataId;

/** ガジェットクラス. */
@Id
@Column(name = "gadget_class", columnDefinition = "varchar2(100)", nullable = false, unique = false)
public String gadgetClass;

/** ガジェットインスタンス. */
@Id
@Column(name = "gadget_instance", columnDefinition = "number(18,0)", nullable = false, unique = false)
public Long gadgetInstance;

/** タブルid. */
@Id
@Column(name = "tuple_id", columnDefinition = "number(18,0)", nullable = false, unique = false)
public Long tupleId;

/** ガジェットid. */
@Column(name = "gadget_id", columnDefinition = "varchar2(220)", nullable = false, unique = false)
public String gadgetId;

/** ガジェットバリエーション. */
@Column(name = "gadget_variation", columnDefinition = "varchar2(100)", nullable = false, unique = false)
public String gadgetVariation;

/** 行番号. */
@Column(name = "row_number", columnDefinition = "number(18,0)", nullable = false, unique = false)
public Long rowNum;

/** 発生日. */
@Column(name = "occur_date", columnDefinition = "timestamp(6)", nullable = true, unique = false)
public Timestamp occurDate;

/** 摘要. */
@Column(name = "summary", columnDefinition = "varchar2(4000)", nullable = true, unique = false)
public String summary;

/** 経費区分コード. */
@Column(name = "exp_type_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String expTypeCd;

/** 勘定科目コード. */
@Column(name = "account_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String accountCd;

/** 税区分コード. */
@Column(name = "tax_type_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String taxTypeCd;

/** 税率. */
@Column(name = "tax_rate", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal taxRate;

/** 数量. */
@Column(name = "quantity", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal quantity;

/** 取引通貨コード. */
@Column(name = "trans_currency_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String transCurrencyCd;

/** 取引通貨額. */
@Column(name = "trans_amount", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal transAmount;

/** 会社通貨コード. */
@Column(name = "company_currency_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)

```

```

@Column(name = "company_currency_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String companyCurrencyCd;

/** 会社通貨換算レート。*/
@Column(name = "company_rate", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyRate;

/** 会社通貨額。*/
@Column(name = "company_amount", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyAmount;

/** 会社通貨税抜額。*/
@Column(name = "company_amount_exc_tax", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyAmountExcTax;

/** 会社通貨税額。*/
@Column(name = "company_amount_tax", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyAmountTax;

/** 会社通貨額合計。*/
@Column(name = "company_amount_sum", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyAmountSum;

/** 支払分類コード。*/
@Column(name = "pay_type_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String payTypeCd;

/** 支払方法コード。*/
@Column(name = "pay_method_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String payMethodCd;

/** 支払先コード。*/
@Column(name = "payee_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String payeeCd;

/** 支払先名。*/
@Column(name = "payee_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String payeeName;

/** 組織セットコード。*/
@Column(name = "department_set_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String departmentSetCd;

/** 組織コード。*/
@Column(name = "department_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String departmentCd;

/** プロジェクトコード。*/
@Column(name = "project_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String projectCd;

/** 連携区分。*/
@Column(name = "relation_type", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String relationType;

/** 連携id。*/
@Column(name = "relation_id", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String relationId;

/** 連携内容。*/
@Lob
@Column(name = "relation_data", columnDefinition = "clob", nullable = true, unique = false)
public String relationData;

/** 予備項目a。*/
@Column(name = "extension_a", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionA;

/** 予備項目b。*/
@Column(name = "extension_b", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionB;

/** 予備項目c。*/
@Column(name = "extension_c", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionC;

/** 予備項目d。*/
@Column(name = "extension_d", columnDefinition = "varchar2(1000)", nullable = true, unique = false)

```

```

public String extensionD;

/** 予備項目e. */
@Column(name = "extension_e", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionE;

/** 予備項目f. */
@Column(name = "extension_f", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionF;

/** 予備項目g. */
@Column(name = "extension_g", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionG;

/** 予備項目h. */
@Column(name = "extension_h", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionH;

/** 予備項目i. */
@Column(name = "extension_i", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionI;

/** 予備項目j. */
@Column(name = "extension_j", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionJ;
}

```

- **Ka30tExpenseDetail.java**

%ModuleProject%/src/main/generated/jp/co/slcs/kaiden2/product\_workflow/foundation/model/entity/Ka30tExpenseDetail.java

```

package jp.co.slcs.kaiden2.product_workflow.foundation.model.entity;

import java.io.Serializable;
import java.math.BigDecimal;
import java.sql.Timestamp;
import javax.annotation.Generated;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.Lob;
import javax.persistence.Table;
import jp.co.slcs.kaiden2.base.foundation.model.entity.GenerateEntity;

/**
 * (過去案件) 経費明細テーブル.
 *
 * @author Sumitomo Life Information Systems Co.,Ltd.
 */
@Entity
@Table(name = "ka30t_expense_detail")
@Generated(value = {"S2JDBC-Gen 2.4.46", "org.seasar.extension.jdbc.gen.internal.model.EntityModelFactoryImpl"})
public final class Ka30tExpenseDetail extends GenerateEntity implements Serializable {

    /** シリアルバージョンUID. */
    private static final long serialVersionUID = 1L;

    /** システム案件id. */
    @Id
    @Column(name = "system_matter_id", columnDefinition = "varchar2(20)", nullable = false, unique = false)
    public String systemMatterId;

    /** ユーザデータid. */
    @Id
    @Column(name = "user_data_id", columnDefinition = "varchar2(20)", nullable = false, unique = false)
    public String userDataId;

    /** ガジェットクラス. */
    @Id
    @Column(name = "gadget_class", columnDefinition = "varchar2(100)", nullable = false, unique = false)
    public String gadgetClass;

    /** ガジェットインスタンス. */
    @Id
    @Column(name = "gadget_instance", columnDefinition = "number(18,0)", nullable = false, unique = false)
    public Long gadgetInstance;

    /** タブルid. */

```

```

@Id
@Column(name = "tuple_id", columnDefinition = "number(18,0)", nullable = false, unique = false)
public Long tupleId;

/** ロケールid. */
@Id
@Column(name = "locale_id", columnDefinition = "varchar2(50)", nullable = false, unique = false)
public String localeId;

/** アーカイブ年月. */
@Column(name = "archive_month", columnDefinition = "varchar2(6)", nullable = false, unique = false)
public String archiveMonth;

/** ガジェットid. */
@Column(name = "gadget_id", columnDefinition = "varchar2(220)", nullable = false, unique = false)
public String gadgetId;

/** ガジェットバリエーション. */
@Column(name = "gadget_variation", columnDefinition = "varchar2(100)", nullable = false, unique = false)
public String gadgetVariation;

/** 行番号. */
@Column(name = "row_number", columnDefinition = "number(18,0)", nullable = false, unique = false)
public Long rowNumber;

/** 発生日. */
@Column(name = "occur_date", columnDefinition = "timestamp(6)", nullable = true, unique = false)
public Timestamp occurDate;

/** 摘要. */
@Column(name = "summary", columnDefinition = "varchar2(4000)", nullable = true, unique = false)
public String summary;

/** 経費区分コード. */
@Column(name = "exp_type_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String expTypeCd;

/** 勘定科目コード. */
@Column(name = "account_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String accountCd;

/** 税区分コード. */
@Column(name = "tax_type_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String taxTypeCd;

/** 税率. */
@Column(name = "tax_rate", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal taxRate;

/** 数量. */
@Column(name = "quantity", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal quantity;

/** 取引通貨コード. */
@Column(name = "trans_currency_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String transCurrencyCd;

/** 取引通貨額. */
@Column(name = "trans_amount", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal transAmount;

/** 会社通貨コード. */
@Column(name = "company_currency_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String companyCurrencyCd;

/** 会社通貨換算レート. */
@Column(name = "company_rate", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyRate;

/** 会社通貨額. */
@Column(name = "company_amount", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyAmount;

/** 会社通貨税抜額. */
@Column(name = "company_amount_exc_tax", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyAmountExcTax;

/** 会社通貨税額. */

```

```
/* 会社通貨名. */
```

```
@Column(name = "company_amount_tax", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyAmountTax;
```

```
/* 会社通貨額合計. */
```

```
@Column(name = "company_amount_sum", columnDefinition = "number(28,10)", nullable = true, unique = false)
public BigDecimal companyAmountSum;
```

```
/* 支払分類コード. */
```

```
@Column(name = "pay_type_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String payTypeCd;
```

```
/* 支払方法コード. */
```

```
@Column(name = "pay_method_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String payMethodCd;
```

```
/* 支払先コード. */
```

```
@Column(name = "payee_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String payeeCd;
```

```
/* 支払先名. */
```

```
@Column(name = "payee_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String payeeName;
```

```
/* 組織セットコード. */
```

```
@Column(name = "department_set_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String departmentSetCd;
```

```
/* 組織コード. */
```

```
@Column(name = "department_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String departmentCd;
```

```
/* プロジェクトコード. */
```

```
@Column(name = "project_cd", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String projectCd;
```

```
/* 経費区分名. */
```

```
@Column(name = "exp_type_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String expTypeName;
```

```
/* 勘定科目名. */
```

```
@Column(name = "account_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String accountName;
```

```
/* 税区分名. */
```

```
@Column(name = "tax_type_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String taxTypeName;
```

```
/* 取引通貨名. */
```

```
@Column(name = "trans_currency_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String transCurrencyName;
```

```
/* 会社通貨名. */
```

```
@Column(name = "company_currency_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String companyCurrencyName;
```

```
/* 支払分類名. */
```

```
@Column(name = "pay_type_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String payTypeName;
```

```
/* 立替フラグ. */
```

```
@Column(name = "payment_flag", columnDefinition = "varchar2(1)", nullable = true, unique = false)
public String paymentFlag;
```

```
/* 支払方法名. */
```

```
@Column(name = "pay_method_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String payMethodName;
```

```
/* 組織名. */
```

```
@Column(name = "department_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String departmentName;
```

```
/* プロジェクト名. */
```

```
@Column(name = "project_name", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String projectName;
```

```
/* 連携区分. */
```

```
@Column(name = "relation_type", columnDefinition = "varchar2(100)", nullable = true, unique = false)
```

```

public String relationType;

/** 連携id. */
@Column(name = "relation_id", columnDefinition = "varchar2(100)", nullable = true, unique = false)
public String relationId;

/** 連携内容. */
@Lob
@Column(name = "relation_data", columnDefinition = "clob", nullable = true, unique = false)
public String relationData;

/** 予備項目a. */
@Column(name = "extension_a", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionA;

/** 予備項目b. */
@Column(name = "extension_b", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionB;

/** 予備項目c. */
@Column(name = "extension_c", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionC;

/** 予備項目d. */
@Column(name = "extension_d", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionD;

/** 予備項目e. */
@Column(name = "extension_e", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionE;

/** 予備項目f. */
@Column(name = "extension_f", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionF;

/** 予備項目g. */
@Column(name = "extension_g", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionG;

/** 予備項目h. */
@Column(name = "extension_h", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionH;

/** 予備項目i. */
@Column(name = "extension_i", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionI;

/** 予備項目j. */
@Column(name = "extension_j", columnDefinition = "varchar2(1000)", nullable = true, unique = false)
public String extensionJ;
}

```

- **K30tExpenseDetailsService.java**

%ModuleProject%/src/main/generated/jp/co/slcs/kaiden2/product\_workflow/foundation/model/service/K30tExpenseDetailsService.java

```

package jp.co.slcs.kaiden2.product_workflow.foundation.model.service;

import javax.annotation.Generated;
import jp.co.slcs.kaiden2.base.foundation.model.service.GenerateService;
import jp.co.slcs.kaiden2.product_workflow.foundation.model.entity.K30tExpenseDetail;

/**
 * {@link K30tExpenseDetail} のサービスクラス.
 *
 * @author Sumitomo Life Information Systems Co.,Ltd.
 */
@Generated(value = {"S2JDBC-Gen 2.4.46", "org.seasar.extension.jdbc.gen.internal.model.ServiceModelFactoryImpl"})
public final class K30tExpenseDetailsService extends GenerateService<K30tExpenseDetail> {
    //
}

```

- **Ka30tExpenseDetailsService.java**

%ModuleProject%/src/main/generated/jp/co/slcs/kaiden2/product\_workflow/foundation/model/service/Ka30tExpenseDetailsService.java

```

package jp.co.slcs.kaiden2.product_workflow.foundation.model.service;

import javax.annotation.Generated;
import jp.co.slcs.kaiden2.base.foundation.model.service.GenerateService;
import jp.co.slcs.kaiden2.product_workflow.foundation.model.entity.Ka30tExpenseDetail;

/**
 * {@link Ka30tExpenseDetail} のサービスクラス.
 *
 * @author Sumitomo Life Information Systems Co.,Ltd.
 */
@Generated(value = {"S2JDBC-Gen 2.4.46", "org.seasar.extension.jdbc.gen.internal.model.ServiceModelFactoryImpl"})
public final class Ka30tExpenseDetailService extends GenerateService<Ka30tExpenseDetail> {
    //
}

```

- **kaiden\_product\_workflow.dicon**

%ModuleProject%/src/main/resources/kaiden\_product\_workflow.dicon

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE components PUBLIC "-//SEASAR//DTD S2Container 2.4//EN"
    "https://www.seasar.org/dtd/components24.dtd">

<components>

    <!-- 略 -->

    <component name="k30tExpenseDetailService"
class="jp.co.slcs.kaiden2.product_workflow.foundation.model.service.K30tExpenseDetailService" instance="prototype">
    </component>
    <component name="ka30tExpenseDetailService"
class="jp.co.slcs.kaiden2.product_workflow.foundation.model.service.Ka30tExpenseDetailService" instance="prototype">
    </component>
</components>

```

## GadgetBuilder定義作成

続いて、GadgetBuilder定義を作成します。

### GadgetBuilder定義ファイル作成

定義用のファイルを作成します。

ファイル名は必ず「%ガジェットクラス%\_%ガジェットバリエーション%.xml (expenseDetail\_v01.xml)」とします。

#### 配置先

%ModuleProject%/src/main/storage/public/kaiden/generic/gadget/expenseDetail\_v01.xml

#### 内容

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">
  <gadgetClass>expenseDetail</gadgetClass>
  <gadgetVariation>v01</gadgetVariation>
  <jspPath>/WEB-INF/view/kaiden2/extension_imw/generic/gadget/generic_v01.jsp</jspPath>
  <javaPath>jp.co.slcs.kaiden2.extension_imw.feature.logic.kaiden.gadget.builder.GadgetBuilderLogic</javaPath>
  <sortKey>51001</sortKey>
  <surfaceKey>chapterTitle</surfaceKey>
  <descriptions>
    <description localeId="en" gadgetName="Expenses Particulars" description="Enter Expenses Particulars.&#xD;Only Japanese Yen is available."/>
    <description localeId="ja" gadgetName="経費明細" description="経費明細の入力を行います。&#xD;日本円のみ利用可能です。"/>
    <description localeId="zh_CN" gadgetName="明細" description="明細の入力。&#xD;只可使用日元。"/>
  </descriptions>
  <blocks>
    <block id="expenseDetail" type="DETAIL" surfaceKey="chapterTitle">
      <service basic="k30tExpenseDetailService" archive="ka30tExpenseDetailService">
        <sqls>
          <sql id="init" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="copy" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="merge" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="archive" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
          <sql id="select" path="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql"/>
        </sqls>
      </service>
    </block>
  </blocks>
</gadgetBuilderSettings>

```

```

</sqis>
</service>
<groups>
  <group id="rowNumber" surfaceKey="rowNumber" requiredType="ALWAYS_ANY" inputType="LABEL">
    <args>
      <arg key="style_pc" value="text-align:center;"/>
    </args>
    <fields>
      <field id="rowNumber" surfaceKey="rowNumber" requiredType="INHERIT" inputType="INHERIT" type="PANEL">
        <!--
          <args>
            <arg key="header" value="true"/>
            <arg key="headerAllowAdd" value="true"/>
            <arg key="headerAllowDel" value="true"/>
            <arg key="detail" value="true"/>
            <arg key="detailAllowAdd" value="true"/>
            <arg key="detailAllowDel" value="true"/>
          </args>
        -->
      </field>
      <field id="imwApplyBaseDate" surfaceKey="imwApplyBaseDate" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DTO"/>
      <field id="searchCriteriaCompany" surfaceKey="searchCriteriaCompany" requiredType="ALWAYS_ANY" inputType="INHERIT"
type="DTO"/>
      <field id="payMethodCd" surfaceKey="payMethod" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
    </fields>
  </group>
  <group id="occurDate" surfaceKey="occurDate" requiredType="ALWAYS_REQ" inputType="CONFIG">
    <fields>
      <field id="occurDate" surfaceKey="occurDate" requiredType="INHERIT" inputType="INHERIT" type="TEXT_DATE">
        <args>
          <arg key="autoFormat" value="true"/>
        </args>
        <validates>
          <validate event="default" config="{ 'id':'date' }"/>
        </validates>
      </field>
    </fields>
  </group>
  <group id="summary" surfaceKey="summary" requiredType="SETTING_REQ" inputType="CONFIG">
    <fields>
      <field id="summary" surfaceKey="summary" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
        <args>
          <arg key="maxLength" value="250"/>
          <arg key="style_pc" value="width:98%"/>
        </args>
        <validates>
          <validate event="default" config="{ 'id':'maxLength', 'args':'250' }"/>
        </validates>
      </field>
    </fields>
  </group>
  <group id="expType" surfaceKey="expType" requiredType="ALWAYS_REQ" inputType="CONFIG">
    <fields>
      <field id="expTypeCd" surfaceKey="expType" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
        <args>
          <arg key="masterId" value="exp_type"/>
          <arg key="extensionId" value="expenseDetail"/>
          <arg key="searchType" value="search"/>
          <arg key="viewName" value="expTypeName"/>
          <arg key="criteria" value="
{ 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany', 'expTypeCd':'expTypeCd', 'includeBlank':'false' }"/>
        </args>
        <arg key="resultChain" value="
{ 'expTypeCd':'expTypeCd', 'expTypeName':'expTypeName', 'accountCd':'accountCd', 'accountName':'accountName', 'taxTypeCd':'taxTypeCd', 'taxT

          <arg key="resultValue" value="expTypeName"/>
          <arg key="primaryTarget" value="expTypeCd"/>
          <arg key="tmtableId" value="expenseDetail"/>
          <arg key="refreshTarget" value="imwApplyBaseDate,searchCriteriaCompany"/>
          <arg key="style_pc" value="display:inline-block; width:35%;"/>
        </args>
        <validates>
          <validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'exp_type',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'expTypeCd':'expTypeCd' } }"/>
        </validates>
      </field>
      <field id="accountCd" surfaceKey="account" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN">

```

```

<validates>
  <validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'tax_type',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'taxTypeCd':'taxTypeCd'}}"/>
</validates>
</field>
<field id="accountName" surfaceKey="account" requiredType="ALWAYS_ANY" inputType="LABEL" type="TEXT">
  <args>
    <arg key="style_pc" value="display:inline-block; width:30%; margin-left:3px; text-overflow:ellipsis; white-space:nowrap;
overflow:hidden; vertical-align:middle;"/>
  </args>
</field>
<field id="taxTypeCd" surfaceKey="taxType" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN">
  <validates>
    <validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'tax_type',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'taxTypeCd':'taxTypeCd'}}"/>
  </validates>
</field>
<field id="taxTypeName" surfaceKey="taxType" requiredType="ALWAYS_ANY" inputType="LABEL" type="TEXT">
  <args>
    <arg key="style_pc" value="display:inline-block; width:25%; margin-left:3px; text-overflow:ellipsis; white-space:nowrap;
overflow:hidden; vertical-align:middle;"/>
  </args>
</field>
<field id="taxRate" surfaceKey="taxRate" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
</fields>
</group>
<group id="department" surfaceKey="department" requiredType="ALWAYS_ANY" inputType="CONFIG">
  <fields>
    <field id="departmentSetCd" surfaceKey="departmentSet" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
    <field id="departmentCd" surfaceKey="department" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
      <args>
        <arg key="masterId" value="department"/>
        <arg key="extensionId" value="expenseDetail"/>
        <arg key="searchType" value="search"/>
        <arg key="viewName" value="departmentName"/>
        <arg key="criteria" value=
{ 'searchCriteriaDate':'imwApplyBaseDate','searchCriteriaCompany':'searchCriteriaCompany','departmentSetCd':'searchCriteriaCompany','depart

        <arg key="resultChain" value="
{ 'departmentCd':'departmentCd','departmentName':'departmentName','departmentSetCd':'departmentSetCd'}"/>
        <arg key="resultValue" value="departmentName"/>
        <arg key="primaryTarget" value="departmentSetCd,departmentCd"/>
        <arg key="tmtableId" value="expenseDetail"/>
        <arg key="refreshTarget" value="imwApplyBaseDate,searchCriteriaCompany,searchCriteriaCompany"/>
        <arg key="style_pc" value="width:85%;"/>
      </args>
    </validates>
    <validate event="default" target="ALWAYS" config="{ 'id':'requiredAny', 'args':['departmentCd','projectCd'] }"/>
    <validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'department',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'departmentSetCd':'searchCriteriaCompany','departmentCd':'departmentCd'}}"/>
  </validates>
</field>
</fields>
</group>
<group id="project" surfaceKey="project" requiredType="ALWAYS_ANY" inputType="CONFIG">
  <fields>
    <field id="projectCd" surfaceKey="project" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
      <args>
        <arg key="masterId" value="project_workflow"/>
        <arg key="extensionId" value="expenseDetail"/>
        <arg key="searchType" value="search"/>
        <arg key="viewName" value="projectName"/>
        <arg key="criteria" value=
{ 'searchCriteriaDate':'imwApplyBaseDate','searchCriteriaCompany':'searchCriteriaCompany','projectCd':'projectCd','includeBlank':'true'}"/>
        <arg key="resultChain" value="{ 'projectCd':'projectCd','projectName':'projectName'}"/>
        <arg key="resultValue" value="projectName"/>
        <arg key="primaryTarget" value="projectCd"/>
        <arg key="tmtableId" value="expenseDetail"/>
        <arg key="refreshTarget" value="imwApplyBaseDate,searchCriteriaCompany"/>
        <arg key="style_pc" value="width:85%;"/>
      </args>
    </validates>
    <validate event="default" target="ALWAYS" config="{ 'id':'requiredAny', 'args':['departmentCd','projectCd'] }"/>
    <validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'project_workflow',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',

```

```

'projectCd':'projectCd'}}"/>
  </validates>
</field>
</fields>
</group>
<group id="payType" surfaceKey="payType" requiredType="ALWAYS_REQ" inputType="CONFIG">
  <fields>
    <field id="payTypeCd" surfaceKey="payType" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
      <args>
        <arg key="masterId" value="pay_type"/>
        <arg key="extensionId" value="expenseDetail"/>
        <arg key="searchType" value="select"/>
        <arg key="viewName" value="payTypeName"/>
        <arg key="criteria" value=""
{searchCriteriaDate':'imwApplyBaseDate','searchCriteriaCompany':'searchCriteriaCompany','payTypeCd':'payTypeCd','includeBlank':'false'}/>
        <arg key="resultChain" value="{ 'payTypeCd':'payTypeCd','payTypeName':'payTypeName','paymentFlag':'paymentFlag'}/>
        <arg key="resultValue" value="payTypeName"/>
        <arg key="primaryTarget" value="payTypeCd"/>
        <arg key="tmtableId" value="expenseDetail"/>
        <arg key="refreshTarget" value="imwApplyBaseDate,searchCriteriaCompany"/>
<!--
        <arg key="singleList" value="true"/> -->
        <arg key="style_pc" value="width:98%;"/>
      </args>
    </validates>
    <validate event="default" target="SERVER" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'pay_type',
'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany',
'payTypeCd':'payTypeCd'}}"/>
  </validates>
</field>
  <field id="paymentFlag" surfaceKey="paymentFlag" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
</fields>
</group>
<group id="payee" surfaceKey="payee" requiredType="SETTING_ANY" inputType="CONFIG">
  <fields>
    <field id="payeeName" surfaceKey="payee" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
      <args>
        <arg key="maxLength" value="250"/>
        <arg key="style_pc" value="width:98%;"/>
      </args>
    </validates>
    <validate event="default" config="{ 'id':'maxLength', 'args':{'250'}}"/>
  </validates>
</field>
</fields>
</group>
<group id="transAmount" surfaceKey="transAmount" requiredType="ALWAYS_REQ" inputType="CONFIG">
  <args>
    <arg key="style_pc" value="text-align:right;"/>
  </args>
  <fields>
    <field id="transAmount" surfaceKey="transAmount" requiredType="INHERIT" inputType="INHERIT" type="TEXT_NUMERIC">
      <args>
        <arg key="style_pc" value="width:98%;"/>
        <arg key="amount" value="true"/>
        <arg key="event" value="change"/>
      </args>
    </validates>
    <validate event="default" config="{ 'id':'currencyAmnt2', 'args':['','searchCriteriaCompany']}/>
    <validate event="default" config="{ 'id':'amountMin', 'args':{'0'}}"/>
  </validates>
</field>
  <field id="transCurrencyCd" surfaceKey="transCurrency" requiredType="ALWAYS_ANY" inputType="INHERIT"
type="TEXT_HIDDEN"/>
  <field id="companyRate" surfaceKey="companyRate" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
  <field id="companyAmount" surfaceKey="companyAmount" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
  <field id="companyCurrencyCd" surfaceKey="companyCurrency" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DTO"/>
  <field id="quantity" surfaceKey="quantity" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN"/>
  <field id="companyAmountSum" surfaceKey="companyAmountSum" requiredType="INHERIT" inputType="INHERIT"
type="TEXT_HIDDEN"/>
  <field id="companyAmountExcTax" surfaceKey="companyAmountExcTax" requiredType="INHERIT" inputType="INHERIT"
type="TEXT_HIDDEN"/>
  <field id="companyAmountTax" surfaceKey="companyAmountTax" requiredType="INHERIT" inputType="INHERIT"
type="TEXT_HIDDEN"/>
</fields>
</group>
</groups>

```

```

<table rowNumber1 target="rowNumber" type="HORIZONTAL">
  <colgroups>
    <colgroup style="width:50px; overflow:hidden"/>
    <colgroup style="overflow:hidden"/>
    <colgroup style="width:23%; overflow:hidden"/>
    <colgroup style="width:23%; overflow:hidden"/>
    <colgroup style="width:23%; overflow:hidden"/>
  </colgroups>
  <rows>
    <row>
      <cols>
        <col groupId="rowNumber" rowspan="3"/>
        <col groupId="occurDate"/>
        <col groupId="summary" colspan="3"/>
      </cols>
    </row>
    <row>
      <cols>
        <col groupId="expType" colspan="2"/>
        <col groupId="department"/>
        <col groupId="project"/>
      </cols>
    </row>
    <row>
      <cols>
        <col groupId="payType"/>
        <col groupId="payee" colspan="2"/>
        <col groupId="transAmount"/>
      </cols>
    </row>
  </rows>
</table>
<completions>
  <completion name="departmentSetCd" position="1" type="claimValueOnInit" args="{ 'claim': 'claimDefaultDepartmentSetCd',
target:'1'}" target="CLIENT" mode="EMPTY"/>
  <completion name="departmentCd" position="1" type="claimValueOnInit" args="{ 'claim': 'claimDefaultDepartmentCd',
target:'1'}" target="CLIENT" mode="EMPTY"/>
  <completion name="projectCd" position="1" type="claimValueOnInit" args="{ 'claim': 'claimDefaultProjectCd', 'target': '1'}"
target="CLIENT" mode="EMPTY"/>
  <completion name="payMethodCd" position="1" type="claimValue" args="{ 'claim': 'claimPayMethod'}"
target="ALWAYS" mode="ALWAYS"/>
  <completion name="transCurrencyCd" position="2" type="variable" args="{ 'field': 'companyCurrencyCd'}"
target="ALWAYS" mode="ALWAYS"/>
  <completion name="companyRate" position="3" type="fixed" args="{ 'value': '1'}"
target="ALWAYS" mode="EMPTY"/>
  <completion name="companyAmount" position="4" type="variable" args="{ 'field': 'transAmount'}"
target="ALWAYS" mode="ALWAYS"/>
  <completion name="quantity" position="5" type="fixed" args="{ 'value': '1'}"
target="ALWAYS" mode="EMPTY"/>
  <completion name="companyAmountSum" position="6" type="calcMul" args="
{'field1': 'companyAmount', 'field2': 'quantity', 'round': '4', 'digits': '0'}" target="ALWAYS" mode="ALWAYS"/>
  <completion name="companyAmountTax" position="7" type="calcTax" args="
{'amount': 'companyAmountSum', 'taxRate': 'taxRate', 'round': '4', 'digits': '0'}" target="ALWAYS" mode="ALWAYS"/>
  <completion name="companyAmountExcTax" position="8" type="calcSub" args="
{'field1': 'companyAmountSum', 'field2': 'companyAmountTax', 'round': '4', 'digits': '0'}" target="ALWAYS" mode="ALWAYS"/>
</completions>
<claims>
  <claim name="claimExpenseAmount" type="claimAmountSummary" args="{ 'amount': 'companyAmountSum'}"/>
  <claim name="claimTemporaryPayAmount" type="claimAmountSummary" args="
{'amount': 'companyAmountSum', 'paymentFlag': 'paymentFlag', 'payment': '1'}"/>
  <claim name="claimCompanyPayAmount" type="claimAmountSummary" args="
{'amount': 'companyAmountSum', 'paymentFlag': 'paymentFlag', 'payment': '0'}"/>
  <claim name="claimDateDetail" type="claimColumnValuesList" args="{ 'field1': 'occurDate', 'field2': 'occurDate'}"/>
  <claim name="claimExpenseDetail" type="claimColumnValuesList" args="{ 'field1': 'companyAmountSum'}"/>
  <claim name="claimExpenseCount" type="claimRowCount" args=""/>
  <claim name="loadData4Settle" type="claimSearchData" args="{ 'systemMatterId': 'imwSystemMatterId',
'userDataId': 'imwUserDataId', 'gadgetClass': 'gadgetClass'}" target="CLIENT"/>
</claims>
<events>
  <event target="matter_end" name="journalSourceDetailEntryForSqlEvent">
    <args>
      <arg key="sqlPath" value="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/journal/ExpenseDetail.sql"/>
    </args>
  </event>
</events>
<includes>
  <include path="/WEB-INF/view/kaiden2/product_workflow/generic/gadget/include/defaultDepartmentProject.jsp" />
</includes>

```

```

</block>
</blocks>
<surfaces>
  <surface surfaceKey="chapterTitle">
    <name localeId="en" surface="Expenses Particulars"/>
    <name localeId="ja" surface="経費明細"/>
    <name localeId="zh_CN" surface="□□明□"/>
  </surface>

  <surface surfaceKey="rowNumber">
    <name localeId="en" surface="No"/>
    <name localeId="ja" surface="No"/>
    <name localeId="zh_CN" surface="No"/>
  </surface>

  <surface surfaceKey="imwApplyBaseDate">
    <name localeId="en" surface="Apply Reference Date"/>
    <name localeId="ja" surface="申請基準日"/>
    <name localeId="zh_CN" surface="申□基準日"/>
  </surface>

  <surface surfaceKey="searchCriteriaCompany">
    <name localeId="en" surface="Search Reference Company"/>
    <name localeId="ja" surface="検索基準会社"/>
    <name localeId="zh_CN" surface="□索基准公司"/>
  </surface>

  <surface surfaceKey="payMethod">
    <name localeId="en" surface="Payment Method"/>
    <name localeId="ja" surface="支払方法"/>
    <name localeId="zh_CN" surface="支付方法"/>
  </surface>

  <surface surfaceKey="occurDate">
    <name localeId="en" surface="Occurrence Date"/>
    <name localeId="ja" surface="発生日"/>
    <name localeId="zh_CN" surface="□生日"/>
  </surface>

  <surface surfaceKey="summary">
    <name localeId="en" surface="Description of Expense"/>
    <name localeId="ja" surface="経費内容"/>
    <name localeId="zh_CN" surface="□□内容"/>
  </surface>

  <surface surfaceKey="expType">
    <name localeId="en" surface="Expense Type"/>
    <name localeId="ja" surface="経費区分"/>
    <name localeId="zh_CN" surface="□□分□"/>
  </surface>

  <surface surfaceKey="account">
    <name localeId="en" surface="Account"/>
    <name localeId="ja" surface="勘定科目"/>
    <name localeId="zh_CN" surface="□□科目"/>
  </surface>

  <surface surfaceKey="taxType">
    <name localeId="en" surface="Tax Category"/>
    <name localeId="ja" surface="税区分"/>
    <name localeId="zh_CN" surface="税分□"/>
  </surface>

  <surface surfaceKey="taxRate">
    <name localeId="en" surface="Tax Rate"/>
    <name localeId="ja" surface="税率"/>
    <name localeId="zh_CN" surface="税率"/>
  </surface>

  <surface surfaceKey="departmentSet">
    <name localeId="en" surface="Organization Set"/>
    <name localeId="ja" surface="組織セット"/>
    <name localeId="zh_CN" surface="□□集合"/>
  </surface>

  <surface surfaceKey="department">
    <name localeId="en" surface="Organization"/>
    <name localeId="ja" surface="組織"/>
    <name localeId="zh_CN" surface="□□"/>
  </surface>

  <surface surfaceKey="project">
    <name localeId="en" surface="Project"/>
    <name localeId="ja" surface="プロジェクト"/>
    <name localeId="zh_CN" surface="□目"/>
  </surface>

  <surface surfaceKey="payType">
    <name localeId="en" surface="Payment Classification"/>
    <name localeId="ja" surface="支払分類"/>
    <name localeId="zh_CN" surface="支付分□"/>
  </surface>

```

```

</surface>
<surface surfaceKey="paymentFlag">
  <name localeId="en" surface="Reimbursement Flag"/>
  <name localeId="ja" surface="立替フラグ"/>
  <name localeId="zh_CN" surface="□付□志"/>
</surface>
<surface surfaceKey="payee">
  <name localeId="en" surface="Payee"/>
  <name localeId="ja" surface="支払先"/>
  <name localeId="zh_CN" surface="收款方"/>
</surface>
<surface surfaceKey="transAmount">
  <name localeId="en" surface="Amount"/>
  <name localeId="ja" surface="金額"/>
  <name localeId="zh_CN" surface="金□"/>
</surface>
<surface surfaceKey="transCurrency">
  <name localeId="en" surface="Currency"/>
  <name localeId="ja" surface="通貨"/>
  <name localeId="zh_CN" surface="□□"/>
</surface>
<surface surfaceKey="companyRate">
  <name localeId="en" surface="rate"/>
  <name localeId="ja" surface="レート"/>
  <name localeId="zh_CN" surface="□率"/>
</surface>
<surface surfaceKey="companyAmount">
  <name localeId="en" surface="Converted"/>
  <name localeId="ja" surface="換算額"/>
  <name localeId="zh_CN" surface="□算□"/>
</surface>
<surface surfaceKey="companyCurrency">
  <name localeId="en" surface="Company Currency"/>
  <name localeId="ja" surface="会社通貨"/>
  <name localeId="zh_CN" surface="公司□□"/>
</surface>
<surface surfaceKey="quantity">
  <name localeId="en" surface="Quantity"/>
  <name localeId="ja" surface="数量"/>
  <name localeId="zh_CN" surface="数量"/>
</surface>
<surface surfaceKey="companyAmountSum">
  <name localeId="en" surface="Total amount"/>
  <name localeId="ja" surface="合計額"/>
  <name localeId="zh_CN" surface="□量"/>
</surface>
<surface surfaceKey="companyAmountExcTax">
  <name localeId="en" surface="Tax excluded amount"/>
  <name localeId="ja" surface="税抜金額"/>
  <name localeId="zh_CN" surface="税抜金量"/>
</surface>
<surface surfaceKey="companyAmountTax">
  <name localeId="en" surface="Tax included amount"/>
  <name localeId="ja" surface="税込金額"/>
  <name localeId="zh_CN" surface="□含税金□"/>
</surface>
</surfaces>
<tags>
  <tag value="appNormal"/>
  <tag value="appAdvance"/>
  <tag value="appSettle"/>
  <tag value="spSupport"/>
</tags>
</gadgetBuilderSettings>

```

## SQL作成

必要に応じて、GadgetBuilderで使用する検索用のSQLを作成します。

### 内容

```
%ModuleProject%/src/main/resources/META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/ExpenseDetail.sql
```

```

select
target.system_matter_id

```

```

,target.user_data_id
,target.gadget_id
,target.gadget_class
,target.gadget_variation
,target.gadget_instance
,target.tuple_id
,target.row_number
,target.occur_date
,target.summary
,target.exp_type_cd
,exp_type.exp_type_name
,target.account_cd
,account.account_name
,target.tax_type_cd
,tax_type.tax_type_name
,target.tax_rate
,target.quantity
,target.trans_currency_cd
,trans_currency.currency_name as trans_currency_name
,target.trans_amount
,target.company_rate
,target.company_currency_cd
,company_currency.currency_name as company_currency_name
,target.company_amount
,target.company_amount_exc_tax
,target.company_amount_tax
,target.company_amount_sum
,target.pay_type_cd
,pay_type.pay_type_name
,pay_type.payment_flag
,target.pay_method_cd
,pay_method.pay_method_name
,target.payee_cd
,target.payee_name
,target.department_set_cd
,target.department_cd
,department.department_name
,target.project_cd
,project.project_name
/*IF sqlld != "copy"*/
,target.relation_type
,target.relation_id
,target.relation_data
/*END*/
,target.extension_a
,target.extension_b
,target.extension_c
,target.extension_d
,target.extension_e
,target.extension_f
,target.extension_g
,target.extension_h
,target.extension_i
,target.extension_j
,target.entry_ts
,target.entry_user_cd
,target.renew_cnt
,target.renew_ts
,target.renew_user_cd
,info.apply_base_date as imw_apply_base_date
,info.auth_company_cd as search_criteria_company_code
/*localeld*/ as locale_id
from
k30t_expense_detail target
left outer join k20c_matter_info info
  on info.system_matter_id = target.system_matter_id
left outer join k30m_exp_type exp_type
  on exp_type.company_cd      = info.auth_company_cd
  and exp_type.exp_type_cd    = target.exp_type_cd
  and exp_type.locale_id     = /*localeld*/'ja'
  and exp_type.start_date    <= info.apply_base_date
  and exp_type.end_date      > info.apply_base_date
  and exp_type.delete_flag   = '0'
left outer join k30m_account account
  on account.company_cd     = info.auth_company_cd
  and target.account_cd    = account.account_cd
  and account.locale_id    = /*localeld*/'ja'
  and account.start date  <= info.apply base date

```

```

and account.end_date > info.apply_base_date
and account.delete_flag = '0'
left outer join k30m_tax_type tax_type
on tax_type.company_cd = info.auth_company_cd
and target.tax_type_cd = tax_type.tax_type_cd
and tax_type.locale_id = /*localeId*/ja'
and tax_type.start_date <= info.apply_base_date
and tax_type.end_date > info.apply_base_date
and tax_type.delete_flag = '0'
left outer join imm_currency trans_currency
on trans_currency.company_cd = info.auth_company_cd
and trans_currency.currency_cd = target.trans_currency_cd
and trans_currency.locale_id = /*localeId*/ja'
and trans_currency.delete_flag = '0'
left outer join imm_currency company_currency
on company_currency.company_cd = info.auth_company_cd
and company_currency.currency_cd = target.company_currency_cd
and company_currency.locale_id = /*localeId*/ja'
and company_currency.delete_flag = '0'
left outer join k30m_pay_type pay_type
on pay_type.company_cd = info.auth_company_cd
and pay_type.pay_type_cd = target.pay_type_cd
and pay_type.locale_id = /*localeId*/ja'
and pay_type.start_date <= info.apply_base_date
and pay_type.end_date > info.apply_base_date
and pay_type.delete_flag = '0'
left outer join k30m_pay_method pay_method
on pay_method.company_cd = info.auth_company_cd
and pay_method.pay_method_cd = target.pay_method_cd
and pay_method.locale_id = /*localeId*/ja'
and pay_method.start_date <= info.apply_base_date
and pay_method.end_date > info.apply_base_date
and pay_method.delete_flag = '0'
left outer join imm_department department
on department.company_cd = info.auth_company_cd
and department.department_set_cd = target.department_set_cd
and department.department_cd = target.department_cd
and department.locale_id = /*localeId*/ja'
and department.start_date <= info.apply_base_date
and department.end_date > info.apply_base_date
and department.delete_flag = '0'
left outer join k30m_project project
on project.company_cd = info.auth_company_cd
and project.project_cd = target.project_cd
and project.locale_id = /*localeId*/ja'
and project.start_date <= info.apply_base_date
and project.end_date > info.apply_base_date
and project.delete_flag = '0'
/*BEGIN*/
where
/*IF systemMatterId != null*/
target.system_matter_id = /*systemMatterId*/systemMatterId'
/*END*/
/*IF userDataId != null*/
and target.user_data_id = /*userDataId*/userDataId'
/*END*/
/*IF gadgetClass != null*/
and target.gadget_class = /*gadgetClass*/gadgetClass'
/*END*/
/*IF gadgetVariation != null*/
and target.gadget_variation = /*gadgetVariation*/gadgetVariation'
/*END*/
/*END*/
order by target.system_matter_id, target.row_number

```

#### 検索条件

設定したLogicクラスに応じて、検索条件が設定されます。

- **jp.co.slcs.kaiden2.base.feature.logic.kaiden.generic.impl.GadgetLogic**

条件名	条件キー
システム案件ID	systemMatterId
ユーザデータID	userDataId
ガジェットクラス	gadgetClass

ガジェットバリエーション gadgetVariation

ロケールID localeId

### **i** コラム

申請基準日と検索基準会社を取得するために、下記の結合条件を必ず追加してください。

上記SQLから抜粋

```
from
k30t_expense_detail target
left outer join k20c_matter_info info
on info.system_matter_id = target.system_matter_id
```

## 動作確認

これまでに設定した内容を動作確認します。

### 配置先



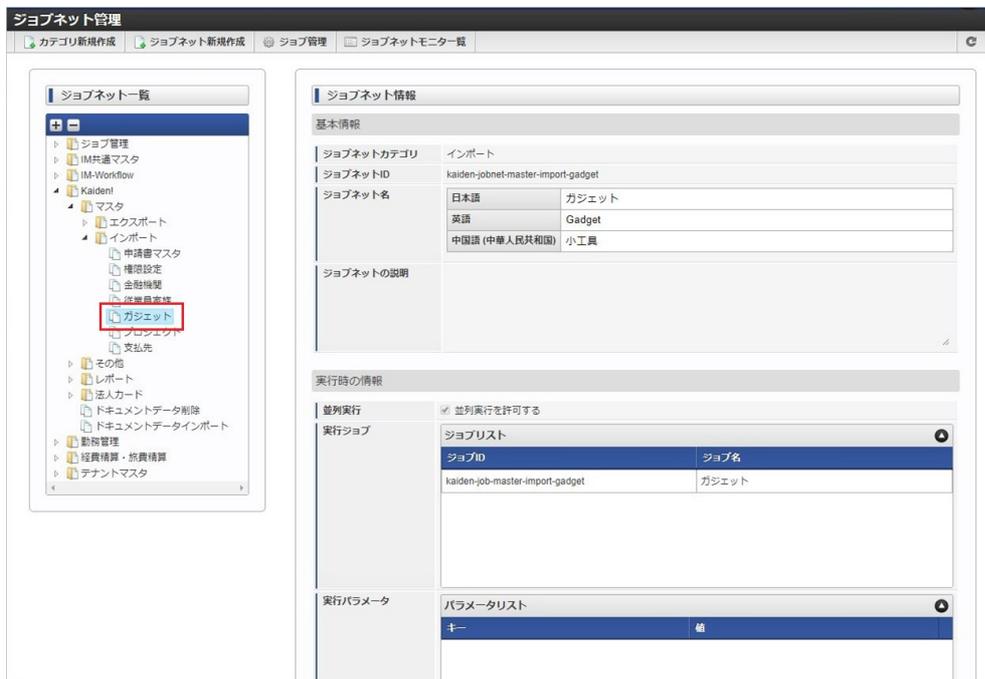
### **i** コラム

配置先には記載していませんが、「[テーブル定義](#)」で定義したテーブルを作成してください。

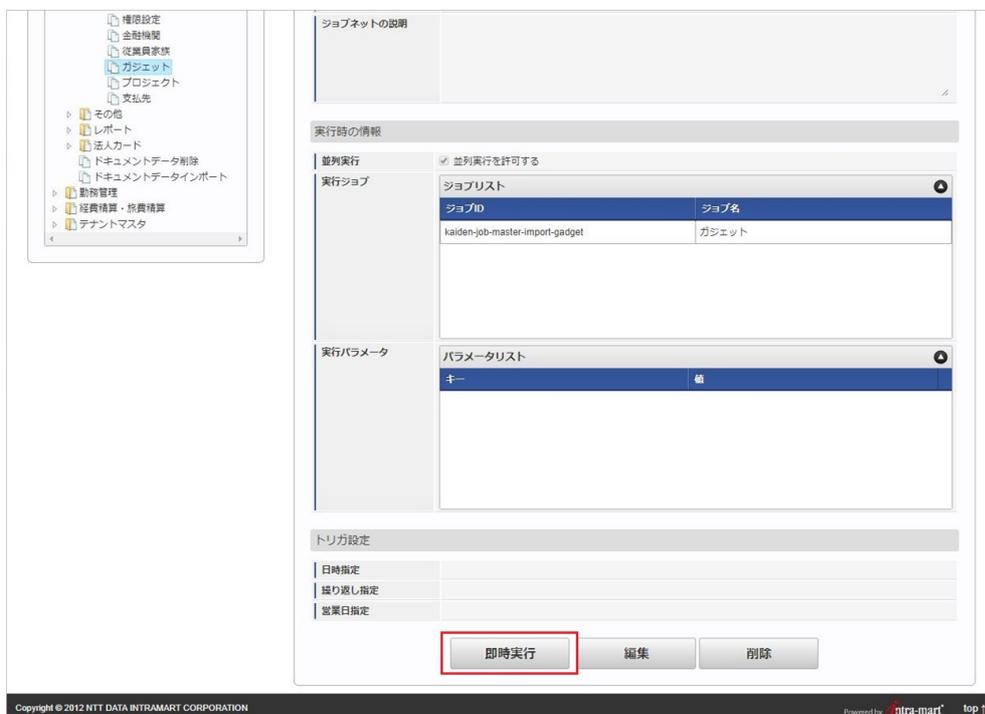
## GadgetBuilder定義インポート

ガジェットのインポートジョブを使用して、GadgetBuilder定義をインポートします。

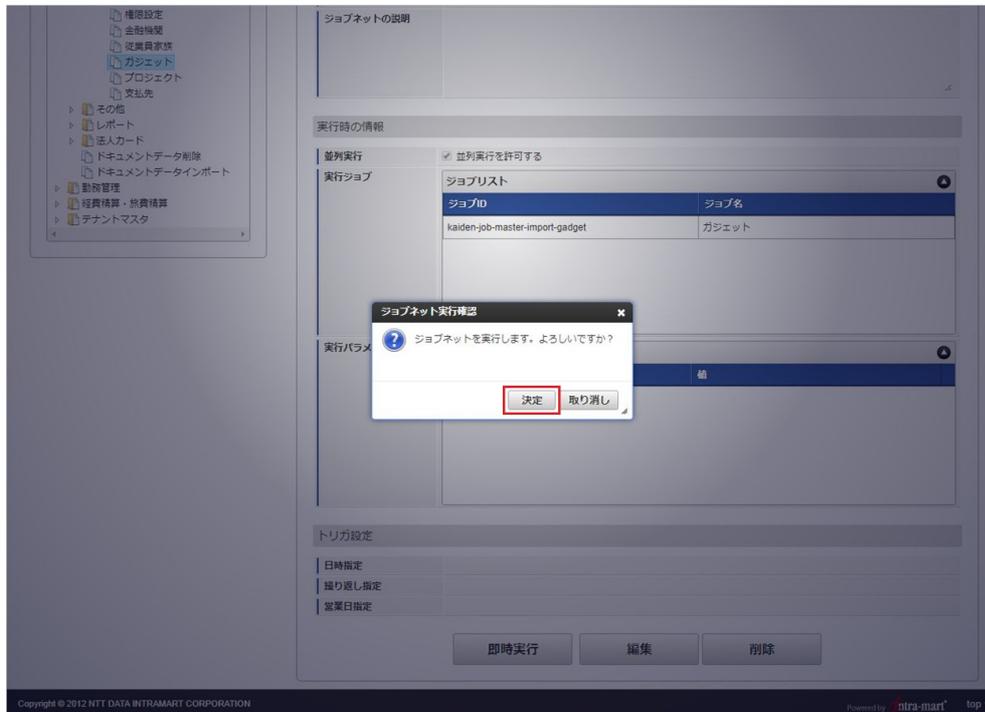
1. 「サイトマップ」→「テナント管理」→「ジョブネット設定」をクリックします。
2. ジョブネット一覧（画面左部）の「ガジェット」をクリックします。



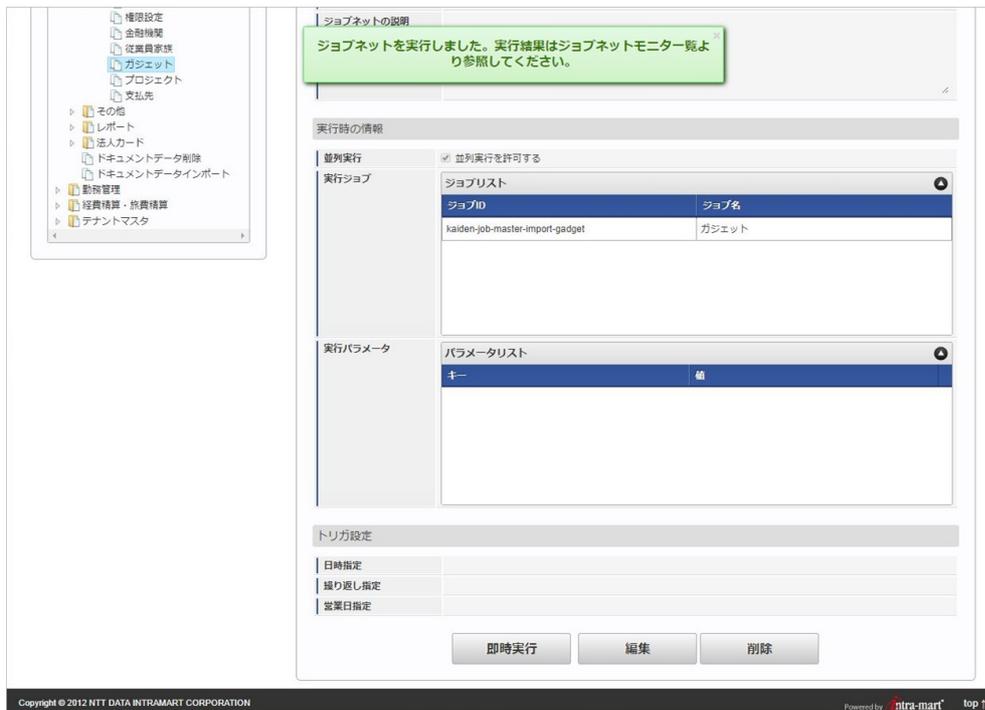
3. 「即時実行」をクリックします。



4. 確認ポップアップの「決定」をクリックします。



5. ジョブネットが実行されました。



### コラム

ガジェットインポートのジョブは、「%パブリックストレージ%/kaiden/generic/gadget」フォルダに配置された GadgetBuilder定義をインポートします。  
システムストレージに配置されたGadgetBuilder定義はインポート対象外です。

### 注意

「%パブリックストレージ%/kaiden/generic/gadget」フォルダに配置されたGadgetBuilder定義(xmlファイル)は全てインポートされます。  
上記のため、ファイル名が異なっても「ガジェットクラス」「ガジェットバリエーション」が同一のGadgetBuilder定義が存在している場合、インポートしたいGadgetBuilder定義を上書きしてしまい想定外の反映がされない場合があります。

### 注意

ガジェットインポート後は、必ずインポートしたガジェットを使用しているコンテンツを申請書マスタから更新してください。  
更新しない場合、申請書を申請する時にエラーが発生します。

## 申請書メンテナンス

続いて申請書メンテナンス機能を使用して、コンテンツを作成します。  
作成方法の詳細は『intra-mart Accel Kaiden! 経費旅費 / 管理者操作ガイド』を参照してください。



## コラム

コンテンツ作成後は、フローを作成してください。

## 申請画面

申請画面を表示し、動作確認を行います。

経費精算申請

← 申請
一時保存
申請書コピー

2013/08/27

申請会社	サンプル会社	申請組織	
申請者(代理申請者)	青柳辰巳		

経費旅費情報

支出目的*	<input style="width: 80%;" type="text"/>
備考	<input style="width: 80%;" type="text"/>

精算情報

案件番号	案件名	仮払残高	精算予定額*
<input type="checkbox"/> 精算対象の事前申請書を追加			
精算手段*	銀行振込		
精算内訳	費用合計	会社負担額合計	精算予定額合計
	0	0	0
			0

経費明細

	発生日*	経費内容*		
<input type="checkbox"/>	経費区分* 勘定科目 税区分	部門	プロジェクト	
	支払分類*	支払先	金額*	

## リファレンス

## フィールドタイプ

GadgetBuilder定義の「[フィールド設定](#)」で設定可能なフィールドタイプについて説明します。

- intra-mart Accel Kaiden!標準のフィールドタイプ
  - テキストボックス [ type : TEXT ]
  - テキストボックス(数値) [ type : TEXT\_NUMERIC ]
  - テキストボックス(コード) [ type : TEXT\_CODE ]
  - テキストボックス(日付) [ type : TEXT\_DATE ]
  - テキストボックス(時間) [ type : TEXT\_TIME ]
  - テキストボックス(パスワード) [ type : TEXT\_PASSWORD ]
  - テキストボックス(Hidden) [ type : TEXT\_HIDDEN ]
  - テキストエリア [ type : TEXTAREA ]
  - チェックボックス [ type : CHECKBOX ]
  - ラジオボタン [ type : RADIO ]
  - サーフィス [ type : SURFACE ]
  - ファイル [ type : FILE ]
  - マスタ検索 [ type : MASTER ]
  - ボタン [ type : BUTTON ]
  - アイコン [ type : ICON ]
  - アンカー [ type : ANCHOR ]
  - 操作パネル [ type : PANEL ]
  - DTO値 [ type : DTO ]
  - 予備タイプA~J [ type : EXTENSION\_A~J ]

## intra-mart Accel Kaiden!標準のフィールドタイプ

本項では、intra-mart Accel Kaiden!で提供しているフィールドタイプと設定方法を説明します。

## テキストボックス [ type : TEXT ]

属性値	設定値
imeMode	IME-MODEを設定します。設定値 : [ on / off / disabled ]
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
maxlength	最大文字数を設定します。
title	ツールチップの文字列を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
disabled	disabled属性を設定します。設定値 : [ true / false ]
placeholder	placeholder属性を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
  <args>
    <arg key="imeMode" value="on"/>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value=""/>
    <arg key="maxlength" value="250"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="readonly" value="false"/>
    <arg key="disabled" value="false"/>
    <arg key="placeholder" value="targetField"/>
    <arg key="event" value="change"/>
  </args>
</field>
```

表示状態

- PC

テキスト

- SP(スマートフォン)

テキスト

テキストボックス(数値) [ type : TEXT\_NUMERIC ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
maxlength	最大文字数を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]
placeholder	placeholder属性を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]
amount	金額のカンマ区切り有無を設定します。設定値 : [ true / false ]

設定方法 (例)

```

<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT_NUMERIC">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value=""/>
    <arg key="maxlength" value="100"/>
    <arg key="readonly" value="false"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="disabled" value="false"/>
    <arg key="placeholder" value="targetField"/>
    <arg key="event" value="change"/>
    <arg key="autoFormat" value="true"/>
  </args>
</field>

```

## 表示状態

- PC

テキスト (数値)

- SP(スマートフォン)

テキスト (数値)

## テキストボックス(コード) [ type : TEXT\_CODE ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
maxlength	最大文字数を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]
placeholder	placeholder属性を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

## 設定方法 (例)

```

<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT_CODE">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value=""/>
    <arg key="maxlength" value="100"/>
    <arg key="readonly" value="false"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="disabled" value="false"/>
    <arg key="placeholder" value="targetField"/>
    <arg key="event" value="change"/>
  </args>
</field>

```

## 表示状態

- PC

テキスト (コード)

- SP(スマートフォン)

テキスト (コード)

## テキストボックス(日付) [ type : TEXT\_DATE ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]
autoFormat	日付自動フォーマット属性を設定します。設定値 : [ true / false ]
placeholder	placeholder属性を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

## 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT_DATE">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value=""/>
    <arg key="readonly" value="false"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="disabled" value="false"/>
    <arg key="autoFormat" value="true"/>
    <arg key="placeholder" value="targetField"/>
    <arg key="event" value="change"/>
  </args>
</field>
```

## 表示状態

- PC

テキスト (日付)

- SP(スマートフォン)

テキスト (日付)

### コラム

#### 日付自動フォーマットについて

日付自動フォーマット属性に「true」を設定した場合、フォーカスアウト時に次の様なフォーマット変換が行われます。  
(次の例はシステム日付が「2017/07/10」の場合の例です。)

入力値	フォーマット値
1	2017/07/01
5	2017/07/05
32	2017/08/01
1/1	2017/01/01
5/5	2017/05/05

### テキストボックス(時間) [ type : TEXT\_TIME ]

属性値	設定値
maxHour	時間の最大値を設定します。
timeFormat	時刻表現を設定します。 省略時は「BaseSettings.xml」の時刻表現 (timeFormat) を使用します。 H24 : 24時までを表現します。 H99 : 24時以降も表現します。(午前2時は「26時」と表現します。)
minutesUnit	時刻の最小入力単位 (分) を設定します。
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。

readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]
placeholder	placeholder属性を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

## 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT_TIME">
  <args>
    <arg key="maxHour" value="24"/>
    <arg key="timeFormat" value="H99"/>
    <arg key="minutesUnit" value="10"/>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value=""/>
    <arg key="readonly" value="false"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="disabled" value="false"/>
    <arg key="placeholder" value="targetField"/>
    <arg key="event" value="change"/>
  </args>
</field>
```

## 表示状態

- PC

テキスト (時間)

テキスト

- SP(スマートフォン)

テキスト (時間)

テキスト (時間)



## テキストボックス(パスワード) [ type : TEXT\_PASSWORD ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されま す。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
maxlength	最大文字数を設定します。
value	値を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]
placeholder	placeholder属性を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

## 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT_PASSWORD">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="maxLength" value="100"/>
    <arg key="value" value=""/>
    <arg key="readonly" value="false"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="disabled" value="false"/>
    <arg key="placeholder" value="targetField"/>
    <arg key="event" value="change"/>
  </args>
</field>
```

表示状態

- PC



- SP(スマートフォン)



### テキストボックス(Hidden) [ type : TEXT\_HIDDEN ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT_HIDDEN">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value=""/>
    <arg key="event" value="change"/>
  </args>
</field>
```

表示状態

表示なし

### テキストエリア [ type : TEXTAREA ]

属性値	設定値
-----	-----

imeMode	IME-MODEを設定します。設定値 : [ on / off / disabled ]
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
maxlength	最大文字数を設定します。
value	値を設定します。
cols	横の長さを設定します。
rows	行数を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]
placeholder	placeholder属性を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

#### 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXTAREA">
  <args>
    <arg key="imeMode" value="on"/>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="maxlength" value="100"/>
    <arg key="value" value=""/>
    <arg key="cols" value="5"/>
    <arg key="rows" value="10"/>
    <arg key="readonly" value="false"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="disabled" value="false"/>
    <arg key="placeholder" value="targetField"/>
    <arg key="event" value="change"/>
  </args>
</field>
```

#### 表示状態

- PC



- SP(スマートフォン)



## チェックボックス [ type : CHECKBOX ]

属性値	設定値
label	チェックボックス用のラベルを設定します。
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

## 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="CHECKBOX">
  <args>
    <arg key="label" value="targetField"/>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value=""/>
    <arg key="readonly" value="false"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="disabled" value="false"/>
    <arg key="event" value="click"/>
  </args>
</field>
```

## 表示状態

- PC

チェックボックス  チェックボックス

- SP(スマートフォン)

チェックボックス

チェックボックス



## コラム

チェックボックス用のラベルはサーフィスキーを設定することで、画面表示時などにサーフィス変換して使用されます。

## ラジオボタン [ type : RADIO ]

属性値	設定値
label	ラジオボタン用のラベルを設定します。
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。

class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]

設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="RADIO">
  <args>
    <arg key="label" value="targetField"/>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value=""/>
    <arg key="readonly" value="false"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="disabled" value="false"/>
  </args>
</field>
```

表示状態

- PC



- SP(スマートフォン)



コラム

ラジオボタン用のラベルはサーフィスキーを設定することで、画面表示時などにサーフィス変換して使用されます。

サーフィス [ type : SURFACE ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されま す。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。

設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="SURFACE">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
  </args>
</field>
```

## 表示状態

- PC

サーフィス サーフィス

- SP(スマートフォン)

サーフィス

サーフィス



## コラム

表示されるサーフィスは、フィールド設定のサーフィスキーを使用します。

## ファイル [ type : FILE ]

属性値	設定値
multiple	複数ファイルを利用する場合設定します。設定値 : [ multiple ]
accept	受け入れるファイル形式を設定します。
url	ファイル操作で使用するURLを設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
disabled	disabled属性を設定します。設定値 : [ true / false ]
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]

## 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="FILE">
  <args>
    <arg key="multiple" value="multiple"/>
    <arg key="accept" value="pdf"/>
    <arg key="url" value=""/>
    <arg key="readonly" value="false"/>
    <arg key="disabled" value="false"/>
    <arg key="event" value="change"/>
  </args>
</field>
```

## 表示状態

- PC

ファイル 📁

- SP(スマートフォン)

ファイル

📁



## 注意

現在、フィールドは配置されますが、データ登録時のファイル操作は行われません。  
 使用する場合は、登録処理などを実装する必要があります。

属性値	設定値
masterId	マスタIDを設定します。
extensionId	マスタ拡張IDを設定します。
searchType	検索タイプを設定します。設定値 : [ search / select ]
viewName	表示フィールドのフィールド名を設定します。(主に名称フィールド)
criteria	検索条件 (JSON形式) を設定します。
resultChain	対象選択時に連動する項目を設定します。
resultValue	対象選択時に本フィールドの「value」属性に設定するキーを設定します。
primaryTarget	未選択状態を判定するフィールドの「name」属性を設定します。
refreshTarget	再検索するトリガーとなるフィールドを設定します。
singleList	個別リストデータ生成有無を設定します。設定値 : [ true / false ]
imeMode	IME-MODEを設定します。設定値 : [ on / off / disabled ]
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
title	ツールチップの文字列を設定します。
disabled	disabled属性を設定します。設定値 : [ true / false ]
placeholder	placeholder属性を設定します。
event	補完処理の実行タイミングを追加設定します。設定値 : [ click / change ]
uniqueListName	trueを設定するとリスト名がユニークになるため、各行ごとにリストがユニークになります。
listRefresh	searchTypeが「search」の場合にtrueを設定すると、初回リフレッシュ時にマスタデータをリストデータ形式で取得し、以降のリフレッシュはリストデータから行います。 サーバーサイドへのアクセスやSQLの実行がなくなるため、レスポンスが改善する場合があります。ただし、マスタデータが数十万件(環境によって件数は異なります)登録されている場合、初回リフレッシュ時のデータ量が増大し、ブラウザの処理が遅くなる可能性があります。 また、各明細のデータを検索条件に含んでいる場合は設定できません。 設定値 : [ true / false ]
ajaxRefresh	falseを設定すると、初回リフレッシュ時にサーバーサイドへ問い合わせせずに値を確定します。 カスタマイズ用の設定のため、想定通りの動作が実現できるか十分に検証の上ご利用ください。 設定値 : [ true / false ]
batchRefresh	trueを設定すると、初期表示時のリフレッシュを、trueが設定されている項目をガジェットごとにまとめて実施します。 サーバーサイドへのアクセス数が減少するため、レスポンスが改善する場合があります。 設定値 : [ true / false ]

## 設定方法 (例)

```
<field id="targetFieldCd" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="MASTER">
  <args>
    <arg key="masterId" value="masterId"/>
    <arg key="extensionId" value="extensionId"/>
    <arg key="searchType" value="search"/>
    <arg key="viewName" value="targetFieldName"/>
    <arg key="criteria" value="{ 'code':targetFieldCd,'includeBlank':false }"/>
    <arg key="resultChain" value="{ 'targetFieldCd':'code','targetFieldName':'name' }"/>
    <arg key="resultValue" value="name"/>
    <arg key="primaryTarget" value="targetFieldCd"/>
    <arg key="refreshTarget" value=""/>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="placeholder" value="targetField"/>
    <arg key="event" value="change"/>
  </args>
</field>
```

表示状態

- PC



- SP(スマートフォン)



**i** コラム

「listRefresh」、「ajaxRefresh」、「batchRefresh」はいずれかを指定してください。  
標準ガジェットでは、「batchRefresh」を設定しています。

ボタン [ type : BUTTON ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
value	値を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
disabled	disabled属性を設定します。設定値 : [ true / false ]
event	補完処理の実行タイミングを追加設定します。設定値 : [ click ]

設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="BUTTON">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="value" value="targetField"/>
    <arg key="readonly" value="false"/>
    <arg key="disabled" value="false"/>
    <arg key="event" value="click"/>
  </args>
</field>
```

## 表示状態

- PC



- SP(スマートフォン)



## アイコン [ type : ICON ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
text	アイコン横の文字列を設定します。
sprite	アイコン用のCSSSpriteを設定します。
title	ツールチップの文字列を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
disabled	disabled属性を設定します。設定値 : [ true / false ]
event	補完処理の実行タイミングを追加設定します。設定値 : [ click ]

## 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="ICON">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="text" value="text_surface_key"/>
    <arg key="sprite" value="im-ui-icon-common-16-settings"/>
    <arg key="title" value="title_surface_key"/>
    <arg key="readonly" value="false"/>
    <arg key="disabled" value="false"/>
    <arg key="event" value="click"/>
  </args>
</field>
```

## 表示状態

- PC

アイコン  text\_surface\_key

- SP(スマートフォン)

アイコン

 text\_surface\_key



## コラム

アイコン横の文字列、ツールチップの文字列はサーフィスキーを設定することで、画面表示時などにサーフィス変換して使用されます。

## アンカー [ type : ANCHOR ]

属性値	設定値
class	class属性の値を設定します。 class_pc、class_spが設定されている場合は、各モードの設定が優先されます。
class_pc	PCモードで使用するclass属性の値を設定します。
class_sp	SP(スマートフォン)モードで使用するclass属性の値を設定します。
style	style属性の値を設定します。 style_pc、style_spが設定されている場合は、各モードの設定が優先されます。
style_pc	PCモードで使用するstyle属性の値を設定します。
style_sp	SP(スマートフォン)モードで使用するstyle属性の値を設定します。
text	アンカー用の文字列を設定します。
readonly	readonly属性を設定します。設定値 : [ true / false ]
disabled	disabled属性を設定します。設定値 : [ true / false ]
event	補完処理の実行タイミングを追加設定します。設定値 : [ click ]

## 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="ANCHOR">
  <args>
    <arg key="class" value="common_class"/>
    <arg key="class_pc" value="pc_mode_class"/>
    <arg key="class_sp" value="sp_mode_class"/>
    <arg key="style" value="common_style"/>
    <arg key="style_pc" value="pc_mode_style"/>
    <arg key="style_sp" value="sp_mode_style"/>
    <arg key="text" value="text_surface_key"/>
    <arg key="readonly" value="false"/>
    <arg key="disabled" value="false"/>
    <arg key="event" value="click"/>
  </args>
</field>
```

## 表示状態

- PC

アンカー text\_surface\_key

- SP(スマートフォン)

アンカー

text\_surface\_key



## コラム

アンカー用の文字列はサーフィスキーを設定することで、画面表示時などにサーフィス変換して使用されます。

## 操作パネル [ type : PANEL ]

属性値	設定値
header	ヘッダー用の操作パネル表示有無を設定します。設定値 : [ true / false ]
headerAllowAdd	ヘッダー用の操作パネルでの追加操作可否を設定します。設定値 : [ true / false ]
headerAllowDel	ヘッダー用の操作パネルでの削除操作可否を設定します。設定値 : [ true / false ]
detail	明細用の操作パネル表示有無を設定します。設定値 : [ true / false ]
detailAllowAdd	明細用の操作パネルでの追加操作可否を設定します。設定値 : [ true / false ]
detailAllowDel	明細用の操作パネルでの削除操作可否を設定します。設定値 : [ true / false ]

## 設定方法 (例)

```
<field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="PANEL">
  <args>
    <arg key="header" value="true"/>
    <arg key="headerAllowAdd" value="true"/>
    <arg key="headerAllowDel" value="true"/>
    <arg key="detail" value="true"/>
    <arg key="detailAllowAdd" value="true"/>
    <arg key="detailAllowDel" value="true"/>
  </args>
</field>
```

## 表示状態

- PC



- SP(スマートフォン)



## DTO値 [ type : DTO ]

## 設定方法 (例)

```
<field id="searchCriteriaCompany" surfaceKey="searchCriteriaCompany" requiredType="ALWAYS_ANY" inputType="INHERIT" type="DTO"/>
```

## 表示状態

表示なし



## コラム

画面表示用のDTOの変数と同じフィールドIDの場合に、DTOから値を取得するフィールドを生成します。  
申請書ガジェットの場合は次の値が使用可能です。

名称	フィールドID
ページパスID	pagePathId
グループID(IMW)	imwGroupId
処理者CD(IMW)	imwUserCode
画面種別(IMW)	imwPageType
ユーザデータID(IMW)	imwUserDataId
システム案件ID(IMW)	imwSystemMatterId
処理対象ノードID(IMW)	imwNodeId
到達種別(IMW)	imwArriveType
権限者CD(IMW)	imwAuthUserCode
申請基準日(IMW)	imwApplyBaseDate
フローID(IMW)	imwFlowId
フローバージョンID(IMW)	imwFlowVersionId
コンテンツID(IMW)	imwContentsId
コンテンツバージョンID(IMW)	imwContentsVersionId
ルートID(IMW)	imwRouteId
ルートバージョンID(IMW)	imwRouteVersionId
呼出元パラメータ(IMW)	imwCallOriginalParams
呼出元ページパス(IMW)	imwCallOriginalPagePath
検索基準会社コード	searchCriteriaCompany
検索基準会社名	searchCriteriaCompanyName
申請基準日	searchCriteriaDate
申請権限者名	applyAuthUserName
申請処理者名	applyUserName
案件名	matterName
案件名ビューモード	matterNameViewMode
案件番号	matterNumber
申請組織セットコード	applyDepartmentSetCd
申請組織コード	applyDepartmentCd
申請組織名	applyDepartmentName
フロー名	flowName
代理フラグ	actFlag
申請者名(代理申請者)	authUserAndActUserName
申請権限者コード	applyAuthUserCd
換算額計算丸めモード	amountConvertRoundMode
会社通貨コード	companyCurrencyCd
印刷ボタン表示設定フラグ	printDispFlag
台紙印刷ボタン表示設定フラグ	pasteBordDispFlag
申請書タイプ	appType



## コラム

本タイプはカスタマイズ用となります。(カスタマイズ時に属性値を決定してください。)

## バリデート処理

GadgetBuilder定義の「[フィールド設定](#)」で設定可能なバリデート処理について説明します。

- intra-mart Accel Kaiden!標準のバリデート処理
  - 必須
  - 必須(checkbox)
  - 条件付き必須(一致)
  - 条件付き必須(不一致)
  - 条件付き必須(空の場合)
  - 条件付き必須(空ではない場合)
  - 条件付き必須(チェック済み)
  - 条件付き必須(ラジオボタン)
  - いずれか必須
  - いずれか必須(checkbox)
  - 条件付きいずれか必須(一致)
  - 条件付きいずれか必須(不一致)
  - 条件付きいずれか必須(空の場合)
  - 条件付きいずれか必須(空ではない場合)
  - 他項目必須
  - コード(intra-mart Accel Kaiden!)
  - コード(intra-mart Accel Platform)
  - ロケール
  - フラグ
  - 最小文字数
  - 最大文字数
  - 固定長文字数
  - 半角英字
  - 半角英数
  - 半角数字
  - 半角英数記号
  - サロゲート文字以外
  - 整数
  - 整数部・小数部
  - 最小数値
  - 最小数値(等号選択)
  - 最大数値
  - 最大数値(等号選択)
  - 数値範囲
  - 数値範囲(等号選択)
  - 最小金額
  - 最小金額(等号選択)
  - 最大金額
  - 最大金額(等号選択)
  - 金額範囲
  - 金額範囲(等号選択)
  - 通貨額
  - 通貨額(カンマ区切りを除外)
  - 日付(形式)
  - 最小日付
  - 最小日付(等号選択)
  - 最大日付
  - 最大日付(等号選択)
  - 日付範囲
  - 日付範囲(等号選択)

- 時刻(形式)
- 最小時刻
- 最小時刻(等号選択)
- 最大時刻
- 最大時刻(等号選択)
- 時刻範囲
- 時刻範囲(等号選択)
- 時間入力単位
- 日時(形式)
- 正規表現
- 不一致
- 一致
- FB文字
- マスタ存在(MasterBuilder)
- マスタ非存在(MasterBuilder)
- LogicDesigner
- 同一日同一経路

## intra-mart Accel Kaiden!標準のバリデート処理

本項では、intra-mart Accel Kaiden!で提供しているバリデート処理と設定方法を説明します。

### 必須

#### 概要

本バリデーションが設定されたフィールドに、値が入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドに値が入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドに値が入力されていない。

#### 設定方法

```
<validate event="default" config="{id: 'required'}"/>
```

#### 詳細

バリデーションID	required
引数	なし
出力メッセージ	必須です。

### 必須(checkbox)

#### 概要

本バリデーションが設定されたフィールド(checkbox)がチェックされているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールド(checkbox)がチェックされている。
- **チェックNG**  
本バリデーションが設定されたフィールド(checkbox)がチェックされていない。

#### 設定方法

```
<validate event="default" config="{id: 'onForCheck'}"/>
```

#### 詳細

バリデーションID	onForCheck
引数	なし
出力メッセージ	必ずチェックしてください。

## 条件付き必須(一致)

## 概要

判定対象1と判定対象2に設定されたフィールドの値が一致した場合、本バリデーションが設定されたフィールドに、値が入力されているか否かを判定します。

- **チェックOK**  
判定対象1と判定対象2に設定されたフィールドの値が一致しない。  
判定対象1と判定対象2に設定されたフィールドの値が一致し、本バリデーションが設定されたフィールドに値が入力されている。
- **チェックNG**  
判定対象1と判定対象2に設定されたフィールドの値が一致し、本バリデーションが設定されたフィールドに値が入力されていない。

## 設定方法

```
<validate event="default" config="{id:'requiredIfEqual', 'args':['targetField1','targetField2']}" />
```

## 詳細

バリデーションID requiredIfEqual

引数 (配列)

1. 判定対象1 (必須)
2. 判定対象2 (必須)

出力メッセージ

{0}と{1}が一致する場合は必須です。  
{0} : 判定対象1に設定されたフィールドの名称を出力します。  
{1} : 判定対象2に設定されたフィールドの名称を出力します。

## 条件付き必須(不一致)

## 概要

判定対象1と判定対象2に設定されたフィールドの値が一致しない場合、本バリデーションが設定されたフィールドに、値が入力されているか否かを判定します。

- **チェックOK**  
判定対象1と判定対象2に設定されたフィールドの値が一致する。  
判定対象1と判定対象2に設定されたフィールドの値が一致せず、本バリデーションが設定されたフィールドに値が入力されている。
- **チェックNG**  
判定対象1と判定対象2に設定されたフィールドの値が一致せず、本バリデーションが設定されたフィールドに値が入力されていない。

## 設定方法

```
<validate event="default" config="{id:'requiredIfNotEqual', 'args':['targetField1','targetField2']}" />
```

## 詳細

バリデーションID requiredIfNotEqual

引数 (配列)

1. 判定対象1 (必須)
2. 判定対象2 (必須)

出力メッセージ

{0}と{1}が一致しない場合は必須です。  
{0} : 判定対象1に設定されたフィールドの名称を出力します。  
{1} : 判定対象2に設定されたフィールドの名称を出力します。

## 条件付き必須(空の場合)

## 概要

判定対象に設定されたフィールドが空の場合、本バリデーションが設定されたフィールドに、値が入力されているか否かを判定します。

- **チェックOK**  
判定対象に設定されたフィールドに値が入力されている。  
判定対象に設定されたフィールドに値が入力されておらず、本バリデーションが設定されたフィールドに値が入力されている。
- **チェックNG**  
判定対象に設定されたフィールドに値が入力されておらず、本バリデーションが設定されたフィールドに値が入力されていない。

## 設定方法

```
<validate event="default" config="{ 'id': 'requiredIfEmpty', 'args': 'targetField1' }"/>
```

#### 詳細

バリデーションID requiredIfEmpty

引数  
1. 判定対象 (必須)

出力メッセージ {0}が空の場合は必須です。  
{0} : 判定対象に設定されたフィールドの名称を出力します。

#### 条件付き必須(空ではない場合)

##### 概要

判定対象に設定されたフィールドが空ではない場合、本バリデーションが設定されたフィールドに、値が入力されているか否かを判定します。

- **チェックOK**  
判定対象に設定されたフィールドに値が入力されていない。  
判定対象に設定されたフィールドに値が入力されており、本バリデーションが設定されたフィールドに値が入力されている。
- **チェックNG**  
判定対象に設定されたフィールドに値が入力されており、本バリデーションが設定されたフィールドに値が入力されていない。

##### 設定方法

```
<validate event="default" config="{ 'id': 'requiredIfNotEmpty', 'args': 'targetField1' }"/>
```

#### 詳細

バリデーションID requiredIfNotEmpty

引数  
1. 判定対象 (必須)

出力メッセージ {0}が空ではない場合は必須です。  
{0} : 判定対象に設定されたフィールドの名称を出力します。

#### 条件付き必須(チェック済み)

##### 概要

判定対象に設定されたフィールドがチェックされている場合、本バリデーションが設定されたフィールドに、値が入力されているか否かを判定します。

- **チェックOK**  
判定対象に設定されたフィールドがチェックされていない。  
判定対象に設定されたフィールドがチェックされており、本バリデーションが設定されたフィールドに値が入力されている。
- **チェックNG**  
判定対象に設定されたフィールドがチェックされており、本バリデーションが設定されたフィールドに値が入力されていない。

##### 設定方法

```
<validate event="default" config="{ 'id': 'requiredIfChecked', 'args': 'targetField1' }"/>
```

#### 詳細

バリデーションID requiredIfChecked

引数  
1. 判定対象 (必須)

出力メッセージ {0}がチェックされている場合は必須です。  
{0} : 判定対象に設定されたフィールドの名称を出力します。

#### 条件付き必須(ラジオボタン)

##### 概要

判定対象に設定されたフィールドのラジオボタンの値に「1」が設定されている場合、本バリデーションが設定されたフィールドに、値が入力されているか否かを判定します。

- **チェックOK**

判定対象に設定されたフィールドのラジオボタンの値に「1」が設定されていて、本バリデーションが設定されたフィールドに値が入力されている。

判定対象に設定されたフィールドのラジオボタンの値に「0」が設定されている。

- **チェックNG**

判定対象に設定されたフィールドのラジオボタンの値に「1」が設定されていて、本バリデーションが設定されたフィールドに値が入力されていない。

#### 設定方法

```
<validate event="default" config="{ 'id': 'requiredIfFlag', 'args': 'targetField1' }"/>
```

#### 詳細

バリデーションID requiredIfFlag

引数

1. 判定対象 (必須)

出力メッセージ {0}が{1}の場合には必須です。

{0} : 判定対象に設定されたフィールドの名称を出力します。

{1} : 判定対象に設定されたフィールドの値が「1」の選択肢の名称を出力します。

#### いずれか必須

##### 概要

引数で設定されたすべてのフィールドに、値が入力されているか否かを判定します。

- **チェックOK**

引数で設定されたいずれかのフィールドに値が入力されている。

- **チェックNG**

引数で設定されたすべてのフィールドに値が入力されていない。

#### 設定方法

```
<validate event="default" config="{ 'id': 'requiredAny', 'args': ['checkField1', 'checkField2', ...] }"/>
```

#### 詳細

バリデーションID requiredAny

引数 (配列)

1. チェック対象 1 (必須)

2. チェック対象 2 (以降は任意設定)

出力メッセージ いずれか必須です。[{0}]

{0} : チェック対象 1 に設定されたフィールドの名称を出力します。

#### いずれか必須(checkbox)

##### 概要

引数で設定されたすべてのフィールド(checkbox)が、選択されているか否かを判定します。

- **チェックOK**

引数で設定されたいずれかのフィールドが選択されている。

- **チェックNG**

引数で設定されたすべてのフィールドが選択されていない。

#### 設定方法

```
<validate event="default" config="{ 'id': 'anyOnForCheck', 'args': ['checkbox1', 'checkbox2', ...] }"/>
```

#### 詳細

バリデーションID anyOnForCheck

引数（配列）	<ol style="list-style-type: none"> <li>1. チェック対象 1（必須）</li> <li>2. チェック対象 2（以降は任意設定）</li> </ol>
出力メッセージ	<p>いずれか必須です。[{0}]</p> <p>{0}：チェック対象 1 に設定されたフィールドの名称を出力します。</p>

### 条件付きいずれか必須(一致)

#### 概要

判定対象 1 と判定対象 2 に設定されたフィールドの値が一致した場合、チェック対象に設定されたすべてのフィールドに、値が入力されているか否かを判定します。

- チェックOK**  
 判定対象 1 と判定対象 2 に設定されたフィールドの値が一致しない。  
 判定対象 1 と判定対象 2 に設定されたフィールドの値が一致し、チェック対象に設定されたいずれかのフィールドに値が入力されている。
- チェックNG**  
 判定対象 1 と判定対象 2 に設定されたフィールドの値が一致し、チェック対象に設定されたすべてのフィールドに値が入力されていない。

#### 設定方法

```
<validate event="default" config="{ 'id':'requiredAnyIfEqual', 'args':['targetField1','targetField2','checkField1','checkField2',...]}"/>
```

#### 詳細

バリデーションID requiredAnyIfEqual

引数（配列）	<ol style="list-style-type: none"> <li>1. 判定対象 1（必須）</li> <li>2. 判定対象 2（必須）</li> <li>3. チェック対象 1（必須）</li> <li>4. チェック対象 2（以降は任意設定）</li> </ol>
出力メッセージ	<p>{0}と{1}が一致する場合はいずれか必須です。[{2}]</p> <p>{0}：判定対象 1 に設定されたフィールドの名称を出力します。</p> <p>{1}：判定対象 2 に設定されたフィールドの名称を出力します。</p> <p>{2}：チェック対象 1 に設定されたフィールドの名称を出力します。</p>

### 条件付きいずれか必須(不一致)

#### 概要

判定対象 1 と判定対象 2 に設定されたフィールドの値が一致しない場合、チェック対象に設定されたすべてのフィールドに、値が入力されているか否かを判定します。

- チェックOK**  
 判定対象 1 と判定対象 2 に設定されたフィールドの値が一致する。  
 判定対象 1 と判定対象 2 に設定されたフィールドの値が一致せず、チェック対象に設定されたいずれかのフィールドに値が入力されている。
- チェックNG**  
 判定対象 1 と判定対象 2 に設定されたフィールドの値が一致せず、チェック対象に設定されたすべてのフィールドに値が入力されていない。

#### 設定方法

```
<validate event="default" config="{ 'id':'requiredAnyIfNotEqual', 'args':['targetField1','targetField2','checkField1','checkField2',...]}"/>
```

#### 詳細

バリデーションID requiredAnyIfNotEqual

引数 (配列)	<ol style="list-style-type: none"> <li>1. 判定対象 1 (必須)</li> <li>2. 判定対象 2 (必須)</li> <li>3. チェック対象 1 (必須)</li> <li>4. チェック対象 2 (以降は任意設定)</li> </ol>
出力メッセージ	<p>{0}と{1}が一致しない場合はいずれか必須です。[{2}]</p> <p>{0}：判定対象 1 に設定されたフィールドの名称を出力します。</p> <p>{1}：判定対象 2 に設定されたフィールドの名称を出力します。</p> <p>{2}：チェック対象 1 に設定されたフィールドの名称を出力します。</p>

### 条件付きいずれか必須(空の場合)

#### 概要

判定対象に設定されたフィールドが空の場合、チェック対象に設定されたすべてのフィールドに、値が入力されているか否かを判定します。

- **チェックOK**  
判定対象に設定されたフィールドに値が入力されている。  
判定対象に設定されたフィールドに値が入力されておらず、チェック対象に設定されたいずれかのフィールドに値が入力されている。
- **チェックNG**  
判定対象に設定されたフィールドに値が入力されておらず、チェック対象に設定されたすべてのフィールドに値が入力されていない。

#### 設定方法

```
<validate event="default" config="{ 'id': 'requiredAnyIfEmpty', 'args': ['targetField', 'checkField1'(', 'checkField2', ...) ] }"/>
```

#### 詳細

バリデーションID requiredAnyIfEmpty

引数 (配列)	<ol style="list-style-type: none"> <li>1. 判定対象 (必須)</li> <li>2. チェック対象 1 (必須)</li> <li>3. チェック対象 2 (以降は任意設定)</li> </ol>
---------	---

出力メッセージ	<p>{0}が空の場合はいずれか必須です。[{1}]</p> <p>{0}：判定対象に設定されたフィールドの名称を出力します。</p> <p>{1}：チェック対象 1 に設定されたフィールドの名称を出力します。</p>
---------	---

### 条件付きいずれか必須(空ではない場合)

#### 概要

判定対象に設定されたフィールドが空ではない場合、チェック対象に設定されたすべてのフィールドに、値が入力されているか否かを判定します。

- **チェックOK**  
判定対象に設定されたフィールドに値が入力されていない。  
判定対象に設定されたフィールドに値が入力されており、チェック対象に設定されたいずれかのフィールドに値が入力されている。
- **チェックNG**  
判定対象に設定されたフィールドに値が入力されており、チェック対象に設定されたすべてのフィールドに値が入力されていない。

#### 設定方法

```
<validate event="default" config="{ 'id': 'requiredAnyIfNotEmpty', 'args': ['targetField', 'checkField1'(', 'checkField2', ...) ] }"/>
```

#### 詳細

バリデーションID requiredAnyIfNotEmpty

引数 (配列)	<ol style="list-style-type: none"> <li>1. 判定対象 (必須)</li> <li>2. チェック対象 1 (必須)</li> <li>3. チェック対象 2 (以降は任意設定)</li> </ol>
---------	---

出力メッセージ {0}が空ではない場合はいずれか必須です。[{1}]  
 {0}：判定対象に設定されたフィールドの名称を出力します。  
 {1}：チェック対象1に設定されたフィールドの名称を出力します。

## 他項目必須

### 概要

本バリデーションが設定されたフィールドが空ではない場合、チェック対象に設定されたすべてのフィールドに、値が入力されているか否かを判定します。

- チェックOK**  
 本バリデーションが設定されたフィールドに値が入力されていない。  
 本バリデーションが設定されたフィールドに値が入力されており、チェック対象に設定されたすべてのフィールドに値が入力されている。
- チェックNG**  
 本バリデーションが設定されたフィールドに値が入力されており、チェック対象に設定されたいずれかのフィールドに値が入力されていない。

### 設定方法

```
<validate event="default" config="{ 'id':'requiredOthers', 'args':['targetField(',')','targetField2',...] }"/>
```

### 詳細

バリデーションID requiredOthers

引数（配列）

- チェック対象1（必須）
- チェック対象2（以降は任意設定）

出力メッセージ 全て必須です。[{0}]  
 {0}：チェック対象に設定されたフィールドの名称を出力します。

## コード(intra-mart Accel Kaiden!)

### 概要

本バリデーションが設定されたフィールドの値が、intra-mart Accel Kaiden!のコード体系で入力されているか否かを判定します。

- チェックOK**  
 本バリデーションが設定されたフィールドの値がintra-mart Accel Kaiden!のコード体系のみで入力されている。
- チェックNG**  
 本バリデーションが設定されたフィールドの値にintra-mart Accel Kaiden!のコード体系以外の文字が含まれている。

### 設定方法

```
<validate event="default" config="{ 'id':'kaidenId' }"/>
```

### 詳細

バリデーションID kaidenId

引数 なし

出力メッセージ 半角英数字、「\_」、「-」で入力してください。



#### コラム

intra-mart Accel Kaiden!のコード体系に含まれる文字は次の通りです。

- 半角英数字：A-Za-z0-9
- 記号：\_-

## コード(intra-mart Accel Platform)

### 概要

本バリデーションが設定されたフィールドの値が、intra-mart Accel Platformのコード体系で入力されているか否かを判定します。

- チェックOK**  
 本バリデーションが設定されたフィールドの値がintra-mart Accel Platformのコード体系のみで入力されている。
- チェックNG**

本バリデーションが設定されたフィールドの値にintra-mart Accel Platformのコード体系以外の文字が含まれている。

#### 設定方法

```
<validate event="default" config="{id:'imId'}"/>
```

#### 詳細

バリデーションID imId

引数 なし

出力メッセージ 半角英数字、「\_」、「-」、「@」、「.」、「+」、「!」で入力してください。

#### ロケール

##### 概要

本バリデーションが設定されたフィールドの値が、intra-mart Accel Platformのシステムロケールに存在するロケールか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値がintra-mart Accel Platformのシステムロケールに存在する。
- **チェックNG**  
本バリデーションが設定されたフィールドの値がintra-mart Accel Platformのシステムロケールに存在しない。

#### 設定方法

```
<validate event="default" config="{id:'locale'}"/>
```

#### 詳細

バリデーションID locale

引数 なし

出力メッセージ 設定されたロケールは存在しません。

#### フラグ

##### 概要

本バリデーションが設定されたフィールドの値が、「0」または「1」で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「0」または「1」で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「0」または「1」で入力されていない。

#### 設定方法

```
<validate event="default" config="{id:'flag'}"/>
```

#### 詳細

バリデーションID flag

引数 なし

出力メッセージ フラグとして認識できない文字です。

#### 最小文字数

##### 概要

本バリデーションが設定されたフィールドの値が、「最小文字数 <= 入力文字数」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小文字数 <= 入力文字数」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小文字数 <= 入力文字数」の範囲を超えて入力されている。

#### 設定方法

```
<validate event="default" config="{ 'id':'minLength', 'args':{'250'}}"/>
```

#### 詳細

バリデーションID minLength

引数  
1. 最小文字数 (必須)

出力メッセージ {0}文字以上で入力してください。  
{0} : 引数で設定された最小文字数を出力します。

#### 最大文字数

##### 概要

本バリデーションが設定されたフィールドの値が、「入力文字数 ≤ 最大文字数」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「入力文字数 ≤ 最大文字数」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「入力文字数 ≤ 最大文字数」の範囲を超えて入力されている。

##### 設定方法

```
<validate event="default" config="{ 'id':'maxLength', 'args':{'250'}}"/>
```

#### 詳細

バリデーションID maxLength

引数  
1. 最大文字数 (必須)

出力メッセージ {0}文字以下で入力してください。  
{0} : 引数で設定された最小文字数を出力します。

#### 固定長文字数

##### 概要

本バリデーションが設定されたフィールドの値が、固定文字数で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が固定文字数で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が固定文字数以外の文字数で入力されている。

##### 設定方法

```
<validate event="default" config="{ 'id':'fixedLength', 'args':{'250'}}"/>
```

#### 詳細

バリデーションID fixedLength

引数  
1. 固定文字数 (必須)

出力メッセージ {0}文字で入力してください。  
{0} : 引数で設定された固定文字数を出力します。

#### 半角英字

##### 概要

本バリデーションが設定されたフィールドの値が、半角英字 (A-Za-z) で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が半角英字のみで入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値に半角英字以外の文字が含まれている。

## 設定方法

```
<validate event="default" config="{id: 'alpha'}"/>
```

## 詳細

バリデーションID	alpha
引数	なし
出力メッセージ	半角英字で入力してください。

## 半角英数

## 概要

本バリデーションが設定されたフィールドの値が、半角英数字（A-Za-z0-9）で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が半角英数字のみで入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値に半角英数字以外の文字が含まれている。

## 設定方法

```
<validate event="default" config="{id: 'alphaNumeric'}"/>
```

## 詳細

バリデーションID	alphaNumeric
引数	なし
出力メッセージ	半角英数字で入力してください。

## 半角数字

## 概要

本バリデーションが設定されたフィールドの値が、半角数字（0-9）で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が半角数字のみで入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値に半角数字以外の文字が含まれている。

## 設定方法

```
<validate event="default" config="{id: 'numeric'}"/>
```

## 詳細

バリデーションID	numeric
引数	なし
出力メッセージ	半角数字で入力してください。

## 半角英数記号

## 概要

本バリデーションが設定されたフィールドの値が、半角英数（A-Za-z0-9）と半角記号で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が半角英数記号のみで入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値に半角英数記号以外の文字が含まれている。

## 設定方法

```
<validate event="default" config="{id: 'alphaNumericSymbol'}"/>
```

## 詳細

バリデーションID alphaNumericSymbol

引数 なし

出力メッセージ 半角英数字記号で入力してください。

## サロゲート文字以外

## 概要

本バリデーションが設定されたフィールドの値が、サロゲート文字以外の文字で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値にサロゲート文字が含まれていない。
- **チェックNG**  
本バリデーションが設定されたフィールドの値にサロゲート文字が含まれている。

## 設定方法

```
<validate event="default" config="{ 'id': 'notSurrogatePair' }"/>
```

## 詳細

バリデーションID notSurrogatePair

引数 なし

出力メッセージ サロゲート文字以外を入力してください。

## 整数

## 概要

本バリデーションが設定されたフィールドの値が、半角数字、マイナス（0-9-）で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が半角数字、マイナスのみで入力されており、整数部桁数以内の文字数で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値にマイナス以外の文字が含まれている、または、整数部桁数を超える文字数で入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'integer', 'args': '10' }"/>
```

## 詳細

バリデーションID integer

引数 1. マイナス記号の桁数を除いた整数部の桁数（必須）

出力メッセージ {0}桁までの数値を入力してください。  
{0}：引数で設定された整数部の桁数を出力します。

## 整数部・小数部

## 概要

本バリデーションが設定されたフィールドの値が、半角数字、ドット、マイナス（0-9.-）で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が半角数字、ドット、マイナスのみで入力されており、整数部、小数部が設定桁数以内の文字数で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値に半角数字、ドット、マイナス以外の文字が含まれている、または、整数部、小数部のいずれかが設定桁数を超える文字数で入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'digits', 'args': ['18', '10'] }"/>
```

#### 詳細

バリデーションID digits

引数 (配列)

1. 整数部の桁数 (必須)
2. 小数部の桁数 (必須)

出力メッセージ 整数部{0}桁、小数部{1}桁を入力してください。  
 {0} : 引数で設定された整数部の桁数を出力します。  
 {1} : 引数で設定された小数部の桁数を出力します。

### 最小数値

#### 概要

本バリデーションが設定されたフィールドの値が、「最小数値  $\leq$  入力値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小数値  $\leq$  入力値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小数値  $\leq$  入力値」の範囲を超えて入力されている。

#### 設定方法

```
<validate event="default" config="{ 'id': 'numericMin', 'args': '0' }"/>
```

#### 詳細

バリデーションID numericMin

引数

1. 最小数値 (必須)  
(最小数値を保持するフィールドを設定することも可能です。)

出力メッセージ {0}以上で入力してください。  
 {0} : 引数で設定された最小数値を出力します。



#### 注意

最小数値、入力値の両方が数値変換できない場合はチェックを実施しません。  
 数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

### 最小数値(等号選択)

#### 概要

本バリデーションが設定されたフィールドの値が、「最小数値  $\leq$  (または  $<$ ) 入力値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小数値  $\leq$  (または  $<$ ) 入力値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小数値  $\leq$  (または  $<$ ) 入力値」の範囲を超えて入力されている。

#### 設定方法

```
<validate event="default" config="{ 'id': 'numericMin2', 'args': ['0', 'true'] }"/>
```

#### 詳細

バリデーションID numericMin2

## 引数 (配列)

1. 最小数値 (必須)  
(最小数値を保持するフィールドを設定することも可能です。)
2. 等号有無 (任意) :省略時は「false」
  - true : 最小数値  $\leq$  入力値
  - false : 最小数値  $<$  入力値

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以上で入力してください。
- {0}より大きい値で入力してください。

{0} : 引数で設定された最小数値を出力します。

 **注意**

最小数値、入力値の両方が数値変換できない場合はチェックを実施しません。  
数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 最大数値

## 概要

本バリデーションが設定されたフィールドの値が、「入力値  $\leq$  最大数値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「入力値  $\leq$  最大数値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「入力値  $\leq$  最大数値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id':'numericMax', 'args':'99999' }"/>
```

## 詳細

バリデーションID numericMax

## 引数

1. 最大数値 (必須)  
(最大数値を保持するフィールドを設定することも可能です。)

出力メッセージ {0}以下で入力してください。

{0} : 引数で設定された最大数値を出力します。

 **注意**

最大数値、入力値の両方が数値変換できない場合はチェックを実施しません。  
数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 最大数値(等号選択)

## 概要

本バリデーションが設定されたフィールドの値が、「入力値  $\leq$  (または  $<$ ) 最大数値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「入力値  $\leq$  (または  $<$ ) 最大数値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「入力値  $\leq$  (または  $<$ ) 最大数値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id':'numericMax2', 'args':['99999','true'] }"/>
```

## 詳細

バリデーションID numericMax2

## 引数 (配列)

1. 最大数値 (必須)  
(最大数値を保持するフィールドを設定することも可能です。)
2. 等号有無 (任意) :省略時は「false」
  - true : 入力値 <= 最大数値
  - false : 入力値 < 最大数値

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以下で入力してください。
- {0}未満で入力してください。

{0} : 引数で設定された最大数値を出力します。

 **注意**

最大数値、入力値の両方が数値変換できない場合はチェックを実施しません。  
数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 数値範囲

## 概要

本バリデーションが設定されたフィールドの値が、「最小数値 <= 入力値 <= 最大数値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小数値 <= 入力値 <= 最大数値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小数値 <= 入力値 <= 最大数値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id':'numericRange', 'args':['0','99999'] }"/>
```

## 詳細

バリデーションID numericRange

## 引数 (配列)

1. 最小数値 (必須)  
(最小数値を保持するフィールドを設定することも可能です。)
2. 最大数値 (必須)  
(最大数値を保持するフィールドを設定することも可能です。)

出力メッセージ {0} - {1}の範囲で入力してください。

{0} : 引数で設定された最小数値を出力します。  
{1} : 引数で設定された最大数値を出力します。

 **注意**

最小数値、最大数値、入力値のすべてが数値変換できない場合はチェックを実施しません。  
数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 数値範囲(等号選択)

## 概要

本バリデーションが設定されたフィールドの値が、「最小数値 <= (または <) 入力値 <= (または <) 最大数値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小数値 <= (または <) 入力値 <= (または <) 最大数値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小数値 <= (または <) 入力値 <= (または <) 最大数値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id':'numericRange2', 'args':['0','99999','true','true'] }"/>
```

## 詳細

バリデーションID numericRange2

引数 (配列)

1. 最小数値 (必須)  
(最小数値を保持するフィールドを設定することも可能です。)
2. 最大数値 (必須)  
(最大数値を保持するフィールドを設定することも可能です。)
3. 等号有無 1 (任意) :省略時は「false」
  - true : 最小数値 <= 入力値
  - false : 最小数値 < 入力値
4. 等号有無 2 (任意) :省略時は「false」
  - true : 入力値 <= 最大数値
  - false : 入力値 < 最大数値

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以上、{1}以下の範囲で入力してください。
- {0}以上、{1}未満の範囲で入力してください。
- {0}超、{1}以下の範囲で入力してください。
- {0}超、{1}未満の範囲で入力してください。

{0} : 引数で設定された最小数値を出力します。

{1} : 引数で設定された最大数値を出力します。

### 注意

最小数値、最大数値、入力値のすべてが数値変換できない場合はチェックを実施しません。  
数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 最小金額

### 概要

本バリデーションが設定されたフィールドのカンマを除外した値が、「最小数値 <= 入力値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドのカンマを除外した値が「最小数値 <= 入力値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドのカンマを除外した値が「最小数値 <= 入力値」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id':'amountMin', 'args':['0'] }"/>
```

## 詳細

バリデーションID amountMin

引数

1. 最小数値 (必須)  
(最小数値を保持するフィールドを設定することも可能です。)

出力メッセージ {0}以上で入力してください。

{0} : 引数で設定された最小数値を出力します。

## 最小金額(等号選択)

### 概要

本バリデーションが設定されたフィールドのカンマを除外した値が、「最小数値 <= (または <) 入力値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドのカンマを除外した値が「最小数値 <= (または <) 入力値」の範囲内で入力されている。

- **チェックNG**

本バリデーションが設定されたフィールドのカンマを除外した値が「最小数値 <= (または <) 入力値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'amountMin2', 'args': ['0', 'true'] }"/>
```

## 詳細

バリデーションID amountMin2

引数 (配列)

1. 最小数値 (必須)  
(最小数値を保持するフィールドを設定することも可能です。)
2. 等号有無 (任意) :省略時は「false」
  - true : 最小数値 <= 入力値
  - false : 最小数値 < 入力値

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以上で入力してください。
- {0}より大きい値で入力してください。

{0} : 引数で設定された最小数値を出力します。

**注意**

最小数値、入力値の両方が数値変換できない場合はチェックを実施しません。  
数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 最大金額

## 概要

本バリデーションが設定されたフィールドのカンマを除外した値が、「入力値 <= 最大数値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドのカンマを除外した値が「入力値 <= 最大数値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドのカンマを除外した値が「入力値 <= 最大数値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'amountMax', 'args': '99999' }"/>
```

## 詳細

バリデーションID amountMax

引数

1. 最大数値 (必須)  
(最大数値を保持するフィールドを設定することも可能です。)

出力メッセージ {0}以下で入力してください。

{0} : 引数で設定された最大数値を出力します。

**注意**

最大数値、入力値の両方が数値変換できない場合はチェックを実施しません。  
数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 最大金額(等号選択)

## 概要

本バリデーションが設定されたフィールドのカンマを除外した値が、「入力値 <= (または <) 最大数値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドのカンマを除外した値が「入力値 <= (または <) 最大数値」の範囲内で入力されている。

- **チェックNG**

本バリデーションが設定されたフィールドのカンマを除外した値が「入力値 <= (または <) 最大数値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'amountMax2', 'args': ['99999', 'true'] }"/>
```

## 詳細

バリデーションID amountMax2

引数 (配列)

1. 最大数値 (必須)  
(最大数値を保持するフィールドを設定することも可能です。)
2. 等号有無 (任意) :省略時は「false」
  - true : 入力値 <= 最大数値
  - false : 入力値 < 最大数値

出力メッセージ

引数の条件で次のメッセージを表示します。

- {0}以下で入力してください。
- {0}未満で入力してください。

{0} : 引数で設定された最大数値を出力します。

**注意**

最大数値、入力値の両方が数値変換できない場合はチェックを実施しません。

数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 金額範囲

## 概要

本バリデーションが設定されたフィールドのカンマを除外した値が、「最小数値 <= 入力値 <= 最大数値」の範囲で入力されているか否かを判定します。

- **チェックOK**

本バリデーションが設定されたフィールドのカンマを除外した値が「最小数値 <= 入力値 <= 最大数値」の範囲内で入力されている。

- **チェックNG**

本バリデーションが設定されたフィールドのカンマを除外した値が「最小数値 <= 入力値 <= 最大数値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'amountRange', 'args': ['0', '99999'] }"/>
```

## 詳細

バリデーションID amountRange

引数 (配列)

1. 最小数値 (必須)  
(最小数値を保持するフィールドを設定することも可能です。)
2. 最大数値 (必須)  
(最大数値を保持するフィールドを設定することも可能です。)

出力メッセージ

{0} - {1}の範囲で入力してください。

{0} : 引数で設定された最小数値を出力します。

{1} : 引数で設定された最大数値を出力します。

**注意**

最小数値、最大数値、入力値のすべてが数値変換できない場合はチェックを実施しません。

数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 金額範囲(等号選択)

## 概要

本バリデーションが設定されたフィールドのカンマを除外した値が、「最小数値 <= (または <) 入力値 <= (または <) 最大数値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドのカンマを除外した値が「最小数値 <= (または <) 入力値 <= (または <) 最大数値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドのカンマを除外した値が「最小数値 <= (または <) 入力値 <= (または <) 最大数値」の範囲を超えて入力されている。

#### 設定方法

```
<validate event="default" config="{ 'id': 'amountRange2', 'args': [ '0', '99999', 'true', 'true' ] }"/>
```

#### 詳細

バリデーションID amountRange2

##### 引数 (配列)

1. 最小数値 (必須)  
(最小数値を保持するフィールドを設定することも可能です。)
2. 最大数値 (必須)  
(最大数値を保持するフィールドを設定することも可能です。)
3. 等号有無 1 (任意) :省略時は「false」
  - true : 最小数値 <= 入力値
  - false : 最小数値 < 入力値
4. 等号有無 2 (任意) :省略時は「false」
  - true : 入力値 <= 最大数値
  - false : 入力値 < 最大数値

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以上、{1}以下の範囲で入力してください。
- {0}以上、{1}未満の範囲で入力してください。
- {0}超、{1}以下の範囲で入力してください。
- {0}超、{1}未満の範囲で入力してください。

{0} : 引数で設定された最小数値を出力します。

{1} : 引数で設定された最大数値を出力します。

#### 注意

最小数値、最大数値、入力値のすべてが数値変換できない場合はチェックを実施しません。  
数値型のチェックは含まれないため、他のチェック (数字・整数・小数) と組み合わせて使用してください。

## 通貨額

#### 概要

本バリデーションが設定されたフィールドの値が、「対象通貨の精度・スケール」で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「対象通貨の精度・スケール」で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「対象通貨の精度・スケール」を超えて入力されている。

#### 設定方法

```
<validate event="default" config="{ 'id': 'currencyAmnt', 'args': [ 'currencyCodeField', 'companyCodeField' ] }"/>
```

#### 詳細

バリデーションID currencyAmnt

## 引数 (配列)

1. 通貨コードを保持するフィールド (必須)
2. 会社コードを保持するフィールド (必須)

## 出力メッセージ

整数部{0}桁、小数部{1}桁を入力してください。  
 {0}：引数で設定された通貨の精度からスケールを引いた数値を出力します。  
 {1}：引数で設定された通貨のスケールを出力します。

 コラム

通貨コードの精度・スケールは通貨精度区分が「KAIDEN」のデータのみをチェック対象とします。  
 通貨精度が見つからなかった場合は「整数部9、小数部4 (constansts.xmlの定義値)」を使用します。

## 通貨額(カンマ区切りを除外)

## 概要

本バリデーションが設定されたフィールドのカンマを除外した値が、「対象通貨の精度・スケール」で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドのカンマを除外した値が「対象通貨の精度・スケール」で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドのカンマを除外した値が「対象通貨の精度・スケール」を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'currencyAmnt2', 'args': [ 'currencyCodeField', 'companyCodeField' ] }"/>
```

## 詳細

バリデーションID currencyAmnt2

## 引数 (配列)

1. 通貨コードを保持するフィールド (必須)
2. 会社コードを保持するフィールド (必須)

## 出力メッセージ

整数部{0}桁、小数部{1}桁を入力してください。  
 {0}：引数で設定された通貨の精度からスケールを引いた数値を出力します。  
 {1}：引数で設定された通貨のスケールを出力します。

 コラム

通貨コードの精度・スケールは通貨精度区分が「KAIDEN」のデータのみをチェック対象とします。  
 通貨精度が見つからなかった場合は「整数部9、小数部4 (constansts.xmlの定義値)」を使用します。

## 日付(形式)

## 概要

本バリデーションが設定されたフィールドの値が、「日付形式 (YYYY/MM/DD) かつ、システムの有効期間内」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が日付形式で入力されており、システムの有効期間の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が日付形式以外で入力されている、またはシステムの有効期間の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'date' }"/>
```

## 詳細

バリデーションID date

引数 なし

出力メッセージ {0} - {1}の範囲で入力してください。  
 {0} : システム有効期間開始日  
 {1} : システム有効期間終了日

### コラム

システムの有効期間（初期値）は次の通りです。

- システムの有効期間開始日：1900/01/01
- システムの有効期間終了日：2999/12/31

## 最小日付

### 概要

本バリデーションが設定されたフィールドの値が、「最小日付 <= 入力値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小日付 <= 入力値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小日付 <= 入力値」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id':'dateMin', 'args':'1900/01/01' }"/>
```

### 詳細

バリデーションID dateMin

#### 引数

1. 最小日付（必須）  
（最小日付を保持するフィールドを設定することも可能です。）

出力メッセージ {0}以上で入力してください。  
 {0} : 引数で設定された最小日付を出力します。

### 注意

最小日付、入力値の両方が日付変換できない場合はチェックを実施しません。  
 日付型のチェックは含まれないため、他のチェック（日付）と組み合わせて使用してください。

## 最小日付(等号選択)

### 概要

本バリデーションが設定されたフィールドの値が、「最小日付 <= （または <） 入力値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小日付 <= （または <） 入力値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小日付 <= （または <） 入力値」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id':'dateMin2', 'args':['1900/01/01','true'] }"/>
```

### 詳細

バリデーションID dateMin2

#### 引数（配列）

1. 最小日付（必須）  
（最小日付を保持するフィールドを設定することも可能です。）
2. 等号有無（任意）:省略時は「false」
  - true : 最小日付 <= 入力値
  - false : 最小日付 < 入力値

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以上で入力してください。
- {0}より大きい値で入力してください。

{0} : 引数で設定された最小日付を出力します。

### 注意

最小日付、入力値の両方が日付変換できない場合はチェックを実施しません。  
日付型のチェックは含まれないため、他のチェック（日付）と組み合わせて使用してください。

## 最大日付

### 概要

本バリデーションが設定されたフィールドの値が、「入力値 <= 最大日付」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「入力値 <= 最大日付」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「入力値 <= 最大日付」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id':'dateMax', 'args':{'2999/12/31'}}"/>
```

### 詳細

バリデーションID dateMax

引数

1. 最大日付（必須）  
(最大日付を保持するフィールドを設定することも可能です。)

出力メッセージ

{0}以下で入力してください。  
{0} : 引数で設定された最大日付を出力します。

### 注意

最大日付、入力値の両方が日付変換できない場合はチェックを実施しません。  
日付型のチェックは含まれないため、他のチェック（日付）と組み合わせて使用してください。

## 最大日付(等号選択)

### 概要

本バリデーションが設定されたフィールドの値が、「入力値 <= (または <) 最大日付」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「入力値 <= (または <) 最大日付」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「入力値 <= (または <) 最大日付」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id':'dateMax2', 'args':['2999/12/31','true'] }"/>
```

### 詳細

バリデーションID dateMax2

引数（配列）

1. 最大日付（必須）  
(最大日付を保持するフィールドを設定することも可能です。)
2. 等号有無（任意）:省略時は「false」
  - true : 入力値 <= 最大日付
  - false : 入力値 < 最大日付

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以下で入力してください。
- {0}未満で入力してください。

{0} : 引数で設定された最大日付を出力します。

### 注意

最大日付、入力値の両方が日付変換できない場合はチェックを実施しません。  
日付型のチェックは含まれないため、他のチェック（日付）と組み合わせて使用してください。

## 日付範囲

### 概要

本バリデーションが設定されたフィールドの値が、「最小日付 <= 入力値 <= 最大日付」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小日付 <= 入力値 <= 最大日付」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小日付 <= 入力値 <= 最大日付」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id': 'dateRange', 'args': ['1900/01/01', '2999/12/31'] }"/>
```

### 詳細

バリデーションID dateRange

引数（配列）

1. 最小日付（必須）  
（最小日付を保持するフィールドを設定することも可能です。）
2. 最大日付（必須）  
（最大日付を保持するフィールドを設定することも可能です。）

出力メッセージ {0} - {1}の範囲で入力してください。  
{0} : 引数で設定された最小日付を出力します。  
{1} : 引数で設定された最大日付を出力します。

### 注意

最小日付、最大日付、入力値のすべてが日付変換できない場合はチェックを実施しません。  
日付型のチェックは含まれないため、他のチェック（日付）と組み合わせて使用してください。

## 日付範囲(等号選択)

### 概要

本バリデーションが設定されたフィールドの値が、「最小日付 <=（または <） 入力値 <=（または <） 最大日付」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小日付 <=（または <） 入力値 <=（または <） 最大日付」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小日付 <=（または <） 入力値 <=（または <） 最大日付」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id': 'dateRange2', 'args': ['1900/01/01', '2999/12/31', 'true', 'true'] }"/>
```

### 詳細

バリデーションID dateRange2

## 引数 (配列)

1. 最小日付 (必須)  
(最小日付を保持するフィールドを設定することも可能です。)
2. 最大日付 (必須)  
(最大日付を保持するフィールドを設定することも可能です。)
3. 等号有無 1 (任意):省略時は「false」
  - true : 最小日付 <= 入力値
  - false : 最小日付 < 入力値
4. 等号有無 2 (任意):省略時は「false」
  - true : 入力値 <= 最大日付
  - false : 入力値 < 最大日付

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以上、{1}以下の範囲で入力してください。
- {0}以上、{1}未満の範囲で入力してください。
- {0}超、{1}以下の範囲で入力してください。
- {0}超、{1}未満の範囲で入力してください。

{0} : 引数で設定された最小日付を出力します。

{1} : 引数で設定された最大日付を出力します。



## 注意

最小日付、最大日付、入力値のすべてが日付変換できない場合はチェックを実施しません。  
日付型のチェックは含まれないため、他のチェック（日付）と組み合わせて使用してください。

## 時刻(形式)

## 概要

本バリデーションが設定されたフィールドの値が、時刻形式(H:mm)で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が時刻形式で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が時刻形式で入力されていない。

## 設定方法

```
<validate event="default" config="{id:'time'}"/>
```

## 詳細

バリデーションID time

引数 なし

出力メッセージ 時刻の形式で入力してください。



## コラム

24時以降(99:59)と入力された場合でもチェックNGにはなりません。

## 最小時刻

## 概要

本バリデーションが設定されたフィールドの値が、「最小時刻 <= 入力値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小時刻 <= 入力値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小時刻 <= 入力値」の範囲を超えて入力されている。

## 設定方法

```
<validate event="default" config="{ 'id': 'timeMin', 'args': '00:00' }"/>
```

## 詳細

バリデーションID timeMin

### 引数

1. 最小時刻（必須）  
(最小時刻を保持するフィールドを設定することも可能です。)

出力メッセージ {0}以上で入力してください。  
{0} : 引数で設定された最小時刻を出力します。

### 注意

最小時刻、入力値の両方が時刻変換できない場合はチェックを実施しません。  
時刻型のチェックは含まれないため、他のチェック（時刻）と組み合わせて使用してください。

## 最小時刻(等号選択)

### 概要

本バリデーションが設定されたフィールドの値が、「最小時刻 <=（または <） 入力値」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小時刻 <=（または <） 入力値」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小時刻 <=（または <） 入力値」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id': 'timeMin2', 'args': ['00:00', 'true'] }"/>
```

## 詳細

バリデーションID timeMin2

### 引数（配列）

1. 最小時刻（必須）  
(最小時刻を保持するフィールドを設定することも可能です。)
2. 等号有無（任意）:省略時は「false」
  - true : 最小時刻 <= 入力値
  - false : 最小時刻 < 入力値

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以上で入力してください。
- {0}より大きい値で入力してください。

{0} : 引数で設定された最小時刻を出力します。

### 注意

最小時刻、入力値の両方が時刻変換できない場合はチェックを実施しません。  
時刻型のチェックは含まれないため、他のチェック（時刻）と組み合わせて使用してください。

## 最大時刻

### 概要

本バリデーションが設定されたフィールドの値が、「入力値 <= 最大時刻」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「入力値 <= 最大時刻」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「入力値 <= 最大時刻」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id':'timeMax', 'args':{'24:00'}}"/>
```

## 詳細

バリデーションID timeMax

### 引数

1. 最大時刻（必須）  
(最大時刻を保持するフィールドを設定することも可能です。)

出力メッセージ {0}以下で入力してください。  
{0}：引数で設定された最大時刻を出力します。

### 注意

最大時刻、入力値の両方が時刻変換できない場合はチェックを実施しません。  
時刻型のチェックは含まれないため、他のチェック（時刻）と組み合わせて使用してください。

## 最大時刻(等号選択)

### 概要

本バリデーションが設定されたフィールドの値が、「入力値 <=（または <） 最大時刻」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「入力値 <=（または <） 最大時刻」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「入力値 <=（または <） 最大時刻」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id':'timeMax2', 'args':['24:00','true'] }"/>
```

## 詳細

バリデーションID timeMax2

### 引数（配列）

1. 最大時刻（必須）  
(最大時刻を保持するフィールドを設定することも可能です。)
2. 等号有無（任意）:省略時は「false」
  - true：入力値 <= 最大時刻
  - false：入力値 < 最大時刻

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以下で入力してください。
- {0}未満で入力してください。

{0}：引数で設定された最大時刻を出力します。

### 注意

最大時刻、入力値の両方が時刻変換できない場合はチェックを実施しません。  
時刻型のチェックは含まれないため、他のチェック（時刻）と組み合わせて使用してください。

## 時刻範囲

### 概要

本バリデーションが設定されたフィールドの値が、「最小時刻 <= 入力値 <= 最大時刻」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小時刻 <= 入力値 <= 最大時刻」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小時刻 <= 入力値 <= 最大時刻」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id': 'timeRange', 'args': ['00:00', '24:00'] }"/>
```

## 詳細

バリデーションID timeRange

引数（配列）

1. 最小時刻（必須）  
（最小時刻を保持するフィールドを設定することも可能です。）
2. 最大時刻（必須）  
（最大時刻を保持するフィールドを設定することも可能です。）

出力メッセージ {0} - {1}の範囲で入力してください。  
 {0}：引数で設定された最小時刻を出力します。  
 {1}：引数で設定された最大時刻を出力します。

### 注意

最小時刻、最大時刻、入力値のすべてが時刻変換できない場合はチェックを実施しません。  
 時刻型のチェックは含まれないため、他のチェック（時刻）と組み合わせて使用してください。

## 時刻範囲(等号選択)

### 概要

本バリデーションが設定されたフィールドの値が、「最小時刻 <=（または <）入力値 <=（または <）最大時刻」の範囲で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「最小時刻 <=（または <）入力値 <=（または <）最大時刻」の範囲内で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が「最小時刻 <=（または <）入力値 <=（または <）最大時刻」の範囲を超えて入力されている。

### 設定方法

```
<validate event="default" config="{ 'id': 'timeRange2', 'args': ['00:00', '24:00', 'true', 'true'] }"/>
```

## 詳細

バリデーションID timeRange2

引数（配列）

1. 最小時刻（必須）  
（最小時刻を保持するフィールドを設定することも可能です。）
2. 最大時刻（必須）  
（最大時刻を保持するフィールドを設定することも可能です。）
3. 等号有無1（任意）:省略時は「false」
  - true：最小時刻 <= 入力値
  - false：最小時刻 < 入力値
4. 等号有無2（任意）:省略時は「false」
  - true：入力値 <= 最大時刻
  - false：入力値 < 最大時刻

出力メッセージ 引数の条件で次のメッセージを表示します。

- {0}以上、{1}以下の範囲で入力してください。
- {0}以上、{1}未満の範囲で入力してください。
- {0}超、{1}以下の範囲で入力してください。
- {0}超、{1}未満の範囲で入力してください。

{0}：引数で設定された最小時刻を出力します。  
 {1}：引数で設定された最大時刻を出力します。

**!** 注意

最小時刻、最大時刻、入力値のすべてが時刻変換できない場合はチェックを実施しません。  
時刻型のチェックは含まれないため、他のチェック（時刻）と組み合わせて使用してください。

## 時間入力単位

## 概要

本バリデーションが設定されたフィールドの値が、引数で設定された時刻単位で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が引数で設定された時刻単位で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が引数で設定された時刻単位で入力されていない。

## 設定方法

```
<validate event="default" config="{ 'id':'timeUnit', 'args':'10' }"/>
```

## 詳細

バリデーションID timeUnit

引数

1. 時刻単位（分）（必須）

出力メッセージ 時刻は{0}単位で入力してください。  
{0}：引数で設定された時刻単位を出力します。

**i** コラム

分（mm）に相当する値を数値に変換し、時刻単位で除算した結果、余りがない場合にチェックOKとします。

**!** 注意

入力値が時刻変換できない場合はチェックを実施しません。  
時刻型のチェックは含まれないため、他のチェック（時刻）と組み合わせて使用してください。

## 日時(形式)

## 概要

本バリデーションが設定されたフィールドの値が、指定されたフォーマットの日時形式で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が指定されたフォーマットの日時形式で入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値が指定されたフォーマットの日時形式で入力されていない。

## 設定方法

```
<validate event="default" config="{ 'id':'datetime', 'args':'yyyy/MM/dd' }"/>
```

## 詳細

バリデーションID datetime

引数

1. フォーマット（任意）：省略時は「yyyy/MM/dd」

出力メッセージ {0}の形式で入力してください。  
{0}：引数で設定されたフォーマットを出力します。

## 正規表現

## 概要

本バリデーションが設定されたフィールドの値が、設定された正規表現で入力されているか否かを判定します。

- **チェックOK**

本バリデーションが設定されたフィールドの値が設定された正規表現に一致する。

- **チェックNG**

本バリデーションが設定されたフィールドの値が設定された正規表現に一致しない。

#### 設定方法

```
<validate event="default" config="{ 'id': 'regularExpression', 'args': [ '^ [ a - z A - Z 0 - 9 - / : - @ \ [ - \ ] \ { - \ } + $ ', 'message' ] }"/>
```

#### 詳細

バリデーションID regularExpression

引数（配列）

1. 正規表現（必須）
2. メッセージID（必須）

出力メッセージ メッセージIDに紐付く値

### 不一致

#### 概要

判定対象1と判定対象2に設定されたフィールドの値が異なる値か否かを判定します。

- **チェックOK**

判定対象1と判定対象2に設定されたフィールドの値が一致しない。

- **チェックNG**

判定対象1と判定対象2に設定されたフィールドの値が一致する。

#### 設定方法

```
<validate event="default" config="{ 'id': 'notEqual', 'args': [ 'targetField1', 'targetField2' ] }"/>
```

#### 詳細

バリデーションID notEqual

引数（配列）

1. 判定対象1（必須）
2. 判定対象2（必須）

出力メッセージ {0}と{1}は同一の設定はできません。  
 {0}：判定対象1に設定されたフィールドの名称を出力します。  
 {1}：判定対象2に設定されたフィールドの名称を出力します。

### 一致

#### 概要

判定対象1と判定対象2に設定されたフィールドの値が一致するか否かを判定します。

- **チェックOK**

判定対象1と判定対象2に設定されたフィールドの値が一致する。

- **チェックNG**

判定対象1と判定対象2に設定されたフィールドの値が一致しない。

#### 設定方法

```
<validate event="default" config="{ 'id': 'equal', 'args': [ 'targetField1', 'targetField2' ] }"/>
```

#### 詳細

バリデーションID equal

引数（配列）

1. 判定対象1（必須）
2. 判定対象2（必須）

出力メッセージ {0}は{1}と同字値で入力してください。  
 {0}：判定対象1に設定されたフィールドの名称を出力します。  
 {1}：判定対象2に設定されたフィールドの名称を出力します。

## FB文字

### 概要

本バリデーションが設定されたフィールドの値が、「振込データで使用可能な文字」で入力されているか否かを判定します。

- **チェックOK**  
本バリデーションが設定されたフィールドの値が「振込データで使用可能な文字」のみで入力されている。
- **チェックNG**  
本バリデーションが設定されたフィールドの値に「振込データで使用可能な文字」以外の文字が含まれている。

### 設定方法

```
<validate event="default" config="{ 'id':'fbChar' }"/>
```

### 詳細

バリデーションID fbChar

引数 なし

出力メッセージ 入力できない文字が含まれています。



#### コラム

振込データで使用可能な文字は次の通りです。

- カナ：アからン（半角）※小文字含まない
- 英字：AからZ（半角）
- 数字：0から9（半角）
- 濁点・半濁点：` `（半角）
- 記号：（）「」-/. スペース（半角）

## マスタ存在(MasterBuilder)

### 概要

設定された条件のデータがマスタに存在しているか否かを判定します。

- **チェックOK**  
設定された条件のデータがマスタに存在している。
- **チェックNG**  
設定された条件のデータがマスタに存在していない、または複数存在する。

### 設定方法

```
<validate event="default" config="{ 'id':'masterBuilderExists', 'args':{'masterId':'exp_type', 'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany', 'expTypeCd':'expTypeCd'} }"/>
```

### 詳細

バリデーションID masterBuilderExists

引数 (JSON) JSON形式で検索条件を設定します。

出力メッセージ 設定された{0}は存在しません。  
 {0}：マスタ名を出力します。



#### コラム

JSON形式の検索条件は、各マスタに合わせて変更してください。

## マスタ非存在(MasterBuilder)

### 概要

設定された条件のデータがマスタに存在しているか否かを判定します。

- **チェックOK**  
設定された条件のデータがマスタに存在していない。
- **チェックNG**  
設定された条件のデータがマスタに存在している。

#### 設定方法

```
<validate event="default" config="{ 'id':'masterBuilderNotExists', 'args':{'masterId':'exp_type', 'extensionId':'expenseDetail', 'searchCriteriaDate':'imwApplyBaseDate', 'searchCriteriaCompany':'searchCriteriaCompany', 'expTypeCd':'expTypeCd'}}"/>
```

#### 詳細

バリデーションID masterBuilderNotExists

引数 (JSON) JSON形式で検索条件を設定します。

出力メッセージ 指定された{0}は存在します。  
{0} : マスタ名を出力します。

#### コラム

JSON形式の検索条件は、各マスタに合わせて変更してください。

## LogicDesigner

#### 概要

LogicDesignerを使用し判定します。

- **チェックOK**  
LogicDesignerからの返却値 (status) に"true"が設定されている。
- **チェックNG**  
LogicDesignerからの返却値 (status) に"true"以外が設定されている。

#### 設定方法

```
<validate event="default" config="{ 'id':'logicDesignerVal', 'args':['logicFlowId':'flow01',...]}"/>
```

#### 詳細

バリデーションID logicDesignerVal

引数 (配列)

1. LogicFlowID (必須)
2. LogicDesignerへの連携値 (以降は任意設定)

出力メッセージ LogicDesignerからの返却値 (message) に設定された値

#### コラム

LogicDesignerへの連携値は、各LogicFlowに合わせて変更してください。

## 同一日同一経路

#### 概要

旅程明細ガジェットで、旅程日付・出発地・到着経路が一致する自身が過去に申請した精算申請書が存在するか否かを判定します。また、同一精算申請書内に旅程日付・出発地・到着経路が一致する行があるか否かを判定します。

- **チェックOK**  
旅程日付・出発地・到着経路が一致する自身が過去に申請した精算申請書がない場合。  
かつ、同一ガジェット内に旅程日付・出発地・到着経路が一致する行がない場合。
- **チェックNG**  
旅程日付・出発地・到着経路が一致する自身が過去に申請した精算申請書がある場合。  
または、同一ガジェット内に旅程日付・出発地・到着経路が一致する行がある場合。

#### 設定方法

```
<validate event="default" config="{ 'id':'sameDateAndRoute', 'args':['routeDate','startingSpot','arrivalSpot','appType']}"/>
```

バリデーションID sameDateAndRoute

## 引数 (配列)

1. 旅程日付 (必須)  
(旅程日付を保持するフィールドを設定することも可能です。)
2. 出発地 (必須)  
(出発地を保持するフィールドを設定することも可能です。)
3. 到着地 (必須)  
(到着地を保持するフィールドを設定することも可能です。)
4. 申請書タイプ (必須)  
(申請書タイプを保持するフィールドを設定することも可能です。)

出力メッセージ 「対象の日付に同じ経路 ({0} - {1}) がすでに申請されています。」  
または、「申請書内に日付・経路 ({0} - {1}) の同じ明細が存在します。」  
{0}: 出発地に設定されたフィールドの値を出力します。  
{1}: 到着地に設定されたフィールドの値を出力します。



## コラム

アーカイブ済の申請書に、旅程日付・出発地・到着経路が一致する申請書が存在する場合、チェック対象外のためチェックOKとなります。また、申請書タイプが「精算申請書(仕訳対象)」の申請書のみがチェック対象となります。

## 補完処理

GadgetBuilder定義の「[補完設定](#)」で設定可能な補完処理について説明します。

- [intra-mart Accel Kaiden!標準の補完処理](#)
  - [固定値補完 \[ type : fixed \]](#)
  - [変数値補完 \[ type : variable \]](#)
  - [計算値補完 \[ type : calc \]](#)
  - [計算値\(加算\)補完 \[ type : calcAdd \]](#)
  - [計算値\(減算\)補完 \[ type : calcSub \]](#)
  - [計算値\(乗算\)補完 \[ type : calcMul \]](#)
  - [計算値\(除算\)補完 \[ type : calcDiv \]](#)
  - [税額補完 \[ type : calcTax \]](#)
  - [値補完 \(ガジェット間通信\) \[ type : claimValue \]](#)
  - [値補完 \(ガジェット間通信\) \(初期表示時\) \[ type : claimValueOnInit \]](#)
  - [値補完 \(ガジェット間通信\) \(行追加時\) \[ type : claimValueAddRow \]](#)
  - [合計値補完 \(ガジェット間通信\) \[ type : claimSummary \]](#)
  - [合計値補完 \(ガジェット間通信\) \(クリック時\) \[ type : claimSummaryOnClick \]](#)
  - [サーフィス補完 \(ガジェット間通信\) \[ type : claimSurface \]](#)
  - [会社通貨コード補完 \[ type : companyCurrencyCode \]](#)
  - [DetailBlock変数値補完 \[ type : variableForDetailBlock \]](#)
  - [NormalBlock変数値補完 \[ type : variableForNormalBlock \]](#)
  - [BaseSettings設定値補完 \[ type : baseSettings \]](#)
  - [レートを補完\(変更時\) \[ type : companyRateOnChange \]](#)
  - [指定した通貨換算コードに基づきレートを補完\(変更時\) \[ type : companyRateUseConvOnChange \]](#)
  - [国際化対応を補完\(変更時\) \[ type : locales \]](#)

## intra-mart Accel Kaiden!標準の補完処理

本項では、intra-mart Accel Kaiden!で提供している補完処理と設定方法を説明します。

## 固定値補完 [ type : fixed ]

## 概要

対象フィールドに固定値を補完します。

## 引数

キー	設定値
value	固定値

**設定方法 (例)**

「targetField」フィールドに固定値「1」を補完

```
<completion name="targetField" position="1" type="fixed" args="{ 'value':'1' }"/>
```

## 変数値補完 [ type : variable ]

**概要**

対象フィールドに変数値（フィールドの値）を補完します。

**引数**

キー	設定値
field	補完値を保持するフィールド名

**設定方法 (例)**

「targetField」フィールドに「variableField」フィールドの値を補完

```
<completion name="targetField" position="1" type="variable" args="{ 'field':'variableField' }"/>
```

## 計算値補完 [ type : calc ]

**概要**

対象フィールドに計算値（加算・減算・乗算・除算）を補完します。

**引数**

キー	設定値
field1	計算値を保持するフィールド名1
field2	計算値を保持するフィールド名2
calc	計算方法
round	丸めモード（未設定時「4」）
digits	丸め桁数(小数点以下何桁までを有効とするか) 未設定時:0
scale	計算時の桁数(小数点以下何桁までを有効とするか) 未指定時:10
brank	計算が正しく行われなかった時に結果を空白にするか 未指定時:false

- **ADD** : field1 + field2
- **SUB** : field1 - field2
- **MUL** : field1 \* field2
- **DIV** : field1 / field2

- **0** : ROUND\_UP (切り上げ)
- **1** : ROUND\_DOWN (切り捨て)
- **2** : ROUND\_CEILING (正の無限大に近づくように丸める)
- **3** : ROUND\_FLOOR (負の無限大に近づくように丸める)
- **4** : ROUND\_HALF\_UP (四捨五入)
- **5** : ROUND\_HALF\_DOWN (五捨六入)
- **6** : ROUND\_HALF\_EVEN (末尾が偶数のほうに丸める)
- **7** : ROUND\_UNNECESSARY (丸めなし)

- **true** : 結果を空白にする
- **false** : 結果を「0」にする

**設定方法 (例)**

「targetField」フィールドに「calcField1」フィールド値と「calcField2」フィールド値の計算結果（加算）を補完

(補完する計算結果は小数点以下0桁まで有効で四捨五入)

```
<completion name="targetField" position="1" type="calc" args="{
'field1':'calcField1','field2':'calcField2','calc':'ADD','round':'4','digits':'0','scale':'10','blank':'true'}"/>
```

## 計算値(加算)補完 [ type : calcAdd ]

### 概要

対象フィールドに計算値（加算）を補完します。

### 引数

キー	設定値
field1	計算値を保持するフィールド名1
field2	計算値を保持するフィールド名2
round	丸めモード（未設定時「4」） <ul style="list-style-type: none"> <li>▪ <b>0</b> : ROUND_UP（切り上げ）</li> <li>▪ <b>1</b> : ROUND_DOWN（切り捨て）</li> <li>▪ <b>2</b> : ROUND_CEILING（正の無限大に近づくように丸める）</li> <li>▪ <b>3</b> : ROUND_FLOOR（負の無限大に近づくように丸める）</li> <li>▪ <b>4</b> : ROUND_HALF_UP（四捨五入）</li> <li>▪ <b>5</b> : ROUND_HALF_DOWN（五捨六入）</li> <li>▪ <b>6</b> : ROUND_HALF_EVEN（末尾が偶数のほうに丸める）</li> <li>▪ <b>7</b> : ROUND_UNNECESSARY（丸めなし）</li> </ul>
digits	丸め桁数(小数点以下何桁までを有効とするか) 未設定時:0
scale	計算時の桁数(小数点以下何桁までを有効とするか) 未指定時:10
brank	計算が正しく行われなかった時に結果を空白にするか 未指定時:false <ul style="list-style-type: none"> <li>▪ <b>true</b> : 結果を空白にする</li> <li>▪ <b>false</b> : 結果を「0」にする</li> </ul>

### 設定方法例

「targetField」フィールドに「calcField1」フィールド値と「calcField2」フィールド値の計算結果（加算）を補完（補完する計算結果は小数点以下0桁まで有効で四捨五入）

```
<completion name="targetField" position="1" type="calcAdd" args="{
'field1':'calcField1','field2':'calcField2','round':'4','digits':'0','scale':'10','blank':'true'}"/>
```

## 計算値(減算)補完 [ type : calcSub ]

### 概要

対象フィールドに計算値（減算）を補完します。

### 引数

キー	設定値
field1	計算値を保持するフィールド名1
field2	計算値を保持するフィールド名2

round	丸めモード (未設定時「4」)
	<ul style="list-style-type: none"> <li>▪ <b>0</b>: ROUND_UP (切り上げ)</li> <li>▪ <b>1</b>: ROUND_DOWN (切り捨て)</li> <li>▪ <b>2</b>: ROUND_CEILING (正の無限大に近づくように丸める)</li> <li>▪ <b>3</b>: ROUND_FLOOR (負の無限大に近づくように丸める)</li> <li>▪ <b>4</b>: ROUND_HALF_UP (四捨五入)</li> <li>▪ <b>5</b>: ROUND_HALF_DOWN (五捨六入)</li> <li>▪ <b>6</b>: ROUND_HALF_EVEN (末尾が偶数のほうに丸める)</li> <li>▪ <b>7</b>: ROUND_UNNECESSARY (丸めなし)</li> </ul>
digits	丸め桁数(小数点以下何桁までを有効とするか) 未設定時:0
scale	計算時の桁数(小数点以下何桁までを有効とするか) 未指定時:10
brank	計算が正しく行われなかった時に結果を空白にするか 未指定時:false
	<ul style="list-style-type: none"> <li>▪ <b>true</b>: 結果を空白にする</li> <li>▪ <b>false</b>: 結果を「0」にする</li> </ul>

**設定方法 (例)**

「targetField」フィールドに「calcField1」フィールド値と「calcField2」フィールド値の計算結果 (減算) を補完 (補完する計算結果は小数点以下0桁まで有効で四捨五入)

```
<completion name="targetField" position="1" type="calcSub" args="{
'field1':'calcField1','field2':'calcField2','round':'4','digits':'0','scale':10,'blank':'true'}"/>
```

## 計算値(乗算)補完 [ type : calcMul ]

**概要**

対象フィールドに計算値 (乗算) を補完します。

**引数**

キー	設定値
field1	計算値を保持するフィールド名1
field2	計算値を保持するフィールド名2
round	丸めモード (未設定時「4」)
	<ul style="list-style-type: none"> <li>▪ <b>0</b>: ROUND_UP (切り上げ)</li> <li>▪ <b>1</b>: ROUND_DOWN (切り捨て)</li> <li>▪ <b>2</b>: ROUND_CEILING (正の無限大に近づくように丸める)</li> <li>▪ <b>3</b>: ROUND_FLOOR (負の無限大に近づくように丸める)</li> <li>▪ <b>4</b>: ROUND_HALF_UP (四捨五入)</li> <li>▪ <b>5</b>: ROUND_HALF_DOWN (五捨六入)</li> <li>▪ <b>6</b>: ROUND_HALF_EVEN (末尾が偶数のほうに丸める)</li> <li>▪ <b>7</b>: ROUND_UNNECESSARY (丸めなし)</li> </ul>
digits	丸め桁数(小数点以下何桁までを有効とするか) 未設定時:0
scale	計算時の桁数(小数点以下何桁までを有効とするか) 未指定時:10
brank	計算が正しく行われなかった時に結果を空白にするか 未指定時:false
	<ul style="list-style-type: none"> <li>▪ <b>true</b>: 結果を空白にする</li> <li>▪ <b>false</b>: 結果を「0」にする</li> </ul>

**設定方法 (例)**

「targetField」フィールドに「calcField1」フィールド値と「calcField2」フィールド値の計算結果 (乗算) を補完 (補完する計算結果は小数点以下0桁まで有効で四捨五入)

```
<completion name="targetField" position="1" type="calcMul" args="{field1:'calcField1',field2:'calcField2',round:'4','digits':'0','scale':'10','blank':'true'}"/>
```

## 計算値(除算)補完 [ type : calcDiv ]

### 概要

対象フィールドに計算値（除算）を補完します。

### 引数

キー	設定値
field1	計算値を保持するフィールド名1
field2	計算値を保持するフィールド名2
round	丸めモード（未設定時「4」） <ul style="list-style-type: none"> <li>▪ <b>0</b> : ROUND_UP（切り上げ）</li> <li>▪ <b>1</b> : ROUND_DOWN（切り捨て）</li> <li>▪ <b>2</b> : ROUND_CEILING（正の無限大に近づくように丸める）</li> <li>▪ <b>3</b> : ROUND_FLOOR（負の無限大に近づくように丸める）</li> <li>▪ <b>4</b> : ROUND_HALF_UP（四捨五入）</li> <li>▪ <b>5</b> : ROUND_HALF_DOWN（五捨六入）</li> <li>▪ <b>6</b> : ROUND_HALF_EVEN（末尾が偶数のほうに丸める）</li> <li>▪ <b>7</b> : ROUND_UNNECESSARY（丸めなし）</li> </ul>
digits	丸め桁数(小数点以下何桁までを有効とするか) 未設定時:0
scale	計算時の桁数(小数点以下何桁までを有効とするか) 未指定時:10
brank	計算が正しく行われなかった時に結果を空白にするか 未指定時:false <ul style="list-style-type: none"> <li>▪ <b>true</b> : 結果を空白にする</li> <li>▪ <b>false</b> : 結果を「0」にする</li> </ul>

### 設定方法（例）

「targetField」フィールドに「calcField1」フィールド値と「calcField2」フィールド値の計算結果（除算）を補完（補完する計算結果は小数点以下0桁まで有効で四捨五入）

```
<completion name="targetField" position="1" type="calcDiv" args="{field1:'calcField1',field2:'calcField2',round:'4','digits':'0','scale':'10','blank':'true'}"/>
```

## 税額補完 [ type : calcTax ]

### 概要

対象フィールドに税額を補完します。

### 引数

キー	設定値
amount	税込金額を保持するフィールド名
taxRate	税率を保持するフィールド名
round	丸めモード（未設定時「4」） <ul style="list-style-type: none"> <li>▪ <b>0</b> : ROUND_UP（切り上げ）</li> <li>▪ <b>1</b> : ROUND_DOWN（切り捨て）</li> <li>▪ <b>2</b> : ROUND_CEILING（正の無限大に近づくように丸める）</li> <li>▪ <b>3</b> : ROUND_FLOOR（負の無限大に近づくように丸める）</li> <li>▪ <b>4</b> : ROUND_HALF_UP（四捨五入）</li> <li>▪ <b>5</b> : ROUND_HALF_DOWN（五捨六入）</li> <li>▪ <b>6</b> : ROUND_HALF_EVEN（末尾が偶数のほうに丸める）</li> <li>▪ <b>7</b> : ROUND_UNNECESSARY（丸めなし）</li> </ul>

digits	丸め桁数(小数点以下何桁までを有効とするか) 未設定時:0
scale	計算時の桁数(小数点以下何桁までを有効とするか) 未指定時:10

**設定方法 (例)**

「targetField」フィールドに「amountField」フィールド値と「taxRateField」フィールド値の計算結果（税額）を補完（補完する計算結果は小数点以下0桁まで有効で四捨五入）

```
<completion name="targetField" position="1" type="calcTax" args="{ 'amount':'amountField','taxRate':'taxRateField','round':'1','digits':'0','scale':'10' }"/>
```

**値補完 (ガジェット間通信) [ type : claimValue ]****概要**

対象フィールドにガジェット間通信で取得した値を補完します。

**引数**

キー	設定値
claim	ガジェット間通信名
target	ガジェット間通信の対象（未設定時「0」） <ul style="list-style-type: none"> <li>▪ <b>0</b> : 全ガジェット対象</li> <li>▪ <b>1</b> : 自身のガジェットのみ対象</li> <li>▪ <b>2</b> : 自身のガジェットを除く全ガジェット対象</li> </ul>

**設定方法 (例)**

「targetField」フィールドに全ガジェット分のガジェット間通信（claimValue）結果を補完（最初に取得できた値）

```
<completion name="targetField" position="1" type="claimValue" args="{ 'claim':'claimValue','target':'0' }"/>
```

**値補完 (ガジェット間通信) (初期表示時) [ type : claimValueOnInit ]****概要**

初期表示時のみ、対象フィールドにガジェット間通信で取得した値を補完します。

**引数**

キー	設定値
claim	ガジェット間通信名
target	ガジェット間通信の対象（未設定時「0」） <ul style="list-style-type: none"> <li>▪ <b>0</b> : 全ガジェット対象</li> <li>▪ <b>1</b> : 自身のガジェットのみ対象</li> <li>▪ <b>2</b> : 自身のガジェットを除く全ガジェット対象</li> </ul>

**設定方法 (例)**

初期表示時、「targetField」フィールドに全ガジェット分のガジェット間通信（claimValue）結果を補完（最初に取得できた値）

```
<completion name="targetField" position="1" type="claimValueOnInit" args="{ 'claim':'claimValue','target':'0' }"/>
```

**値補完 (ガジェット間通信) (行追加時) [ type : claimValueAddRow ]****概要**

行追加時に、対象フィールドにガジェット間通信で取得した値を補完します。

**引数**

キー	設定値
claim	ガジェット間通信名

target    ガジェット間通信の対象（未設定時「0」）

- **0** : 全ガジェット対象
- **1** : 自身のガジェットのみ対象
- **2** : 自身のガジェットを除く全ガジェット対象)

#### 設定方法（例）

行追加時、「targetField」フィールドに全ガジェット分のガジェット間通信（claimValue）結果を補完（最初に取得できた値）

```
<completion name="targetField" position="1" type="claimValueAddRow" args="{ 'claim': 'claimValue', 'target': '0' }"/>
```

### 合計値補完（ガジェット間通信） [ type : claimSummary ]

#### 概要

対象フィールドにガジェット間通信の返却値合計を補完します。

#### 引数

キー	設定値
claim	ガジェット間通信名
target	ガジェット間通信の対象（未設定時「0」） <ul style="list-style-type: none"> <li>▪ <b>0</b> : 全ガジェット対象</li> <li>▪ <b>1</b> : 自身のガジェットのみ対象</li> <li>▪ <b>2</b> : 自身のガジェットを除く全ガジェット対象)</li> </ul>

#### 設定方法（例）

「targetField」フィールドに全ガジェット分のガジェット間通信（claimSummary）結果の合計値を補完

```
<completion name="targetField" position="1" type="claimSummary" args="{ 'claim': 'claimSummary', 'target': '0' }"/>
```

### 合計値補完（ガジェット間通信）（クリック時） [ type : claimSummaryOnClick ]

#### 概要

設定フィールドのクリック時のみ、対象フィールドにガジェット間通信の返却値合計を補完します。

#### 引数

キー	設定値
claim	ガジェット間通信名
target	ガジェット間通信の対象（未設定時「0」） <ul style="list-style-type: none"> <li>▪ <b>0</b> : 全ガジェット対象</li> <li>▪ <b>1</b> : 自身のガジェットのみ対象</li> <li>▪ <b>2</b> : 自身のガジェットを除く全ガジェット対象)</li> </ul>
field	当補完を実行するonClickフィールド名（未設定の場合は補完しません）

#### 設定方法（例）

「onClickField」フィールドのクリックイベントで「targetField」フィールドに全ガジェット分のガジェット間通信（claimSummary）結果の合計値を補完

```
<completion name="targetField" position="1" type="claimSummary" args="{ 'claim': 'claimSummary', 'target': '0', 'field': 'onClickField' }"/>
```

### サーフィス補完（ガジェット間通信） [ type : claimSurface ]

#### 概要

対象フィールドのサーフィスにガジェット間通信で取得した値を補完します。

#### 引数

キー	設定値
claim	ガジェット間通信名
target	ガジェット間通信の対象（未設定時「0」） <ul style="list-style-type: none"> <li>▪ <b>0</b> : 全ガジェット対象</li> <li>▪ <b>1</b> : 自身のガジェットのみ対象</li> <li>▪ <b>2</b> : 自身のガジェットを除く全ガジェット対象)</li> </ul>

**設定方法（例）**

「targetField」フィールドのサーフィスに全ガジェット分のガジェット間通信（claimSurface）結果を補完（最初に取得できた値）

```
<completion name="targetField" position="1" type="claimSurface" args="{ 'claim': 'claimSurface', 'target': '0' }"/>
```

## 会社通貨コード補完 [ type : companyCurrencyCode ]

**概要**

対象フィールドに会社通貨コードを補完します。

**引数**

キー	設定値
field	会社コードを保持するフィールド名

**設定方法（例）**

「targetField」フィールドに会社通貨コードを補完

```
<completion name="targetField" position="1" type="companyCurrencyCode" args="{ 'field': 'companyCodeField' }"/>
```

## DetailBlock変数値補完 [ type : variableForDetailBlock ]

**概要**

対象フィールドに「DetailBlock」の変数値（フィールドの値）を補完します。

本補完処理は「NormalBlock」のみ使用可能です。（取得する変数値は1行目となります。）

**引数**

キー	設定値
block	補完値を保持するフィールドのブロック名（DetailBlock）
field	補完値を保持するフィールド名

**設定方法（例）**

「targetField」フィールドに「DetailBlock」ブロックの「variableField」フィールドの値を補完（1行目）

```
<completion name="targetField" position="1" type="variableForDetailBlock" args="{ 'block': 'DetailBlock', 'field': 'variableField' }"/>
```

## NormalBlock変数値補完 [ type : variableForNormalBlock ]

**概要**

対象フィールドに「NormalBlock」の変数値（フィールドの値）を補完します。

本補完処理は「DetailBlock」のみ使用可能です。

**引数**

キー	設定値
block	補完値を保持するフィールドのブロック名（NormalBlock）
field	補完値を保持するフィールド名

**設定方法（例）**

「targetField」フィールドに「NormalBlock」ブロックの「variableField」フィールドの値を補完

```
<completion name="targetField" position="1" type="variableForNormalBlock" args="{ 'block':'NormalBlock','field':'variableField' }"/>
```

## BaseSettings設定値補完 [ type : baseSettings ]

### 概要

対象フィールドに「BaseSettings.xml」の設定を補完します。

補完可能な値は『intra-mart Accel Kaiden! セットアップガイド』 - 「warファイルの作成」 - 「intra-mart Accel Kaiden! の設定ファイル」 - 「intra-mart Accel Kaiden! 基盤モジュール」 - 「モジュール設定」を参照ください。

### 引数

キー	設定値
key	取得するキー

### 設定方法 (例)

「targetField」フィールドに「defaultDepartmentSetCd」（「BaseSettings.xml」に設定した組織セットコード）の値を補完

```
<completion name="targetField" position="1" type="baseSettings" args="{ 'key':'defaultDepartmentSetCd' }"/>
```

## レートを補完(変更時) [ type : companyRateOnChange ]

### 概要

指定したフィールドの変更時、対象フィールドにレートを補完します。

レート取得時の通貨換算コードは「KAIDEN」を使用します。

### 引数

キー	設定値
transCryCd	通貨コードを保持するフィールド名
companyCryCd	会社通貨コードを保持するフィールド名
companyCd	会社コードを保持するフィールド名
date	検索基準日を保持するフィールド名
field	当補完を実行するonChangeフィールド名

### 設定方法 (例)

「transCurrencyCd」のonChangeイベント時、「targetField」フィールドにレートの値を補完

```
<completion name="targetField" position="1" type="companyRateOnChange" args="{ 'transCryCd':'transCurrencyCd','companyCryCd':'companyCurrencyCd','companyCd':'searchCriteriaCompany','date':'imwApplyBaseDate','field':
```

## 指定した通貨換算コードに基づきレートを補完(変更時) [ type : companyRateUseConvOnChange ]

### 概要

指定したフィールドの変更時、対象フィールドにレートを補完します。

レート取得時の通貨換算コードを指定可能です。

### 引数

キー	設定値
transCryCd	通貨コードを保持するフィールド名
companyCryCd	会社通貨コードを保持するフィールド名
transCryConvCd	通貨換算コードを保持するフィールド名 (未設定時「KAIDEN」)
companyCd	会社コードを保持するフィールド名
date	検索基準日を保持するフィールド名
field	当補完を実行するonChangeフィールド名

### 設定方法 (例)

「transCurrencyCd」のonChangeイベント時、「targetField」フィールドにレートの値を補完

```
<completion name="targetField" position="1" type="companyRateUseConvOnChange" args="{
  'transCryCd':'transCurrencyCd','companyCryCd':'companyCurrencyCd','transCryConvCd':'transCurrencyConversionCd','companyCd':'searchCriteriaCd'}"/>
```

## 国際化対応を補完(変更時) [ type : locales ]

### 概要

対象フィールドのデフォルトに設定された値を、各言語に補完します。

本補完処理は対象フィールドが国際化対応のテキストボックス、テキストエリアの場合のみ使用可能です。

### 設定方法 (例)

「targetField」フィールドのデフォルトに設定された値を、各言語に補完

```
<completion name="targetField" position="1" type="locales"/>
```

## 通信処理

GadgetBuilder定義の「[通信設定](#)」で設定可能な通信処理について説明します。

- intra-mart Accel Kaiden!標準の通信処理
  - 値返却 [ type : claimColumnValue ]
  - 値返却 (配列) [ type : claimColumnValues ]
  - 値返却 (配列 x 配列) [ type : claimColumnValuesList ]
  - 値返却 (Map) [ type : claimColumnValuesMap ]
  - 値返却 (MapList) [ type : claimColumnValuesMapList ]
  - 行数返却 [ type : claimRowCount ]
  - サーフィス返却 [ type : claimSurface ]
  - 金額合計 [ type : claimAmountSummary ]
  - 検索&結果設定を行うガジェット間通信 [ type : claimSearchData ]

## intra-mart Accel Kaiden!標準の通信処理

本項では、intra-mart Accel Kaiden!で提供している通信処理と設定方法を説明します。

### 値返却 [ type : claimColumnValue ]

#### 概要

設定したフィールドの入力値データを返却するガジェット間通信です。

#### 引数

キー	設定値
field	入力値データを返却するフィールド名 (任意) 未設定の場合は「空白」を返却

#### 設定方法 (例)

```
<claim name="claimName" type="claimColumnValue" args="{field:'valueField'}"/>
```

### 値返却 (配列) [ type : claimColumnValues ]

#### 概要

設定したフィールドの入力値データを配列で返却するガジェット間通信です。

取得対象がTmtableの場合は、1行目のデータのみ返却します。

#### 引数

キー	設定値
----	-----

field1	入力値データを返却するフィールド名（任意） 未設定の場合は「空の配列」を返却
field2	入力値データを返却するフィールド名（任意）
field3	入力値データを返却するフィールド名（任意）
field4	入力値データを返却するフィールド名（任意）
field5	入力値データを返却するフィールド名（任意）

## 設定方法（例）

```
<claim name="claimName" type="claimColumnValues" args="{field1:'valueField1',field2:'valueField2',field3:'valueField3',field4:'valueField4',field5:'valueField5'}"/>
```

## 値返却（配列 x 配列） [ type : claimColumnValuesList ]

## 概要

設定したフィールドの入力値データを配列x配列で返却するガジェット間通信です。

## 引数

## キー 設定値

field1	入力値データを返却するフィールド名（任意） 未設定の場合は「空の配列」を返却
field2	入力値データを返却するフィールド名（任意）
field3	入力値データを返却するフィールド名（任意）
field4	入力値データを返却するフィールド名（任意）
field5	入力値データを返却するフィールド名（任意）

## 設定方法（例）

```
<claim name="claimName" type="claimColumnValuesList" args="{field1:'valueField1',field2:'valueField2',field3:'valueField3',field4:'valueField4',field5:'valueField5'}"/>
```

## 値返却（Map） [ type : claimColumnValuesMap ]

## 概要

設定したフィールドの入力値データをMapで返却するガジェット間通信です。  
取得対象がTmtableの場合は、1行目のデータのみ返却します。

## 引数

複数設定が可能です。

## キー 設定値

Mapキー	入力値データを返却するフィールド名（任意）
-------	-----------------------

## 設定方法（例）

```
<claim name="claimName" type="claimColumnValuesMap" args="{key1:'valueField1',{key2:'valueField2',...}}"/>
```

## 値返却（MapList） [ type : claimColumnValuesMapList ]

## 概要

設定したフィールドの入力値データをMapListで返却するガジェット間通信です。

#### 引数

複数設定が可能です。

キー	設定値
Mapキー	入力値データを返却するフィールド名 (任意)

#### 設定方法 (例)

```
<claim name="claimName" type="claimColumnValuesMapList" args="{ 'key1': 'valueField1', 'key2': 'valueField2', ... }"/>
```

### 行数返却 [ type : claimRowCount ]

#### 概要

行数を返却するガジェット間通信です。

#### 引数

なし

#### 設定方法 (例)

```
<claim name="claimName" type="claimRowCount"/>
```

### サーフィス返却 [ type : claimSurface ]

#### 概要

設定したフィールドのサーフィスを返却するガジェット間通信です。

#### 引数

キー	設定値
field	データを返却するフィールド名 (必須)

#### 設定方法 (例)

```
<claim name="claimName" type="claimSurface" args="{ 'field': 'surfaceField' }"/>
```

### 金額合計 [ type : claimAmountSummary ]

#### 概要

金額合計を返却するガジェット間通信です。

#### 引数

キー	設定値
amount	金額のフィールド名 (任意) 未設定の場合は「0」を返却
paymentFlag	立替フラグのフィールド名 未設定の場合は金額のフィールドの合計
payment	合計する立替フラグのコード 未設定の場合は金額のフィールドの合計

#### 設定方法 (例)

```
<claim name="claimName" type="claimAmountSummary" args="{ 'amount': 'amountField', 'paymentFlag': 'paymentFlagField', 'payment': '0' }"/>
```

### 検索&結果設定を行うガジェット間通信 [ type : claimSearchData ]

#### 概要

検索&結果設定を行うガジェット間通信です。

#### 引数

複数設定が可能です。

キー	設定値
----	-----

検索キー	データを返却するフィールド名 (任意)
------	---------------------

設定方法 (例)

```
<claim name="claimName" type="claimSearchData" args="{ 'key1':!valueField1!({, 'key2':!valueField2!,...}) }"/>
```

## イベント処理

GadgetBuilder定義の「[イベント設定](#)」で設定可能なイベント処理について説明します。

- イベントキー
  - 共通イベント
  - 申請書ガジェット用イベント
- intra-mart Accel Kaiden!標準のイベント処理
  - IM-LogicDesigner起動イベント [ name : logicDesignerCallEvent ]
  - 案件プロパティ設定イベント [ name : setMatterPropertyEvent ]
  - 仕訳ヘッダーデータ登録イベント (Mapping) [ name : journalSourceHeaderEntryForMappingEvent ]
  - 仕訳ヘッダーデータ登録イベント (SQL) [ name : journalSourceHeaderEntryForSqlEvent ]
  - 仕訳明細データ登録イベント (Mapping) [ name : journalSourceDetailEntryForMappingEvent ]
  - 仕訳明細データ登録イベント (SQL) [ name : journalSourceDetailEntryForSqlEvent ]
  - 仮払残高データ処理イベント [ name : advanceBalanceEvent ]
  - 仮払精算データ処理イベント [ name : advanceSettleEvent ]
  - 法人カードデータ利用状況登録イベント [ name : searchCorporateCardEvent ]
  - ICカードデータ利用状況登録イベント [ name : searchTransitManagerEvent ]

## イベントキー

本項では、イベント処理を設定する際のイベントキーを説明します。

### 共通イベント

共通イベントとして次の値が使用可能です。

イベント名	イベントキー
常時	always
登録処理時	insert
更新処理時	update
削除処理時	delete

### 申請書ガジェット用イベント

申請書ガジェットの場合は次の値が使用可能です。

イベント名	イベントキー
申請処理	apply
一時保存案件の申請処理	apply_from_temp_save
未申請案件の申請処理	apply_from_unapply
承認処理	approve
承認終了処理	approve_end
否認処理	deny
取止め処理	discontinue
引戻し処理	pull_back

再申請処理	reapply
保留処理	reserve
保留解除処理	reserve_cancel
差戻し処理	send_back
差戻し後の引戻し処理	send_back_to_pull_back
一時保存(新規登録)処理	temp_save_create
一時保存(削除)処理	temp_save_delete
一時保存(更新)処理	temp_save_update
案件操作処理	matter_handle
案件開始処理	matter_start
案件終了処理	matter_end
到達処理	arrive
未完了案件削除処理	actv_matter_delete
完了案件削除処理	cpl_matter_delete
過去案件削除処理	arc_matter_delete
案件退避処理	matter_archive

### コラム

申請書ガジェットの場合は上記のイベントの他にバリデート処理時にもイベントを呼び出すことが可能です。

- 個別バリデート処理 ( %上記イベントキー%\_post\_single\_validate)
- 大域バリデート処理 ( %上記イベントキー%\_global\_validate)
- バリデート定義処理 ( %上記イベントキー%\_define\_validate)

## intra-mart Accel Kaiden!標準のイベント処理

本項では、intra-mart Accel Kaiden!で提供しているイベント処理と設定方法を説明します。

### IM-LogicDesigner起動イベント [ name : logicDesignerCallEvent ]

#### 概要

IM-LogicDesigner起動するイベントです。

#### 引数

キー	設定値
logicFlowId	LogicFlowID (必須)

#### 設定方法 (例)

```
<event target="always" name="logicDesignerCallEvent">
  <args>
    <arg key="logicFlowId" value="logicFlow"/>
    <arg key="key1" value="value1"/>
    <arg key="key2" value="value2"/>
    <arg key="key3" value="value3"/>
  </args>
</event>
```

### コラム

LogicDesignerには、次の値を連携します。

- ガジェットデータ (キー:gadgetData, 型:Map<String, Object>)
- ブロックデータ (キー:blockData, 型:Map<String, Object> or List<Map<String, Object>>)
- 引数データ (キー:args, 型:Map<String, Object>)  
(「LogicFlowID」以外にも引数に設定した内容は本データとして連携します。)

## 案件プロパティ設定イベント [ name : setMatterPropertyEvent ]

## 概要

案件プロパティを設定するイベントです。

主に「登録処理」時の利用を想定したイベントです。

引数に明細行のフィールド名を設定した場合は1行目の値を設定します。

## 引数

キー	設定値
propertyKey	案件プロパティキー (必須)
propertyValue	案件プロパティ値、または案件プロパティ値を保持するフィールド名 (必須)

## 設定方法 (例)

```
<event target="insert" name="setMatterPropertyEvent">
  <args>
    <arg key="propertyKey" value="key"/>
    <arg key="propertyValue" value="value"/>
  </args>
</event>
```

## 仕訳ヘッダーデータ登録イベント (Mapping) [ name : journalSourceHeaderEntryForMappingEvent ]

## 概要

ブロックデータを引数の設定に従って、仕訳連携用ヘッダーテーブルに登録するイベントです。

主に「案件終了処理」時の利用を想定したイベントです。

## 引数

キー	設定値
headerDate1	仕訳連携用ヘッダーテーブルのカラム「headerDate1」に登録するデータを保持するフィールド名
headerDate2	仕訳連携用ヘッダーテーブルのカラム「headerDate2」に登録するデータを保持するフィールド名
headerCurrencyCd1	仕訳連携用ヘッダーテーブルのカラム「headerCurrencyCd1」に登録するデータを保持するフィールド名
headerCurrencyCd2	仕訳連携用ヘッダーテーブルのカラム「headerCurrencyCd2」に登録するデータを保持するフィールド名
headerCurrencyCd3	仕訳連携用ヘッダーテーブルのカラム「headerCurrencyCd3」に登録するデータを保持するフィールド名
headerGeneralCd1	仕訳連携用ヘッダーテーブルのカラム「headerGeneralCd1」に登録するデータを保持するフィールド名
headerGeneralCd2	仕訳連携用ヘッダーテーブルのカラム「headerGeneralCd2」に登録するデータを保持するフィールド名
headerGeneralCd3	仕訳連携用ヘッダーテーブルのカラム「headerGeneralCd3」に登録するデータを保持するフィールド名
headerAmount1	仕訳連携用ヘッダーテーブルのカラム「headerAmount1」に登録するデータを保持するフィールド名
headerAmount2	仕訳連携用ヘッダーテーブルのカラム「headerAmount2」に登録するデータを保持するフィールド名
headerAmount3	仕訳連携用ヘッダーテーブルのカラム「headerAmount3」に登録するデータを保持するフィールド名
headerSummary1	仕訳連携用ヘッダーテーブルのカラム「headerSummary1」に登録するデータを保持するフィールド名
headerSummary2	仕訳連携用ヘッダーテーブルのカラム「headerSummary2」に登録するデータを保持するフィールド名
headerSummary3	仕訳連携用ヘッダーテーブルのカラム「headerSummary3」に登録するデータを保持するフィールド名
headerSummary4	仕訳連携用ヘッダーテーブルのカラム「headerSummary4」に登録するデータを保持するフィールド名
headerSummary5	仕訳連携用ヘッダーテーブルのカラム「headerSummary5」に登録するデータを保持するフィールド名
headerNote	仕訳連携用ヘッダーテーブルのカラム「headerNote」に登録するデータを保持するフィールド名
headerType1	仕訳連携用ヘッダーテーブルのカラム「headerType1」に登録するデータを保持するフィールド名
headerType2	仕訳連携用ヘッダーテーブルのカラム「headerType2」に登録するデータを保持するフィールド名
headerType3	仕訳連携用ヘッダーテーブルのカラム「headerType3」に登録するデータを保持するフィールド名
headerGeneralFlag1	仕訳連携用ヘッダーテーブルのカラム「headerGeneralFlag1」に登録するデータを保持するフィールド名
headerGeneralFlag2	仕訳連携用ヘッダーテーブルのカラム「headerGeneralFlag2」に登録するデータを保持するフィールド名

headerGeneralFlag3	仕訳連携用ヘッダテーブルのカラム「headerGeneralFlag3」に登録するデータを保持するフィールド名
headerExtensionA	仕訳連携用ヘッダテーブルのカラム「headerExtensionA」に登録するデータを保持するフィールド名
headerExtensionB	仕訳連携用ヘッダテーブルのカラム「headerExtensionB」に登録するデータを保持するフィールド名
headerExtensionC	仕訳連携用ヘッダテーブルのカラム「headerExtensionC」に登録するデータを保持するフィールド名
headerExtensionD	仕訳連携用ヘッダテーブルのカラム「headerExtensionD」に登録するデータを保持するフィールド名
headerExtensionE	仕訳連携用ヘッダテーブルのカラム「headerExtensionE」に登録するデータを保持するフィールド名
headerExtensionF	仕訳連携用ヘッダテーブルのカラム「headerExtensionF」に登録するデータを保持するフィールド名
headerExtensionG	仕訳連携用ヘッダテーブルのカラム「headerExtensionG」に登録するデータを保持するフィールド名
headerExtensionH	仕訳連携用ヘッダテーブルのカラム「headerExtensionH」に登録するデータを保持するフィールド名
headerExtensionI	仕訳連携用ヘッダテーブルのカラム「headerExtensionI」に登録するデータを保持するフィールド名
headerExtensionJ	仕訳連携用ヘッダテーブルのカラム「headerExtensionJ」に登録するデータを保持するフィールド名
extensionA	仕訳連携用ヘッダテーブルのカラム「extensionA」に登録するデータを保持するフィールド名
extensionB	仕訳連携用ヘッダテーブルのカラム「extensionB」に登録するデータを保持するフィールド名
extensionC	仕訳連携用ヘッダテーブルのカラム「extensionC」に登録するデータを保持するフィールド名
extensionD	仕訳連携用ヘッダテーブルのカラム「extensionD」に登録するデータを保持するフィールド名
extensionE	仕訳連携用ヘッダテーブルのカラム「extensionE」に登録するデータを保持するフィールド名
extensionF	仕訳連携用ヘッダテーブルのカラム「extensionF」に登録するデータを保持するフィールド名
extensionG	仕訳連携用ヘッダテーブルのカラム「extensionG」に登録するデータを保持するフィールド名
extensionH	仕訳連携用ヘッダテーブルのカラム「extensionH」に登録するデータを保持するフィールド名
extensionI	仕訳連携用ヘッダテーブルのカラム「extensionI」に登録するデータを保持するフィールド名
extensionJ	仕訳連携用ヘッダテーブルのカラム「extensionJ」に登録するデータを保持するフィールド名

#### 設定方法（例）

```

<event target="matter_end" name="journalSourceHeaderEntryForMappingEvent">
  <args>
    <arg key="headerDate1" value="headerDate1"/>
    <arg key="headerDate2" value="headerDate2"/>
    <arg key="headerCurrencyCd1" value="headerCurrencyCd1"/>
    <arg key="headerCurrencyCd2" value="headerCurrencyCd2"/>
    <arg key="headerCurrencyCd3" value="headerCurrencyCd3"/>
    <arg key="headerGeneralCd1" value="headerGeneralCd1"/>
    <arg key="headerGeneralCd2" value="headerGeneralCd2"/>
    <arg key="headerGeneralCd3" value="headerGeneralCd3"/>
    <arg key="headerAmount1" value="headerAmount1"/>
    <arg key="headerAmount2" value="headerAmount2"/>
    <arg key="headerAmount3" value="headerAmount3"/>
    <arg key="headerSummary1" value="headerSummary1"/>
    <arg key="headerSummary2" value="headerSummary2"/>
    <arg key="headerSummary3" value="headerSummary3"/>
    <arg key="headerSummary4" value="headerSummary4"/>
    <arg key="headerSummary5" value="headerSummary5"/>
    <arg key="headerNote" value="headerNote"/>
    <arg key="headerType1" value="headerType1"/>
    <arg key="headerType2" value="headerType2"/>
    <arg key="headerType3" value="headerType3"/>
    <arg key="headerGeneralFlag1" value="headerGeneralFlag1"/>
    <arg key="headerGeneralFlag2" value="headerGeneralFlag2"/>
    <arg key="headerGeneralFlag3" value="headerGeneralFlag3"/>
    <arg key="headerExtensionA" value="headerExtensionA"/>
    <arg key="headerExtensionB" value="headerExtensionB"/>
    <arg key="headerExtensionC" value="headerExtensionC"/>
    <arg key="headerExtensionD" value="headerExtensionD"/>
    <arg key="headerExtensionE" value="headerExtensionE"/>
    <arg key="headerExtensionF" value="headerExtensionF"/>
    <arg key="headerExtensionG" value="headerExtensionG"/>
    <arg key="headerExtensionH" value="headerExtensionH"/>
    <arg key="headerExtensionI" value="headerExtensionI"/>
    <arg key="headerExtensionJ" value="headerExtensionJ"/>
    <arg key="extensionA" value="extensionA"/>
    <arg key="extensionB" value="extensionB"/>
    <arg key="extensionC" value="extensionC"/>
    <arg key="extensionD" value="extensionD"/>
    <arg key="extensionE" value="extensionE"/>
    <arg key="extensionF" value="extensionF"/>
    <arg key="extensionG" value="extensionG"/>
    <arg key="extensionH" value="extensionH"/>
    <arg key="extensionI" value="extensionI"/>
    <arg key="extensionJ" value="extensionJ"/>
  </args>
</event>

```

## 仕訳ヘッダーデータ登録イベント (SQL) [ name : journalSourceHeaderEntryForSqlEvent ]

### 概要

ブロックデータを引数のSQLに従って、仕訳連携用ヘッダーテーブルに登録するイベントです。  
主に「案件終了処理」時の利用を想定したイベントです。

### 引数

キー	設定値
sqlPath	仕訳連携用ヘッダーデータ取得用SQL (必須)

### 設定方法 (例)

```

<event target="matter_end" name="journalSourceHeaderEntryForSqlEvent">
  <args>
    <arg key="sqlPath" value="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/journal/Header.sql"/>
  </args>
</event>

```

## 仕訳明細データ登録イベント (Mapping) [ name : journalSourceDetailEntryForMappingEvent ]

### 概要

ブロックデータを引数の設定に従って、仕訳連携用明細テーブルに登録するイベントです。

主に「案件終了処理」時の利用を想定したイベントです。

## 引数

キー	設定値
journalDataTypeCd	仕訳連携用明細テーブルのカラム「journalDataTypeCd」に登録するデータを保持するフィールド名
tupleId	仕訳連携用明細テーブルのカラム「tupleId」に登録するデータを保持するフィールド名
rowNumber	仕訳連携用明細テーブルのカラム「rowNumber」に登録するデータを保持するフィールド名
expTypeCd	仕訳連携用明細テーブルのカラム「expTypeCd」に登録するデータを保持するフィールド名
accountCd	仕訳連携用明細テーブルのカラム「accountCd」に登録するデータを保持するフィールド名
taxTypeCd	仕訳連携用明細テーブルのカラム「taxTypeCd」に登録するデータを保持するフィールド名
taxRate	仕訳連携用明細テーブルのカラム「taxRate」に登録するデータを保持するフィールド名
quantity	仕訳連携用明細テーブルのカラム「quantity」に登録するデータを保持するフィールド名
transCurrencyCd	仕訳連携用明細テーブルのカラム「transCurrencyCd」に登録するデータを保持するフィールド名
transAmount	仕訳連携用明細テーブルのカラム「transAmount」に登録するデータを保持するフィールド名
companyCurrencyCd	仕訳連携用明細テーブルのカラム「companyCurrencyCd」に登録するデータを保持するフィールド名
companyRate	仕訳連携用明細テーブルのカラム「companyRate」に登録するデータを保持するフィールド名
companyAmount	仕訳連携用明細テーブルのカラム「companyAmount」に登録するデータを保持するフィールド名
companyAmountExcTax	仕訳連携用明細テーブルのカラム「companyAmountExcTax」に登録するデータを保持するフィールド名
companyAmountTax	仕訳連携用明細テーブルのカラム「companyAmountTax」に登録するデータを保持するフィールド名
companyAmountSum	仕訳連携用明細テーブルのカラム「companyAmountSum」に登録するデータを保持するフィールド名
customerCd	仕訳連携用明細テーブルのカラム「customerCd」に登録するデータを保持するフィールド名
customerName	仕訳連携用明細テーブルのカラム「customerName」に登録するデータを保持するフィールド名
departmentSetCd	仕訳連携用明細テーブルのカラム「departmentSetCd」に登録するデータを保持するフィールド名
departmentCd	仕訳連携用明細テーブルのカラム「departmentCd」に登録するデータを保持するフィールド名
projectCd	仕訳連携用明細テーブルのカラム「projectCd」に登録するデータを保持するフィールド名
payMethodCd	仕訳連携用明細テーブルのカラム「payMethodCd」に登録するデータを保持するフィールド名
payTypeCd	仕訳連携用明細テーブルのカラム「payTypeCd」に登録するデータを保持するフィールド名
advMatterNumber	仕訳連携用明細テーブルのカラム「advMatterNumber」に登録するデータを保持するフィールド名
advMatterName	仕訳連携用明細テーブルのカラム「advMatterName」に登録するデータを保持するフィールド名
detailDate1	仕訳連携用明細テーブルのカラム「detailDate1」に登録するデータを保持するフィールド名
detailDate2	仕訳連携用明細テーブルのカラム「detailDate2」に登録するデータを保持するフィールド名
detailSummary1	仕訳連携用明細テーブルのカラム「detailSummary1」に登録するデータを保持するフィールド名
detailSummary2	仕訳連携用明細テーブルのカラム「detailSummary2」に登録するデータを保持するフィールド名
detailSummary3	仕訳連携用明細テーブルのカラム「detailSummary3」に登録するデータを保持するフィールド名
detailSummary4	仕訳連携用明細テーブルのカラム「detailSummary4」に登録するデータを保持するフィールド名
detailSummary5	仕訳連携用明細テーブルのカラム「detailSummary5」に登録するデータを保持するフィールド名
detailNote	仕訳連携用明細テーブルのカラム「detailNote」に登録するデータを保持するフィールド名
detailType1	仕訳連携用明細テーブルのカラム「detailType1」に登録するデータを保持するフィールド名
detailType2	仕訳連携用明細テーブルのカラム「detailType2」に登録するデータを保持するフィールド名
detailType3	仕訳連携用明細テーブルのカラム「detailType3」に登録するデータを保持するフィールド名
detailExtensionA	仕訳連携用明細テーブルのカラム「detailExtensionA」に登録するデータを保持するフィールド名
detailExtensionB	仕訳連携用明細テーブルのカラム「detailExtensionB」に登録するデータを保持するフィールド名
detailExtensionC	仕訳連携用明細テーブルのカラム「detailExtensionC」に登録するデータを保持するフィールド名
detailExtensionD	仕訳連携用明細テーブルのカラム「detailExtensionD」に登録するデータを保持するフィールド名
detailExtensionE	仕訳連携用明細テーブルのカラム「detailExtensionE」に登録するデータを保持するフィールド名

detailExtensionF	仕訳連携用明細テーブルのカラム「detailExtensionF」に登録するデータを保持するフィールド名
detailExtensionG	仕訳連携用明細テーブルのカラム「detailExtensionG」に登録するデータを保持するフィールド名
detailExtensionH	仕訳連携用明細テーブルのカラム「detailExtensionH」に登録するデータを保持するフィールド名
detailExtensionI	仕訳連携用明細テーブルのカラム「detailExtensionI」に登録するデータを保持するフィールド名
detailExtensionJ	仕訳連携用明細テーブルのカラム「detailExtensionJ」に登録するデータを保持するフィールド名
journalStatus	仕訳連携用明細テーブルのカラム「journalStatus」に登録するデータを保持するフィールド名
wishDate	仕訳連携用明細テーブルのカラム「wishDate」に登録するデータを保持するフィールド名
accountCdPayable	仕訳連携用明細テーブルのカラム「accountCdPayable」に登録するデータを保持するフィールド名
accountCdWithholdingTax	仕訳連携用明細テーブルのカラム「accountCdWithholdingTax」に登録するデータを保持するフィールド名
withholdingTax	仕訳連携用明細テーブルのカラム「withholdingTax」に登録するデータを保持するフィールド名
fileId	仕訳連携用明細テーブルのカラム「fileId」に登録するデータを保持するフィールド名
procNo	仕訳連携用明細テーブルのカラム「procNo」に登録するデータを保持するフィールド名
extensionA	仕訳連携用明細テーブルのカラム「extensionA」に登録するデータを保持するフィールド名
extensionB	仕訳連携用明細テーブルのカラム「extensionB」に登録するデータを保持するフィールド名
extensionC	仕訳連携用明細テーブルのカラム「extensionC」に登録するデータを保持するフィールド名
extensionD	仕訳連携用明細テーブルのカラム「extensionD」に登録するデータを保持するフィールド名
extensionE	仕訳連携用明細テーブルのカラム「extensionE」に登録するデータを保持するフィールド名
extensionF	仕訳連携用明細テーブルのカラム「extensionF」に登録するデータを保持するフィールド名
extensionG	仕訳連携用明細テーブルのカラム「extensionG」に登録するデータを保持するフィールド名
extensionH	仕訳連携用明細テーブルのカラム「extensionH」に登録するデータを保持するフィールド名
extensionI	仕訳連携用明細テーブルのカラム「extensionI」に登録するデータを保持するフィールド名
extensionJ	仕訳連携用明細テーブルのカラム「extensionJ」に登録するデータを保持するフィールド名

設定方法（例）

```

<event target="matter_end" name="journalSourceDetailEntryForMappingEvent">
  <args>
    <arg key="journalDataTypeCd" value="journalDataTypeCd"/>
    <arg key="tupleId" value="tupleId"/>
    <arg key="rowNumber" value="rowNumber"/>
    <arg key="expTypeCd" value="expTypeCd"/>
    <arg key="accountCd" value="accountCd"/>
    <arg key="taxTypeCd" value="taxTypeCd"/>
    <arg key="taxRate" value="taxRate"/>
    <arg key="quantity" value="quantity"/>
    <arg key="transCurrencyCd" value="transCurrencyCd"/>
    <arg key="transAmount" value="transAmount"/>
    <arg key="companyCurrencyCd" value="companyCurrencyCd"/>
    <arg key="companyRate" value="companyRate"/>
    <arg key="companyAmount" value="companyAmount"/>
    <arg key="companyAmountExcTax" value="companyAmountExcTax"/>
    <arg key="companyAmountTax" value="companyAmountTax"/>
    <arg key="companyAmountSum" value="companyAmountSum"/>
    <arg key="customerCd" value="customerCd"/>
    <arg key="customerName" value="customerName"/>
    <arg key="departmentSetCd" value="departmentSetCd"/>
    <arg key="departmentCd" value="departmentCd"/>
    <arg key="projectCd" value="projectCd"/>
    <arg key="payMethodCd" value="payMethodCd"/>
    <arg key="payTypeCd" value="payTypeCd"/>
    <arg key="advMatterNumber" value="advMatterNumber"/>
    <arg key="advMatterName" value="advMatterName"/>
    <arg key="detailDate1" value="detailDate1"/>
    <arg key="detailDate2" value="detailDate2"/>
    <arg key="detailSummary1" value="detailSummary1"/>
    <arg key="detailSummary2" value="detailSummary2"/>
    <arg key="detailSummary3" value="detailSummary3"/>
    <arg key="detailSummary4" value="detailSummary4"/>
    <arg key="detailSummary5" value="detailSummary5"/>
    <arg key="detailNote" value="detailNote"/>
    <arg key="detailType1" value="detailType1"/>
    <arg key="detailType2" value="detailType2"/>
    <arg key="detailType3" value="detailType3"/>
    <arg key="detailExtensionA" value="detailExtensionA"/>
    <arg key="detailExtensionB" value="detailExtensionB"/>
    <arg key="detailExtensionC" value="detailExtensionC"/>
    <arg key="detailExtensionD" value="detailExtensionD"/>
    <arg key="detailExtensionE" value="detailExtensionE"/>
    <arg key="detailExtensionF" value="detailExtensionF"/>
    <arg key="detailExtensionG" value="detailExtensionG"/>
    <arg key="detailExtensionH" value="detailExtensionH"/>
    <arg key="detailExtensionI" value="detailExtensionI"/>
    <arg key="detailExtensionJ" value="detailExtensionJ"/>
    <arg key="journalStatus" value="journalStatus"/>
    <arg key="wishDate" value="wishDate"/>
    <arg key="accountCdPayable" value="accountCdPayable"/>
    <arg key="accountCdWithholdingTax" value="accountCdWithholdingTax"/>
    <arg key="withholdingTax" value="withholdingTax"/>
    <arg key="fileId" value="fileId"/>
    <arg key="procNo" value="procNo"/>
    <arg key="extensionA" value="extensionA"/>
    <arg key="extensionB" value="extensionB"/>
    <arg key="extensionC" value="extensionC"/>
    <arg key="extensionD" value="extensionD"/>
    <arg key="extensionE" value="extensionE"/>
    <arg key="extensionF" value="extensionF"/>
    <arg key="extensionG" value="extensionG"/>
    <arg key="extensionH" value="extensionH"/>
    <arg key="extensionI" value="extensionI"/>
    <arg key="extensionJ" value="extensionJ"/>
  </args>
</event>

```

#### 仕訳明細データ登録イベント (SQL) [ name : journalSourceDetailEntryForSqlEvent ]

##### 概要

ブロックデータを引数のSQLに従って、仕訳連携用明細テーブルに登録するイベントです。  
主に「案件終了処理」時の利用を想定したイベントです。

##### 引数

キー	設定値
sqlPath	仕訳連携用明細データ取得用SQL (必須)

設定方法 (例)

```
<event target="matter_end" name="journalSourceDetailEntryForSqlEvent">
  <args>
    <arg key="sqlPath" value="META-INF/sql/jp/co/slcs/kaiden2/generic/gadget/journal/Detail.sql"/>
  </args>
</event>
```

## 仮払残高データ処理イベント [ name : advanceBalanceEvent ]

概要

仮払残高テーブルのデータを登録・削除するイベントです。

引数

キー	設定値
companyCurrencyCd	会社通貨コードを保持するフィールド名 (必須)
companyAmount	会社通貨額を保持するフィールド名 (必須)
advanceBalance	仮払残高額を保持するフィールド名 (必須)

設定方法 (例)

```
<event target="always" name="advanceBalanceEvent">
  <args>
    <arg key="companyCurrencyCd" value="companyCurrencyCd"/>
    <arg key="companyAmount" value="companyAmount"/>
    <arg key="advanceBalance" value="companyAmount"/>
  </args>
</event>
```

## 仮払精算データ処理イベント [ name : advanceSettleEvent ]

概要

仮払精算テーブルのデータを登録・更新・削除するイベントです。

引数

なし

設定方法 (例)

```
<event target="always" name="advanceSettleEvent"/>
```

## 法人カードデータ利用状況登録イベント [ name : searchCorporateCardEvent ]

概要

法人カードデータの利用状況を登録するイベントです。

引数

キー	設定値
resultId	法人カードデータの主キー連携用ID (必須)

設定方法 (例)

```
<event target="always" name="searchCorporateCardEvent">
  <args>
    <arg key="resultId" value="result"/>
  </args>
</event>
```

## ICカードデータ利用状況登録イベント [ name : searchTransitManagerEvent ]

概要

ICカードデータの利用状況を登録するイベントです。

#### 引数

キー	設定値
resultId	ICカードデータの主キー連携用ID（必須）
relation	申請情報の「transit manager」連携有無（必須）

#### 設定方法（例）

```
<event target="always" name="searchTransitManagerEvent">
  <args>
    <arg key="resultId" value="result"/>
    <arg key="relation" value="true"/>
  </args>
</event>
```

## インクルードJSP

GadgetBuilder定義の「[インクルード設定](#)」で設定可能なインクルードJSPについて説明します。

- [intra-mart Accel Kaiden!標準のインクルードJSP](#)
  - [検索結果取得ガジェット用インクルードJSP（共通）](#)
  - [検索結果取得ガジェット用インクルードJSP](#)
  - [検索結果取得ガジェット用インクルードJSP（TransitManager）](#)
  - [事前申請書検索用インクルードJSP](#)
  - [個人設定取得用インクルードJSP](#)
  - [明細行数上限設定用インクルードJSP](#)
  - [案件名設定用インクルードJSP](#)
  - [CSVファイル取り込み用インクルードJSP](#)

## intra-mart Accel Kaiden!標準のインクルードJSP

本項では、intra-mart Accel Kaiden!で提供しているインクルードJSPと設定方法を説明します。

### 検索結果取得ガジェット用インクルードJSP（共通）

#### 概要

検索系ガジェットの検索結果を取得するガジェットで共通利用する処理を実装したJSPです。

#### インクルードパス

/WEB-INF/view/kaiden2/extension\_imw/generic/gadget/include/search\_v00.jsp

#### 引数

キー	説明
notSearchable	検索ガジェット利用不可の状態を表す要素
notSearchable / display	検索ガジェットが利用不可の場合の自ガジェット表示有無 <ul style="list-style-type: none"> <li>▪ <b>true</b> : 自ガジェットを表示しない</li> <li>▪ <b>false</b> : 自ガジェットを表示する</li> </ul>

#### 設定方法（例）

```
<include path="/WEB-INF/view/kaiden2/extension_imw/generic/gadget/include/search_v00.jsp"
  args="{
    'notSearchable':{
      'display':'false'
    }
  }"
/>
```

### 検索結果取得ガジェット用インクルードJSP

## 概要

検索系ガジェットの検索結果を取得するガジェット用の処理を実装したJSPです。

## インクルードパス

/WEB-INF/view/kaiden2/extension\_imw/generic/gadget/include/search\_v01.jsp

## 引数

キー	設定値
type	検索種別 <ul style="list-style-type: none"> <li>▪ <b>CorporateCard</b> : 法人カード検索</li> <li>▪ <b>TransitManager</b> : ICカード検索</li> <li>▪ <b>JnaviCorporateCard</b> : 法人カード検索(J'sNAVI Jr.)</li> <li>▪ <b>BusinessTripArrangement</b> : 出張手配実績データ検索</li> <li>▪ <b>BtmService</b> : BTMサービスデータ検索</li> </ul>
settings	設定を表す要素
settings / toolbar	ツールバーの検索アイコン設定を表す要素
settings / toolbar / use	ツールバーの検索アイコン利用有無 <ul style="list-style-type: none"> <li>▪ <b>true</b> : ツールバーの検索アイコンを利用する (表示する)</li> <li>▪ <b>false</b> : ツールバーの検索アイコンを利用しない (表示しない)</li> </ul>
settings / toolbar / surface	ツールバーの検索リンク利用時のガジェット名 (サーフィスキー)
settings / search	自ガジェットに表示する検索アイコン設定を表す要素 (検索画面を表示するアイコン)
settings / search / use	検索アイコン利用有無 <ul style="list-style-type: none"> <li>▪ <b>true</b> : 検索アイコンを利用する (表示する)</li> <li>▪ <b>false</b> : 検索アイコンを利用しない (表示しない)</li> </ul>
settings / search / surface	検索アイコン横のラベル (サーフィスキー)
settings / search / sprite	検索アイコンのCSS Sprite
settings / import	自ガジェットに表示する取得アイコン設定を表す要素 (検索画面を表示せず、検索内容をすべて取得するアイコン)
settings / import / use	取得アイコン利用有無 <ul style="list-style-type: none"> <li>▪ <b>true</b> : 取得アイコンを利用する (表示する)</li> <li>▪ <b>false</b> : 取得アイコンを利用しない (表示しない)</li> </ul>
settings / import / surface	取得アイコン横のラベル (サーフィスキー)
settings / import / sprite	取得アイコンのCSS Sprite
relations	検索結果の連携方法を表す要素
relations / %FieldKey%	検索結果の連携先フィールドを表す要素 「%FieldKey%」には、ガジェットのフィールドIDを設定します。 本要素はフィールド別に複数設定可能です。
relations / %FieldKey% / value	連携先フィールドに設定する値、または検索結果のキー
relations / %FieldKey% / mode	連携先フィールド設定モード <ul style="list-style-type: none"> <li>▪ <b>1</b> : フィールドに値がない場合、値を設定</li> <li>▪ <b>2</b> : フィールドに値がある場合、値を設定</li> <li>▪ <b>0</b> : 常に値を設定</li> </ul>
relations / %FieldKey% / disabled	値設定時の連携先フィールドの利用不可設定有無 <ul style="list-style-type: none"> <li>▪ <b>true</b> : 値設定時に連携先フィールドを利用不可に設定する</li> <li>▪ <b>false</b> : 値設定時に連携先フィールドを利用不可に設定しない</li> </ul> 連携解除時には、利用不可設定が解除されます。

relations / %FieldKey% / readonly	値設定時の連携先フィールドの読み取り専用設定有無 <ul style="list-style-type: none"> <li>▪ <b>true</b> : 値設定時に連携先フィールドを読み取り専用を設定する</li> <li>▪ <b>false</b> : 値設定時に連携先フィールドを読み取り専用に設定しない</li> </ul> 連携解除時には、読み取り専用設定が解除されます。
display	表示内容を表す要素
display / target	連携済アイコンを表示するフィールドID 指定フィールドの前要素 (before) に連携済アイコンを表示します。
display / sprite	連携済アイコンのCSS Sprite
display / titleRelation	連携済アイコン表示時のツールチップ (サーフィスキー)
display / titleCancel	連携解除アイコン表示時のツールチップ (サーフィスキー)
display / deleteRow	連携解除時の連携行削除有無 <ul style="list-style-type: none"> <li>▪ <b>true</b> : 連携解除時に連携行を削除する</li> <li>▪ <b>false</b> : 連携解除時に連携行を削除しない</li> </ul>
additional	連携済データ返却時の追加情報を表す要素
additional / %additionalkey%	連携済データ返却時の追加データ 「%additionalkey%」には、追加データのキーを設定します。

#### 設定方法 (例)

```
<include path="/WEB-INF/view/kaiden2/extension_imw/generic/gadget/include/search_v01.jsp"
  args="{
    'type':'CorporateCard',
    'settings':{
      'toolbar':{'use':'false', 'surface':'chapterTitle'},
      'search':{'use':'true', 'surface':'searchCorporateCard', 'sprite':'kaiden-ui-icon-corporatecard_color-16'},
      'import':{'use':'false', 'surface':'importCorporateCard', 'sprite':'kaiden-ui-icon-corporatecard_color-16'}
    },
    'relations':{
      'occurDate' :{'value':'useDate', 'mode':'0', 'disabled':'true', 'readonly':'true'},
      'transAmount' :{'value':'amount', 'mode':'0', 'disabled':'true', 'readonly':'true'},
      'payTypeCd' :{'value':'0005', 'mode':'0', 'disabled':'true'},
      'summary' :{'value':'multiArea', 'mode':'1'}
    },
    'display':{
      'target' :'relation',
      'sprite' :'kaiden-ui-icon-corporatecard_color-24',
      'titleRelation':'relationCorporateCard',
      'titleCancel' :'cancelCorporateCard'
      'deleteRow' :'false',
    },
    'additional':{'cancelFlag':'0'}
  }"
/>
```

#### 検索結果取得ガジェット用インクルードJSP (TransitManager)

##### 概要

ICカード (TransitManager) 検索ガジェットの検索結果を取得するガジェット用の処理を実装したJSPです。当JSPはJSP版のICカード検索ガジェットを使用する場合インクルードします。

##### インクルードパス

/WEB-INF/view/kaiden2/product\_workflow/generic/gadget/include/searchTransitManager.jsp

##### 引数

キー	設定値
settings	設定を表す要素
settings / search	自ガジェットに表示する検索アイコン設定を表す要素 (検索画面を表示するアイコン)

settings / search / use	検索アイコン利用有無 <ul style="list-style-type: none"> <li><b>true</b> : 検索アイコンを利用する (表示する)</li> <li><b>false</b> : 検索アイコンを利用しない (表示しない)</li> </ul>
settings / search / surface	検索アイコン横のラベル (サーフィスキー)
settings / search / sprite	検索アイコンのCSS Sprite
settings / import	自ガジェットに表示する取得アイコン設定を表す要素 (検索画面を表示せず、検索内容をすべて取得するアイコン)
settings / import / use	取得アイコン利用有無 <ul style="list-style-type: none"> <li><b>true</b> : 取得アイコンを利用する (表示する)</li> <li><b>false</b> : 取得アイコンを利用しない (表示しない)</li> </ul>
settings / import / surface	取得アイコン横のラベル (サーフィスキー)
settings / import / sprite	取得アイコンのCSS Sprite
relations	検索結果の連携方法を表す要素
relations / %FieldKey%	検索結果の連携先フィールドを表す要素 「%FieldKey%」には、ガジェットのフィールドIDを設定します。 本要素はフィールド別に複数設定可能です。
relations / %FieldKey% / value	連携先フィールドに設定する値、または検索結果のキー
relations / %FieldKey% / mode	連携先フィールド設定モード <ul style="list-style-type: none"> <li><b>1</b> : フィールドに値がない場合、値を設定</li> <li><b>2</b> : フィールドに値がある場合、値を設定</li> <li><b>0</b> : 常に値を設定</li> </ul>
relations / %FieldKey% / disabled	値設定時の連携先フィールドの利用不可設定有無 <ul style="list-style-type: none"> <li><b>true</b> : 値設定時に連携先フィールドを利用不可に設定する</li> <li><b>false</b> : 値設定時に連携先フィールドを利用不可に設定しない</li> </ul> <p>連携解除時には、利用不可設定が解除されます。</p>
relations / %FieldKey% / readonly	値設定時の連携先フィールドの読み取り専用設定有無 <ul style="list-style-type: none"> <li><b>true</b> : 値設定時に連携先フィールドを読み取り専用を設定する</li> <li><b>false</b> : 値設定時に連携先フィールドを読み取り専用に設定しない</li> </ul> <p>連携解除時には、読み取り専用設定が解除されます。</p>
display	表示内容を表す要素
display / target	連携済アイコンを表示するフィールドID 指定フィールドの前要素 (before) に連携済アイコンを表示します。
display / sprite	連携済アイコンのCSS Sprite
display / titleRelation	連携済アイコン表示時のツールチップ (サーフィスキー)
display / titleCancel	連携解除アイコン表示時のツールチップ (サーフィスキー)
display / deleteRow	連携解除時の連携行削除有無 <ul style="list-style-type: none"> <li><b>true</b> : 連携解除時に連携行を削除する</li> <li><b>false</b> : 連携解除時に連携行を削除しない</li> </ul>
additional	連携済データ返却時の追加情報を表す要素
additional / %additionalkey%	連携済データ返却時の追加データ 「%additionalkey%」には、追加データのキーを設定します。

設定方法 (例)

```
<include path="/WEB-INF/view/kaiden2/product_workflow/generic/gadget/include/searchTransitManager.jsp"
  args="{
    'settings':{
      'search':{:'use':'true', 'surface':'searchTransitManager', 'sprite':'kaiden-ui-icon-iccard_gray-16'},
      'import':{:'use':'false', 'surface':'importTransitManager', 'sprite':'kaiden-ui-icon-iccard_gray-16'}
    },
    'relations':{
      'occurDate' :{:'value':'payDate', 'mode':'0', 'disabled':'true', 'readonly':'true'},
      'transAmount':{:'value':'payment', 'mode':'0', 'disabled':'true', 'readonly':'true'},
      'payTypeCd' :{:'value':'0001', 'mode':'0', 'disabled':'true'},
      'summary' :{:'value':'memo', 'mode':'1'}
    },
    'display':{
      'target' ::'relation',
      'sprite' ::'kaiden-ui-icon-iccard_gray-24',
      'deleteRow' ::'false',
      'titleRelation':'relationTransitManager',
      'titleCancel' ::'cancelTransitManager'
    }
  }"
/>
```

## 事前申請書検索用インクルードJSP

### 概要

事前申請書検索機能を実装したJSPです。

### インクルードパス

/WEB-INF/view/kaiden2/product\_workflow/generic/gadget/include/searchAdvance.jsp

### 引数

キー	設定値
advSystemMatterId	事前申請書のシステム案件IDを保持するフィールドID
advUserDataId	事前申請書のユーザデータIDを保持するフィールドID

### 設定方法（例）

```
<include path="/WEB-INF/view/kaiden2/product_workflow/generic/gadget/include/searchAdvance.jsp"
  args="{
    'advSystemMatterId':'advSystemMatterId',
    'advUserDataId':'advUserDataId'
  }"
/>
```

## 個人設定取得用インクルードJSP

### 概要

個人設定（組織・プロジェクト）の返却用ガジェット間通信を実装したJSPです。

### インクルードパス

/WEB-INF/view/kaiden2/product\_workflow/generic/gadget/include/defaultDepartmentProject.jsp

### 引数

なし

### 設定方法（例）

```
<include path="/WEB-INF/view/kaiden2/product_workflow/generic/gadget/include/defaultDepartmentProject.jsp" />
```

## 明細行数上限設定用インクルードJSP

### 概要

明細ガジェットにおける明細行数の上限値を実装したJSPです。

### インクルードパス

/WEB-INF/view/kaiden2/base/generic/gadget/include/checkGadgetMaxRowCnt.jsp

### 引数

キー	設定値
maxRowCnt	明細行の上限行数 10を設定した場合、明細行数を10行より多く追加できません。 マイナスの値が設定された場合、明細行の上限をチェックしません。

#### 設定方法（例）

```
<include path="/WEB-INF/view/kaiden2/base/generic/gadget/include/checkGadgetMaxRowCnt.jsp"
  args="{
    'maxRowCnt':'10'
  }"
/>
```

### 案件名設定用インクルードJSP

#### 概要

申請画面表示時に指定された引数に基づく値を案件名に設定します。  
ただし、案件名が既に設定されている場合は設定しません。  
引数に明細行のフィールドIDを設定した場合は1行目の値を設定します。

#### インクルードパス

/WEB-INF/view/kaiden2/extension\_imw/generic/gadget/include/setMatterName.jsp

#### 引数

キー	設定値
setFieldIds	案件名に設定する値を保持するフィールドIDの配列
splitRegex	複数指定されたフィールドIDを連結する際に区切る文字 「setFieldIds」で複数のフィールドIDを指定した際の区切り文字を指定します。 区切り文字の設定は任意のため、省略可能です。

#### 設定方法（例）

```
<include path="/WEB-INF/view/kaiden2/extension_imw/generic/gadget/include/setMatterName.jsp"
  args="{
    'setFieldIds':'imwApplyBaseDate, flowName',
    'splitRegex':'_'
  }"
/>
```

### CSVファイル取り込み用インクルードJSP

#### 概要

明細ブロックにCSVファイルをドラッグアンドドロップすると、CSVファイルの内容を明細行として追加します。

#### インクルードパス

/WEB-INF/view/kaiden2/product\_workflow/generic/gadget/include/csvUpload.jsp

#### 引数

キー	設定値
columnList	CSVファイルに定義するカラムのリスト（カンマ区切りで指定） 省略した場合、CSVファイルの1行目を使用します。
character	CSVファイルの文字コード 省略した場合、「UTF-8」を設定します。
enclosing	CSVファイルの囲み文字 「none」を指定すると、囲み文字なしの設定になります。 省略した場合、「"」を設定します。
delimiter	CSVファイルの区切り文字 「\t」を指定すると、タブ区切りの設定になります。 省略した場合、「,」を設定します。

startRows	CSVファイルの読み込み開始行 省略した場合、1行目から読み込みます。 ただし、「columnList」が未指定の場合2行目から読み込みます。
maxRows	CSVファイルの最大読み込み行数 「0」以下を指定した場合、読み込み行数の制限がない設定になります。 省略した場合、「20」を設定します。

#### 設定方法（例）

```
<include path="/WEB-INF/view/kaiden2/product_workflow/generic/gadget/include/csvUpload.jsp"
  args="{
    'columnList':'routeDate,startingSpot,payTypeCd'
    'character':'UTF-8',
    'enclosing':'none',
    'delimiter':',',
    'startRows':'2',
    'maxRows':'20'
  }"
/>
```

本項では、GadgetBuilderを用いたカスタマイズをする場合の手順を解説します。

- 手順① : GadgetBuilder定義コピー
- 手順② : ガジェットクラス、ガジェットバリエーション編集
- 手順③ : ガジェットインポート
- 手順④ : ガジェット設定
- 手順⑤ : GadgetBuilder定義編集
- 手順⑥-1 : GadgetBuilder定義反映(申請書ガジェットマスタ更新)
- 手順⑥-2 : GadgetBuilder定義反映(申請書ガジェットマスタ更新不要)

## 手順① : GadgetBuilder定義コピー

基にする既存ガジェットのGadgetBuilder定義(XMLファイル)をコピーします。

ファイル名を、次のように変更します。

「%ガジェットクラス%\_%ガジェットバリエーション%.xml」

コピーしたGadgetBuilder定義(XMLファイル)を、次のパスの配下へ配置します。

「%パブリックストレージ%/テナント%/kaiden/generic/gadget」

### 注意

「%ガジェットバリエーション%」は、製品で「v」を使用しているため、カスタマイズ用では「v」以外を使用してください。

### 注意

「%ガジェットクラス%」と「%ガジェットバリエーション%」の間に存在する「\_」以外は全角文字の使用を制限していませんが全角文字入りのGadgetBuilder定義をWARファイルに含めた場合、デプロイに失敗する可能性があるため推奨しません。

## 手順② : ガジェットクラス、ガジェットバリエーション編集

「[手順① : GadgetBuilder定義コピー](#)」でコピーしたGadgetBuilder定義(XMLファイル)のガジェットクラス、ガジェットバリエーションを変更します。

編集例

- 編集前

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">
  <gadgetClass>expenseDetail</gadgetClass>
  <gadgetVariation>v01</gadgetVariation>

  <!-- 略 -->

</gadgetBuilderSettings>
```

- 編集後

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">
  <!-- ガジェットクラス変更 -->
  <gadgetClass>sampleDetail</gadgetClass>
  <!-- ガジェットバリエーション変更 -->
  <gadgetVariation>c01</gadgetVariation>

  <!-- 略 -->

</gadgetBuilderSettings>
```

**注意**

ガジェットクラスとガジェットバリエーションは、ファイル名に合わせてください。

## 手順③：ガジェットインポート

「[手順②：ガジェットクラス、ガジェットバリエーション編集](#)」で編集したGadgetBuilder定義(XMLファイル)をインポートします。

実行方法は、「[GadgetBuilder定義インポート](#)」を参照してください。

## 手順④：ガジェット設定

「[手順③：ガジェットインポート](#)」を実行後、申請書マスタメンテナンスでガジェットを設定します。

設定方法は、「[申請書マスタメンテナンス](#)」を参照してください。

**コラム**

ここまでの手順で、既存ガジェットのコピーが完了します。

## 手順⑤：GadgetBuilder定義編集

「[手順④：ガジェット設定](#)」までで、コピーが完了しているGadgetBuilder定義(XMLファイル)を編集します。

編集例①：申請書ガジェットマスタの更新が必要な編集

- 編集前

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

<!-- 略 -->

<groups>
  <group id="targetField" surfaceKey="targetField" requiredType="SETTING_ANY" inputType="CONFIG">
    <fields>
      <field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
        <args>
          <arg key="class" value="common_class"/>
          <arg key="style" value="common_style"/>
          <arg key="maxlength" value="250"/>
          <arg key="readonly" value="false"/>
          <arg key="disabled" value="false"/>
        </args>
      </field>
    </fields>
  </group>
</groups>

<!-- 略 -->

</gadgetBuilderSettings>
```

- 編集後

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

<!-- 略 -->

<groups>
  <group id="targetField" surfaceKey="targetField" requiredType="SETTING_ANY" inputType="CONFIG">
    <fields>
      <!-- フィールド変更 -->
      <field id="targetField1" surfaceKey="targetField1" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
        <args>
          <arg key="class" value="common_class"/>
          <!-- スタイル変更 -->
          <arg key="style" value="width:calc(100% - 10px);"/>
          <arg key="maxLength" value="250"/>
          <arg key="readonly" value="false"/>
          <arg key="disabled" value="false"/>
        </args>
        <!--バリデート追加-->
        <validates>
          <validate event="default" config="{ 'id':'maxLength', 'args':'250' }"/>
        </validates>
      </field>
      <!-- フィールド追加 -->
      <field id="targetField2" surfaceKey="targetField2" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
        <args>
          <arg key="class" value="common_class"/>
          <arg key="style" value="common_style"/>
          <arg key="maxLength" value="250"/>
          <arg key="readonly" value="false"/>
          <arg key="disabled" value="false"/>
        </args>
      </field>
    </fields>
  </group>
</groups>

<!-- 略 -->

</gadgetBuilderSettings>

```

反映方法は、「手順①-1 : GadgetBuilder定義反映(申請書ガジェットマスタ更新)」を参照してください。



## コラム

申請書ガジェットマスタの更新が必要なタグ・属性の一覧です。

名称	タグ・属性
ガジェットクラス	gadgetBuilderSettings / gadgetClass
ガジェットバリエーション	gadgetBuilderSettings / gadgetVariation
JSPパス	gadgetBuilderSettings / jspPath
Javaパス	gadgetBuilderSettings / javaPath
ソートキー	gadgetBuilderSettings / sortKey
サーフィスキー	gadgetBuilderSettings / surfaceKey
ガジェット概要設定	gadgetBuilderSettings / descriptions / description
ガジェットブロック設定	gadgetBuilderSettings / blocks / block
フィールドグループ設定 / フィールドグループID	gadgetBuilderSettings / blocks / block / groups / group @ id
フィールドグループ設定 / サーフィスキー	gadgetBuilderSettings / blocks / block / groups / group @ surfaceKey
フィールドグループ設定 / 必須タイプ	gadgetBuilderSettings / blocks / block / groups / group @ requiredType
フィールドグループ設定 / 入力タイプ	gadgetBuilderSettings / blocks / block / groups / group @ inputType
フィールドグループ設定 / ソートキー	gadgetBuilderSettings / blocks / block / groups / group @ sortKey
フィールド設定 / フィールドID	gadgetBuilderSettings / blocks / block / groups / group / fields / field @ id
フィールド設定 / サーフィスキー	gadgetBuilderSettings / blocks / block / groups / group / fields / field @ surfaceKey
フィールド設定 / 必須タイプ	gadgetBuilderSettings / blocks / block / groups / group / fields / field @ requiredType
フィールド設定 / 入力タイプ	gadgetBuilderSettings / blocks / block / groups / group / fields / field @ inputType
フィールド設定 / ソートキー	gadgetBuilderSettings / blocks / block / groups / group / fields / field @ sortKey
ガジェットサーフィス設定	gadgetBuilderSettings / surfaces / surface
ガジェットタグ設定	gadgetBuilderSettings / tags / tag

## 編集例②：申請書ガジェットマスタの更新が不要な編集

- 編集前

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

<!-- 略 -->

<groups>
  <group id="targetField" surfaceKey="targetField" requiredType="SETTING_ANY" inputType="CONFIG">
    <fields>
      <field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
        <args>
          <arg key="class" value="common_class"/>
          <arg key="style" value="common_style"/>
          <arg key="maxLength" value="250"/>
          <arg key="readonly" value="false"/>
          <arg key="disabled" value="false"/>
        </args>
      </field>
    </fields>
  </group>
</groups>

<!-- 略 -->

</gadgetBuilderSettings>

```

- 編集後

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<gadgetBuilderSettings xmlns="http://kaiden.slcs.co.jp/xmlschema/gadgetBuilderSettings">

<!-- 略 -->

<groups>
  <group id="targetField" surfaceKey="targetField" requiredType="SETTING_ANY" inputType="CONFIG">
    <fields>
      <field id="targetField" surfaceKey="targetField" requiredType="INHERIT" inputType="INHERIT" type="TEXT">
        <args>
          <arg key="class" value="common_class"/>
          <!-- スタイル変更 -->
          <arg key="style" value="width:calc(100% - 10px);"/>
          <arg key="maxLength" value="250"/>
          <arg key="readonly" value="false"/>
          <arg key="disabled" value="false"/>
        </args>
        <!-- バリデート追加 -->
        <validates>
          <validate event="default" config="{ 'id': 'maxLength', 'args': '250' }"/>
        </validates>
      </field>
    </fields>
  </group>
</groups>

<!-- 略 -->

</gadgetBuilderSettings>

```

反映方法は、「[手順②-2 : GadgetBuilder 定義反映\(申請書ガジェットマスタ更新不要\)](#)」を参照してください。

## コラム

申請書ガジェットマスタの更新が不要なタグ・属性の一覧です。

名称	タグ・属性
ガジェットタイトル表示有無	gadgetBuilderSettings / displayTitle
ガジェット引数設定	gadgetBuilderSettings / args / arg
拡張設定 (Logic)	gadgetBuilderSettings / extension / logic
拡張設定 (URL)	gadgetBuilderSettings / extension / url
ガジェットブロック設定 / サービス設定	gadgetBuilderSettings / blocks / block / service
ガジェットブロック設定 / サービス設定 / SQL設定	gadgetBuilderSettings / blocks / block / service / sqls / sql
ガジェットブロック設定 / 構成設定	gadgetBuilderSettings / blocks / block / table
ガジェットブロック設定 / 構成設定 / カラムグループ設定	gadgetBuilderSettings / blocks / block / table / colgroups / colgroup
ガジェットブロック設定 / 構成設定 / 行 - 列設定 (HORIZONTAL)	gadgetBuilderSettings / blocks / block / table / rows / row / cols / col
ガジェットブロック設定 / 構成設定 / 行 - 列設定 (VERTICAL)	gadgetBuilderSettings / blocks / block / table / cols / col / rows / row
ガジェットブロック設定 / ブロック引数設定	gadgetBuilderSettings / blocks / block / args / arg
フィールドグループ設定 / 必須マーク	gadgetBuilderSettings / blocks / block / groups / group @ requiredMark
フィールドグループ設定 / 引数設定	gadgetBuilderSettings / blocks / block / groups / group / args / arg
フィールド設定 / フィールドタイプ	gadgetBuilderSettings / blocks / block / groups / group / fields / field @ type
フィールド設定 / 引数設定	gadgetBuilderSettings / blocks / block / groups / group / fields / field / args/ arg
フィールド設定 / バリデート設定	gadgetBuilderSettings / blocks / block / groups / group / fields / field / validates / validate
ガジェットブロック設定 / 補完設定	gadgetBuilderSettings / blocks / block / completions / completion
ガジェットブロック設定 / 通信設定	gadgetBuilderSettings / blocks / block / claims / claim
ガジェットブロック設定 / イベント設定	gadgetBuilderSettings / blocks / block / events / event
ガジェットブロック設定 / インクルード設定	gadgetBuilderSettings / blocks / block / includes / include

## 手順⑥-1 : GadgetBuilder定義反映(申請書ガジェットマスタ更新)

「[手順⑤ : GadgetBuilder定義編集](#)」の編集例①で編集したGadgetBuilder定義(XMLファイル)を反映させます。

申請書ガジェットマスタの更新が必要な場合、「[手順③ : ガジェットインポート](#)」と「[手順④ : ガジェット設定](#)」を再度行います。

## 手順⑥-2 : GadgetBuilder定義反映(申請書ガジェットマスタ更新不要)

「[手順⑤ : GadgetBuilder定義編集](#)」の編集例②で編集したGadgetBuilder定義(XMLファイル)を反映させます。

申請書ガジェットマスタの更新が不要な場合、キャッシュ削除ジョブを実行後、申請書マスタメンテナンスの更新を行います。

## コラム

「[手順⑥-1 : GadgetBuilder定義反映\(申請書ガジェットマスタ更新\)](#)」でも反映は行われますが、申請書ガジェットマスタの更新が不要なため、余分な操作を行わずに反映ができます。

**i** コラム

キャッシュ削除ジョブは、ジョブネット一覧（画面左部）の「キャッシュ削除」で実行できます。

**ジョブネット管理**

カテゴリ新規作成 | ジョブネット新規作成 | ジョブ管理 | ジョブネットモニター

**ジョブネット一覧**

- IM-ContentsSearch クローラ
- IM-FileExchange
- ジョブ管理
- IM共通マスタ
- IM-Workflow
- Kaiden!
  - JstNAVI Jr.
  - マスタ
  - その他
    - 期間作成 (共通)
    - 期間作成 (会社)
    - キャッシュ削除**
  - レポート
  - 法人カード
  - ドキュメントデータ削除
  - ドキュメントデータインポート
- 勤務管理
- 経費精算・旅費精算
- 多要素認証機能
- テナントマスタ

**ジョブネット情報**

基本情報

ジョブネットカテゴリ: その他

ジョブネットID: kaiden-jobnet-other-delete-cache

ジョブネット名

日本語	キャッシュ削除
英語	Delete cache
中国語 (中華人民共和国)	缓存清除

ジョブネットの説明

実行時の情報

並列実行  並列実行を許可する

実行ジョブ

ジョブID	ジョブ名
kaiden-job-other-delete-cache	キャッシュ削除

実行パラメータ

キー	値
----	---

トリガ設定

日時指定

繰り返し指定

営業日指定

**即時実行** | 編集 | 削除

**i** コラム

申請書マスタメンテナンスの更新では、ガジェットの再設定は不要です。  
コンテンツ定義、コンテンツバージョンどちらの更新でも可です。

**申請書マスタ(コンテンツ定義)メンテナンス**

← コンテンツバージョン編集

コンテンツID \* sample\_v01

コンテンツ名 \*

日本語	サンプル申請
英語	sample
中国語 (中華人民共和国)	样品申请

備考

日本語	
英語	
中国語 (中華人民共和国)	

申請書タイプ \* 一般

**更新** | 削除 | 一覧に戻る

