Copyright © 2012 NTT DATA INTRAMART CORPORATION

目次

- 1. 改訂情報
- 2. はじめに
  - 2.1. 本書の目的
  - 2.2. 対象読者
  - 2.3. 本書の構成
  - 2.4. 用語解説
- 3. IM-PDFDesigner for Accel Platform について
  - 3.1. IM-PDFDesigner for Accel Platform とは
  - 3.2. IM-PDFDesigner for Accel Platform の全体像
  - 3.3. 埋め込むデータの指定方法について
  - 3.4. IM-PDFDesigner for Accel Platform を利用したPDFファイル作成の流れ
  - 3.5. PDFファイルの作成パターン
  - 3.6. 開発手法
  - 3.7. プログラム開発における注意点
- 4. 帳票デザインツール IOWebDOC について
  - 4.1. 帳票デザインツール概要
  - 4.2. 帳票デザインツールに関する注意点
  - 4.3. 帳票デザインツールマニュアル
- 5. サンプル
  - 5.1. サンプルプログラム
  - 5.2. サンプル帳票レイアウト
- 6. チュートリアル
  - 6.1. 基本
  - 6.2. 応用
- 7. ステータスコード表
- 8. サポート
- 9. 付録
  - 9.1. intra-mart e Builder for Accel Platform との連携方法
  - 9.2. 帳票デザインツール IODOC 補足情報
  - 9.3. PDFファイルへの印影付与について
  - 9.4. 「^」の扱いについて
  - 9.5. IM-PDFDesigner for Accel Platform で廃止されたAPI
  - 9.6. 連票形式 IOCELA の出力カスタマイズ

改訂情報

変更年月日	変更内容
2012-12-21	初版
2013-12-20	第2版 下記を追加・変更しました。 ▪ ドキュメント全般 Windows Server 2012 向けの記述を追加
2014-07-04	第3版 下記を追加・変更しました。 ▪ プログラミングガイドの内容をセットアップガイドを基準に修正
2014-12-01	第4版 下記を追加・変更しました。 ▪ ドキュメント全般 Windows Server 2012 R2 向けの記述を追加 ▪ IOCELA（連票形式）の出力カスタマイズの項目を追加 ▪ その他（便利な機能）の記述を追加
2015-07-01	第5版 下記を追加・変更しました。 ▪ intra-mart Accel Platform へのバージョンアップ時の注意事項の記述を追加 ▪ その他（便利な機能）の記述を追加
2016-08-01	第6版 下記を追加・変更しました。 ▪ intra-mart Accel Platform へのバージョンアップ時の注意事項の記述を追加 ▪ 「レイアウト定義ファイルの作成」の注意事項を追加
2016-12-01	第7版 下記を追加・変更しました。 ▪ 「ツールマニュアル（帳票レイアウト作成）」 の項目を追加
2017-08-01	第8版 下記を追加・変更しました。 ▪ 新帳票ツール「IODBDOC」 の記述を追加 ▪ 新機能「RESTインタフェース機能」 の記述を追加
2017-12-01	第9版 下記を追加・変更しました。 ▪ 「 トラブルシューティング 」の内容を変更 ▪ 「 *チュートリアル* 」の内容を変更
2018-04-01	第10版 下記を追加・変更しました。 ▪ 「 **IODOC 簡易マニュアル** 」より「レイアウト作成ツール」を「レイアウトデザインツール」に変更 ▪ 「 トラブルシューティング 」の内容を変更 ▪ 「 *ステータスコード表* 」の内容を変更 ▪ 「 *サンプル帳票レイアウト* 」の内容を変更
2018-08-01	第11版 下記を追加・変更しました。 ▪ 「 ツールマニュアル（帳票レイアウト作成）:連票ツール：IOCELA」で「**IOCELAマニュアル**」の文書プロパティを削除 ▪ 「 ツールマニュアル（帳票レイアウト作成）:連票ツール：IOCELA」で「**IOCELA簡易マニュアル**」の文書プロパティを削除

変更年月日	変更内容
2018-12-01	第12版 下記を追加・変更しました。 ▪ 「 トラブルシューティング 」で対応ケースを追記 ▪ 表記のゆれを訂正
2019-04-01	第13版 下記を追加・変更しました。 ▪ 「 トラブルシューティング 」を本書から独立（トラブルシューティングは、「 IM-PDFDesigner for Accel Platform トラブルシューティング 」を参照してください。）
2019-12-01	第14版 下記を追加・変更しました。 ▪ 「 *ステータスコード表* 」の記載を追加・変更 ▪ 見出しを「エラーコード表」から「ステータスコード表」へ変更 ▪ ステータスコード表のステータスコード 0 の説明を見直し ▪ ステータスコード表外の説明文を見直し ▪ 「 *サンプルプログラムに関する注意点* 」にデータファイルの文字コードに関する記載を追加
2020-08-01	第15版 下記を追加・変更しました。 ▪ 「intra-mart Accel Platform へのバージョンアップ時の注意事項」を「 intra-mart WebPlatform から intra-mart Accel Platform への移行 」へ変更 ▪ 「 intra-mart WebPlatform から intra-mart Accel Platform への移行 」の記載を見直し、IM-PDFDesigner for Accel Platform 移行ガイド を参照するよう変更
2020-12-01	第16版 下記を追加・変更しました。 ▪ 「 プログラムのコンパイル 」の記述を変更
2021-04-01	第17版 下記を追加・変更しました。 ▪ 「 レイアウト定義ファイルの作成 」に印影付与についての記述を追加 ▪ 「 スクリプト開発モデル 」 ▪ 「 連票形式（ IODBDOC ） 」を追加 ▪ 「 連票形式 」の見出しを「 連票形式（ IOCELA ） 」へ変更 ▪ 「 JavaEE 開発モデル 」 ▪ 「 連票形式（ IODBDOC ） 」を追加 ▪ 「 連票形式 」の見出しを「 連票形式（ IOCELA ） 」へ変更 ▪ 「 APIの概要 」のコラムのリンク先を変更 ▪ 「 サンプルプログラム・データの保存位置 」の記述を変更 ▪ 「 メニュー構成 」の記述を変更 ▪ 「 業務日報 」を追加 ▪ 「 *PDF ファイルへの印影付与について* 」を追加 ▪ 「 **IODBDOCマニュアル** 」を更新 ▪ 「 **IODBDOC簡易マニュアル** 」を更新
2021-08-01	第18版 下記を追加・変更しました。 ▪ 「 プログラム開発における注意点 」へ記述を追加 ▪ 「 サンプルプログラムの説明 」に注意を追加 ▪ 「 *ステータスコード表* 」へ記述を追加 ▪ 「 プログラム開発における注意点 」で画像のパス指定に関してCSVファイルと限定していた為、「CSVファイルに」の文言を削除

変更年月日	変更内容
2022-06-01	第19版 下記を追加・変更しました。 ■ 「 *用語解説* 」の「 Storageのディレクトリ（ %PUBLIC_STORAGE_PATH% ） 」を削除し、「 PublicStorageのディレクトリ（ %PUBLIC_STORAGE_PATH% ） 」を追加 ■ ドキュメント全体の「 Storageのディレクトリ（ %PUBLIC_STORAGE_PATH% ） 」の記述を見直し、「 PublicStorageのディレクトリ（ %PUBLIC_STORAGE_PATH% ） 」へ変更 ■ 「 開発手法 」の「 プログラム開発における注意点 」の記載場所を移動 ■ 「 開発手法 」の「 スクリプト開発モデル 」、「JavaEE開発モデル」、および、「プログラムのコンパイル」の記述を、「 開発手法 」に移動し、移動元の各項目を削除 ■ 「 プログラム開発における注意点 」の相対パス、および、絶対パスに関する記述を見直し、変更 ■ 「 プログラム開発における注意点 」のダブルクリック防止に関するAPIの記述を削除 ■ 「 *付録* 」に「 *「^」の扱いについて* 」を追加 ■ 「 *サポート* 」を追加
2023-04-01	第20版 下記を追加・変更しました。 ■ 「 プログラム開発における注意点 」 　■ setCol()のnull指定について記述を追加 　■ setCol()使用時のカラムデータについて記述を追加 　■ 連票形式の指定可能なデータについて記述を変更 ■ 「 サンプルプログラムの説明 」 　■ 連票形式の指定可能なデータについて記述を変更 　■ 「 メモリデータを用いて連票形式のPDFを作成するサンプル 」を追加 　■ 「 異なるレイアウトを結合してPDFを作成するサンプル 」を追加 　■ 「 レイアウトを重ね合わせるサンプル 」を削除 ■ 「 *ステータスコード表* 」 　■ 「-11」、および、「-23」のコードに対するエラー内容を追加 　■ 「-515」、および、「-516」のコード、および、エラー内容を追加 　■ 一覧をコード単位での記載に変更

変更年月日	変更内容
2023-10-01	第21版 下記を追加・変更しました。 ■ 「 連票用（ IOCELA ） 」に出力制御に関するコラムを追加 ■ 「 ツールマニュアル（帳票レイアウト作成） 」を「 レイアウト定義ファイルの作成 」に移動 ■ 「 プログラム開発における注意点 」の箇条書きの各記述を章として追加 ■ 「 intra-mart WebPlatform から移行する際、ソースコード等の調整が必要な場合があります 」を追加 ■ 「 概要 」に IM-LogicDesigner に関するコラムを追加 ■ 「 スクリプト開発モデル 」 ■ 「 単票形式 」 ■ 「 入力画面の作成 」のサンプルプログラムを変更 ■ 「 連票形式（ IODBDOC ） 」 ■ 「 入力画面の作成 」のサンプルプログラムを変更 ■ 「 連票形式（ IOCELA ） 」 ■ 「 入力画面の作成 」のサンプルプログラムを変更 ■ 「 JavaEE 開発モデル 」 ■ 「 単票形式 」 ■ 「 入力画面処理の作成 」のサンプルプログラムを変更 ■ 「 出力画面処理の作成 」のサンプルプログラムを変更 ■ 「 連票形式（ IODBDOC ） 」 ■ 「 入力画面処理の作成 」のサンプルプログラムを変更 ■ 「 出力画面処理の作成 」のサンプルプログラムを変更 ■ 「 連票形式（ IOCELA ） 」 ■ 「 入力画面処理の作成 」のサンプルプログラムを変更 ■ 「 出力画面処理の作成 」のサンプルプログラムを変更 ■ 「 *サンプル* 」を追加 ■ 「 *サンプルプログラム* 」を「 *サンプル* 」に移動 ■ 「 *サンプル帳票レイアウト* 」を「 *サンプル* 」に移動 ■ 「 *サポート* 」のサポート窓口先の記述を変更 ■ 「 intra-mart WebPlatform から intra-mart Accel Platform への移行 」を「*付録* 」に移動 ■ 「 *連票形式 IOCELA の出力カスタマイズ*」を「 *付録* 」に移動 ■ 「 IM-PDFDesigner for Accel Platform を使って IM-LogicDesigner でIODファイルを作成・連結後、PDF帳票を出力する方法 」を追加

変更年月日	変更内容
2024-04-01	第22版 下記を追加・変更しました。 ■ 「 *対象読者* 」の記述を変更 ■ 「 *本書の構成* 」を追加 ■ 「 *用語解説* 」の記述を変更 ■ 「 概要 」を削除 ■ 「*IM-PDFDesigner for Accel Platform について*」を追加 ■ 「 レイアウト定義ファイルの作成 」を削除 ■ 「 *帳票デザインツール IOWebDOC について* 」を追加 ■ 「 PDFファイル作成用のプログラムの開発 」を削除 ■ 「 *サンプルプログラム* 」の構成、および、記述を変更 ■ 「 サンプルプログラム・データの保存位置 」を削除 ■ 「 *サンプルプログラム・データの設置場所* 」を追加 ■ 「 サンプルプログラムの説明 」を削除 ■ 「 *サンプルプログラムに関する注意点* 」の記述を変更 ■ 「 *サンプルプログラムの実行方法* 」の構成、および、記述を変更 ■ 「 *サンプル帳票レイアウト* 」の構成を変更 ■ 「 *チュートリアル* 」の構成、および、記述を変更 ■ 「*intra-mart e Builder for Accel Platform との連携方法*」の記述を変更 ■ 「 グループ化機能の使い方 」を削除 ■ 「 文字サイズの自動縮小機能 」を削除 ■ 「 PDFファイルのサイズ縮小設定 」を削除 ■ 「 *帳票デザインツール IODOC 補足情報* 」を追加 ■ 「 PDFファイルの自動印刷機能（直接印刷） 」を削除 ■ 「 *PDFファイルへの印影付与について* 」の構成、および、記述を変更 ■ 「 *「^」の扱いについて* 」の記述を変更 ■ 「 intra-mart WebPlatform から intra-mart Accel Platform への移行 」の構成、および、記述を変更 ■ 「 *連票形式 IOCELA の出力カスタマイズ* 」の構成、および、記述を変更 ■ 「 IM-PDFDesigner for Accel Platform を使って IM-LogicDesigner でIODファイルを作成・連結後、PDF帳票を出力する方法 」を削除

変更年月日	変更内容
2024-10-01	第23版 下記を追加・変更しました。 ■ 「*IM-PDFDesigner for Accel Platform について*」 ■■ 「*データファイル*」の intra-mart Accel Platform の文字コードに関する注意を削除 ■■ 「*プログラム開発における注意点*」 ■■■ 「intra-mart WebPlatform から移行する際、ソースコード等の調整が必要な場合があります」を削除 ■■■ 「*IM-PDFDesigner for Accel Platform で廃止されたAPIは利用しないでください*」を追加 ■ 「*帳票デザインツールに関する注意点*」 ■■ 「*IODBDOC のデータ設定画面でのデータ型は「テキスト」を選択してください*」を追加 ■ 「*サンプル*」 ■■ 「*サンプルプログラムに関する注意点*」 ■■■ 「IOWebDOC 3.x の場合、データファイルの文字コードを UTF-8（BOMあり） に変更してください」を削除 ■■■ 「*IOWebDOC 1.x の場合、データファイルの文字コードを Shift_JIS に変更してください*」を追加 ■ 「*付録*」 ■■ 「intra-mart WebPlatform から intra-mart Accel Platform への移行」を削除 ■■ 「*IM-PDFDesigner for Accel Platform で廃止されたAPI*」を追加

はじめに

本書の目的

本書では IM-PDFDesigner for Accel Platform を利用したユーザプログラムを開発する場合の基本的な方法や注意点等について説明します。

対象読者

次の開発者を対象としています。

- IM-PDFDesigner for Accel Platform で初めてプログラミングを行う開発者
- IM-PDFDesigner for Accel Platform の各機能を利用したい開発者

Java でプログラムを実装する場合は、 JavaEE開発モデル によるプログラムの開発方法を理解している必要があります。

JavaScript でプログラムを実装する場合は、 スクリプト開発モデル によるプログラムの開発方法を理解している必要があります。

ロジックフロー でプログラムを実装する場合は、 IM-LogicDesigner の仕様、操作方法を理解している必要があります。

> **注意**
>
> IM-LogicDesigner と連携して、PDF帳票の出力処理を作成できるアプリケーションとして、 IM-PDFDesigner FullPack for Accel Platform IM-LogicDesigner連携 （以下 FullPack版 IM-LogicDesigner 連携 と記述します）があります。
>
> FullPack版 IM-LogicDesigner 連携 を使用する場合は、 IM-PDFDesigner FullPack for Accel Platform が必要です。

本書の構成

本書は次の章で構成されています。

- *IM-PDFDesigner for Accel Platform について*

  IM-PDFDesigner for Accel Platform の概要について説明します。

- *帳票デザインツール IOWebDOC について*

  帳票デザインツール IOWebDOC の概要について説明します。

- *サンプル*

  IM-PDFDesigner for Accel Platform のサンプル用のプログラムや帳票レイアウトについて説明します。

- *チュートリアル*

  IM-PDFDesigner for Accel Platform を使用したPDFファイルの作成方法を基本編／応用編に分けて説明します。

- *ステータスコード表*

  IM-PDFDesigner for Accel Platform のAPIで返却される戻り値の一覧です。

- *サポート*

  IM-PDFDesigner for Accel Platform に関する不明点の対応について説明します。

- *付録*

  IM-PDFDesigner for Accel Platform に関連する補足情報について説明します。

用語解説

用語	説明
帳票レイアウト	作成するPDFファイルの雛形
帳票デザインツール	帳票レイアウトを作成するためのツール
単票形式	帳票レイアウトの種類で、帳票上の各項目がそれぞれ独立している様式のもの 次の帳票を作成するのに適している - 複雑なレイアウトの帳票 - 明細表を持たない帳票 - 行数固定の明細表を扱う帳票
連票形式	帳票レイアウトの種類で、帳票上の各項目が連続した複数の行となっている様式（Excel様式）のもの 次の帳票を作成するのに適している - 連続した表形式の帳票 - 行数可変の明細表を扱う帳票
識別子	帳票レイアウト上に、文字列や画像などの情報を埋め込むための項目
%RESIN_HOME%	Resin をインストールしたディレクトリ
%HOME_PATH%	intra-mart Accel Platform ホームディレクトリ
%PUBLIC_STORAGE_PATH%	PublicStorageとして使用するディレクトリ
%IOWEBDOC_HOME%	連携エンジン IOWebDOC をインストールしたディレクトリ

IM-PDFDesigner for Accel Platform について

IM-PDFDesigner for Accel Platform について説明します。

IM-PDFDesigner for Accel Platform とは

IM-PDFDesigner for Accel Platform は、 intra-mart Accel Platform 上でPDFファイルを作成するためのアプリケーションです。

IM-PDFDesigner for Accel Platform の特徴は、次の通りです。

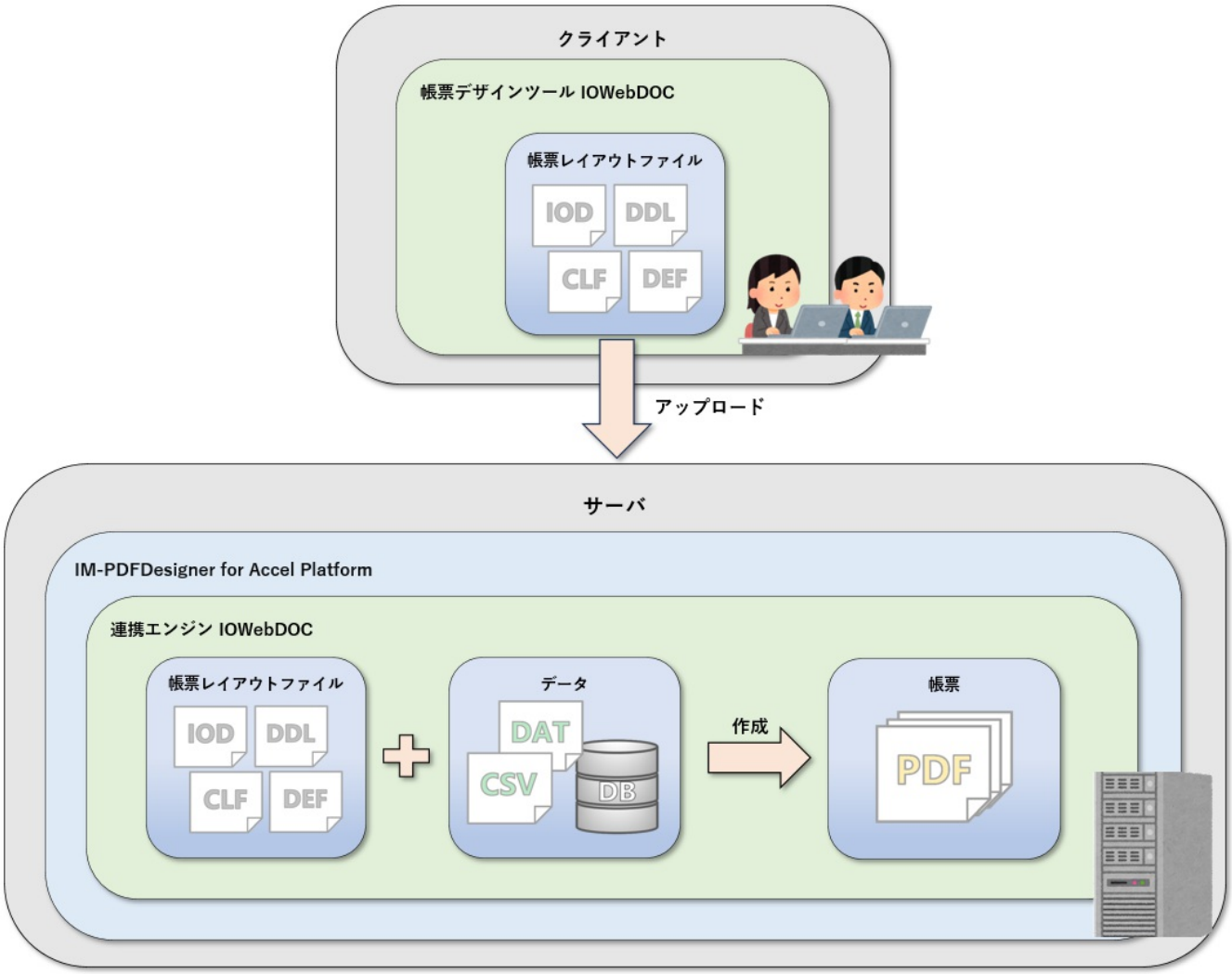
- 帳票レイアウトと文字列等のデータを組み合わせることによりPDFファイルを作成
- 単票、連票、複数票の結合、および、改ページの自動制御に対応
- IM-LogicDesigner の JavaScript定義 に対応
- FullPack版 IM-LogicDesigner 連携 を使用することにより、プログラミングの知識不要で、GUI上の操作だけ（ノーコード）でPDFファイルを作成することが可能

> **コラム**
>
> 本書では、 FullPack版 IM-LogicDesigner 連携 の詳細については記載していません。
>
> FullPack版 IM-LogicDesigner 連携 の詳細については、「IM-PDFDesigner FullPack for Accel Platform IM-LogicDesigner連携 操作ガイド 」を参照してください。

> **コラム**
>
> PDFファイルをプリンタへ自動印刷（直接印刷）したい場合は、 IM-PDFDirectPrint for Accel Platform を使用してください。

IM-PDFDesigner for Accel Platform の全体像

IM-PDFDesigner for Accel Platform の全体像は、次の通りです。



- クライアント

  帳票レイアウトを作成するツールとして、 IOWebDOC を使用します。

  作成した帳票レイアウトファイルは、サーバ側にアップロードします。

- サーバ

  IM-PDFDesigner for Accel Platform の内部のエンジンとして、 IOWebDOC を使用します。

  帳票レイアウトファイルと帳票レイアウトに埋め込むデータを、 連携エンジン IOWebDOC を使用して組み合わせることにより帳票を作成します。

埋め込むデータの指定方法について

PDFファイルの作成時、帳票レイアウトファイルに埋め込むデータの指定方法は、次の通りです。

- CSV形式、または、DAT形式でデータを記述したファイル（以下データファイルと記述します）を使用して、データをまとめて一括で指定する
- プログラム変数を使用してデータを個別に指定する

コラム

帳票デザインツールごとの、PDFファイルの作成時に埋め込むデータの指定方法は、次の通りです。

帳票デザインツール	指定方法
単票形式 IODOC	CSVファイル／DATファイル／プログラム変数
連票形式 IODBDOC	プログラム変数
連票形式 IOCELA	CSVファイル／DATファイル／プログラム変数

注意

CSVファイルでデータを指定する場合、帳票レイアウトファイルとCSVファイルのデータを関連付ける定義ファイル（以下CDDファイルと記述します）が必要です。

CDDファイルは、帳票デザインツールのアプリケーションの1つである CDDエディタ を使用し、作成します。

CDDエディタ の詳細については、「 *単票形式 IODOC* 」を参照してください。

注意

文字列、画像、および、QRコード等のデータを帳票レイアウトファイルに埋め込むことが可能です。

文字列以外のデータの埋め込み方法については、次を参照してください。

- 「 *帳票デザインツールマニュアル* 」
- 「 IM-PDFDesigner for Accel Platform API ドキュメント」
- 「 *PDF ファイルへの印影付与について* 」

データファイル

帳票レイアウトファイルに埋め込むデータを記述したCSVファイル、および、DATファイルについて示します。

コラム

データファイルのサンプルについては、「 *サンプルデータ* 」を参照してください。

注意

連携エンジン IOWebDOC が対応しているデータファイルの文字コードは、次の通りです。

- IOWebDOC 3.x ： UTF-8（BOMあり）
- IOWebDOC 1.x ： Shift_JIS

文字コードが UTF-8（BOMあり） の場合は、実装水準1に対応しています。また、結合文字は含まれません。

注意

画像データのファイルパスを指定する場合は、絶対パスで指定してください。

CSVファイル

CSVファイルは、CDDファイルで定義した内容と合わせるように、識別子にセットするデータをカンマ区切り、または、TAB区切りで記述したファイルです。

フォーマットの例は、次の通りです。

```
IOWebDOC V2.0 Windows版,10,304500,
IOWebDOC Windows版 年間保守,10,456750,
PDFコンバータ V3.0,100,4500,
PDFコンバータ 年間保守,100,67500,
PDF製本工房 V1.5,10,17850,
PDF製本工房 年間保守,10,26775,
```

DATファイル

DATファイルは、帳票レイアウトファイルに指定した識別子名と、識別子にセットするデータを記述したファイルです。

フォーマットの例は、次の通りです。

```
no#1 01
data#1 データ０１
atextframe: tf#1 0 0 EOF
ＩＯＷｅｂＤＯＣサンプルのデータ０１
ＩＯＷｅｂＤＯＣ
EOF
barcode#1 BAR01
```

IM-PDFDesigner for Accel Platform を利用したPDFファイル作成の流れ

IM-PDFDesigner for Accel Platform を利用したPDFファイル作成・開発の流れは次の通りです。

帳票レイアウトファイルの作成

PDFファイルの雛形を作成する作業です。

帳票デザインツールを使用し、帳票レイアウトファイルを作成します。

作成した帳票レイアウトファイルは、< %PUBLIC_STORAGE_PATH% >配下の任意のフォルダに設置します。

帳票デザインツールごとの、設置対象の帳票レイアウトファイルは、次の通りです。

帳票デザインツール	**帳票レイアウトファイル**
単票形式 IODOC	iodファイル
連票形式 IODBDOC	ddlファイル
連票形式 IOCELA	clfファイル、または、defファイル

> **コラム**
> 帳票デザインツールの詳細については、「*帳票デザインツール IOWebDOC について*」を参照してください。

帳票レイアウトファイルに渡すデータの準備

帳票レイアウトファイルに埋め込むためのデータを準備する作業です。

帳票レイアウトファイルに埋め込むデータを取得します。

CSVファイル、または、DATファイルを使用してデータを指定する場合は、次の作業を行います。

- 取得したデータを基に、CSVファイルを作成する
- 取得したデータを基に、DATファイルを作成する

作成したデータファイルは、< %PUBLIC_STORAGE_PATH% >配下の任意のフォルダに設置します。

> **コラム**
> 埋め込むデータの指定方法については、「 *埋め込むデータの指定方法について* 」を参照してください。

PDFファイルの作成

IM-PDFDesigner for Accel Platform を使用し、PDFファイルを作成する作業です。

IM-PDFDesigner for Accel Platform のAPIを使用し、PDFファイルを作成するプログラムを作成後、実行します。

プログラムを作成する際、PDFファイルの出力先は、< %PUBLIC_STORAGE_PATH% >配下の任意のフォルダを指定します。

プログラム変数を使用して埋め込むデータを指定する場合は、プログラム変数を使用し、事前に準備したデータを渡すプログラムの作成も必要となります。

> **注意**
> プログラムを作成する際、各ファイルパスは、< %PUBLIC_STORAGE_PATH% >からの相対パスで指定してください。
>
> ファイルパスに関する注意点については「 *ファイルパスの指定方法について注意してください* 」を参照してください。

PDFファイルの作成パターン

IM-PDFDesigner for Accel Platform での、票形式ごとのPDFファイルの作成パターンについて示します。

> **コラム**
> 本項目の作成パターンと同様のプログラムが、 IM-PDFDesigner for Accel Platform のサンプルプログラムとして、本製品に一部同梱されています。
>
> 詳細については、「 *サンプルプログラム* 」を参照してください。

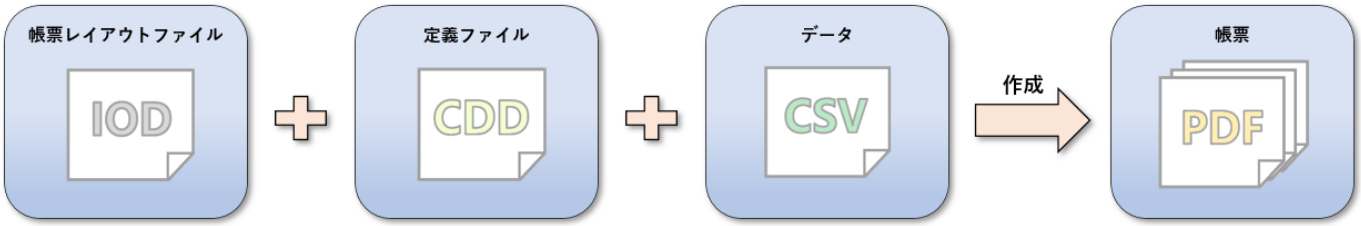
項目

- 単票形式 IODOC
  - CSVファイルを用いて、単票形式のPDFファイルを作成する
  - DATファイルを用いて、単票形式のPDFファイルを作成する
  - プログラム変数を用いて、単票形式のPDFファイルを作成する
- 連票形式 IODBDOC
  - プログラム変数を用いて、連票形式のPDFファイルを作成する
- 結合（単票形式／連票形式／複数の票形式）
  - 中間ファイルを結合し、PDFファイルを作成する

単票形式 IODOC

CSVファイルを用いて、単票形式のPDFファイルを作成する

1. 帳票レイアウトファイルとして、IODファイルを指定します。
2. IODファイルとCSVファイルのデータを関連付けるための定義ファイルとして、CDDファイルを指定します。
3. CSVファイルを使用し、データを指定します。
4. データを埋め込んだPDFファイルの帳票を作成します。

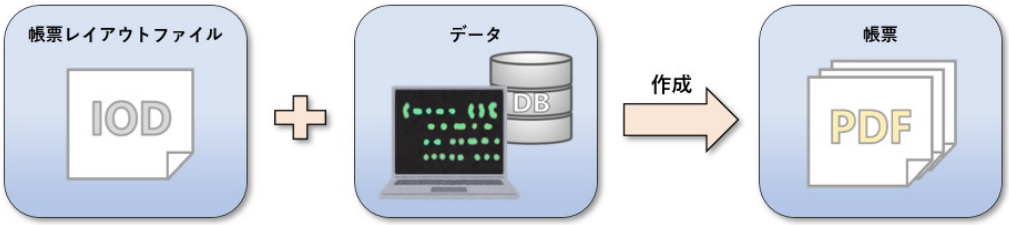


DATファイルを用いて、単票形式のPDFファイルを作成する

1. 帳票レイアウトファイルとして、IODファイルを指定します。
2. DATファイルを使用し、データを指定します。
3. データを埋め込んだPDFファイルの帳票を作成します。

プログラム変数を用いて、単票形式のPDFファイルを作成する

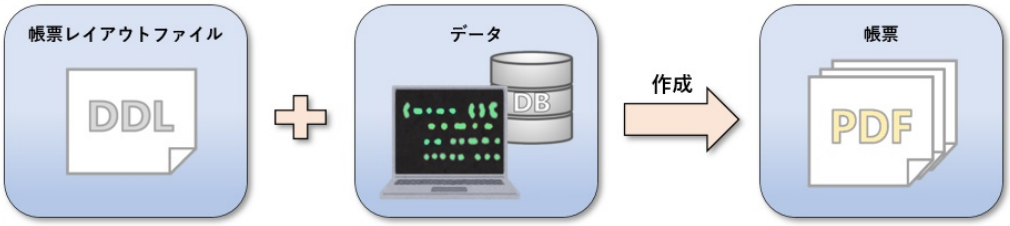
1. 帳票レイアウトファイルとして、IODファイルを指定します。
2. プログラム内部でプログラム変数を使用し、データを指定します。
3. データを埋め込んだPDFファイルの帳票を作成します。



連票形式 IODBDOC

プログラム変数を用いて、連票形式のPDFファイルを作成する

1. 帳票レイアウトファイルとして、DDLファイルを指定します。
2. プログラム内部でプログラム変数を使用し、データを指定します。
3. データを埋め込んだPDFファイルの帳票を作成します。



結合（単票形式／連票形式／複数の票形式）

IM-PDFDesigner for Accel Platform では、PDFファイルだけではなく、帳票レイアウトファイルにデータを埋め込んだIODファイル

（以下中間ファイルと記述します）を作成することも可能です。

作成した中間ファイル同士を結合し、PDFファイルを作成できます。

次のように結合の順番やファイル数を自由に指定することが可能です。

- 表紙（1ページ目）を単票形式 IODOC
- 2ページ目以降を連票形式 IODBDOC

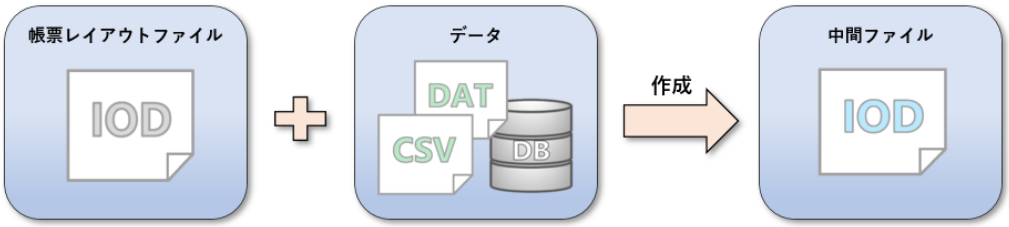
> **コラム**
> 結合される順番は、対象の中間ファイルを追加した順となります。

> **注意**
> 中間ファイルを作成する際に、PDFファイルの文書情報、および、セキュリティを設定しても、その中間ファイルからのPDFファイル作成時には設定が引き継がれません。
>
> 上記設定は、PDFファイルを作成する際に設定してください。

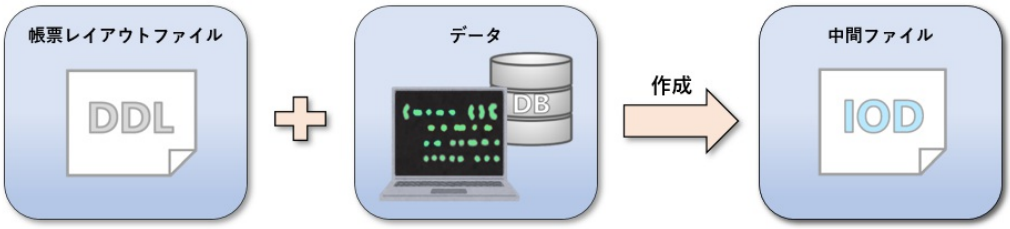
> **注意**
> IM-PDFDesigner for Accel Platform では、PDFファイル同士を結合することはできません。
>
> PDFファイル同士を結合する場合は、 IM-PDFCoordinator for Accel Platform （ページ機能）を利用してください。

中間ファイルを結合し、PDFファイルを作成する

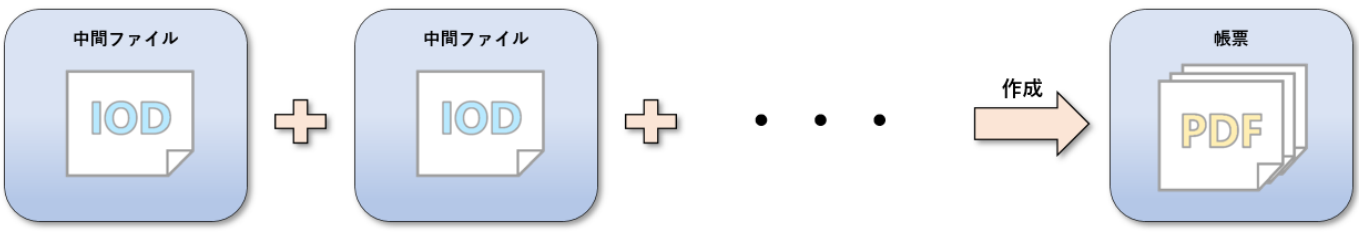
1. 上記PDFファイルの作成手順と同様の手順で、データを埋め込んだ中間ファイル（IODファイル）を作成します。
   - 単票形式の中間ファイルを作成する場合



   - 連票形式の中間ファイルを作成する場合



2. 結合する対象の中間ファイル（IODファイル）を指定します。
3. 中間ファイルを結合し、PDFファイルを作成します。



開発手法

IM-PDFDesigner for Accel Platform のAPIは、intra-mart Accel Platform が標準で提供している他のAPIと同様に使用することができます。

利用可能な開発モデルは、次の通りです。

- スクリプト開発モデル
- JavaEE開発モデル

> **注意**
>
> JavaEE開発モデル 用APIを使用したプログラムをコンパイルする場合、< %RESIN_HOME%/lib >に設置したJARファイルをクラスパスに設定してください。
>
> JARファイルの詳細については「IM-PDFDesigner for Accel Platform セットアップガイド」-「セットアップフォルダ構成」を参照してください。

> **注意**
>
> intra-mart e Builder for Accel Platform との連携方法については、「*intra-mart e Builder for Accel Platform との連携方法*」を参照してください。

APIの概要

IM-PDFDesigner for Accel Platform のAPIの概要は、次の通りです。

各APIの仕様に関しては、「IM-PDFDesigner for Accel Platform API ドキュメント」を参照してください。

- スクリプト開発モデル 用API

用途	クラス（ スタンドアローン構成 ／ 分散構成 ）
単票用	IODoc／IODocRemote
連票用	IODBDoc／IODBDocRemote IOCela／IOCelaRemote
結合用	IOIntegration／IOIntegrationRemote
その他	PdfdRemoteFactory（ 分散構成 のみ）

- JavaEE開発モデル 用API

用途	クラス（ スタンドアローン構成 ／ 分散構成 ）
単票用	CSVDoc／CSVDocRemote
連票用	DBDoc／DBDocRemote CSVCela／CSVCelaRemote
結合用	IOIntegration／IOIntegrationRemote
その他	PdfdRemoteFactory（ 分散構成 のみ）

> **コラム**
>
> スタンドアローン構成 、および、 分散構成 の詳細については、「IM-PDFDesigner for Accel Platform リリースノート」-「スタンドアローン構成と分散構成」を参照してください。

コラム

IM-PDFDesigner for Accel Platform のサンプルプログラムを 分散構成 の環境で使用する場合は、クラス名を変更してください。

```
// スクリプト開発モデル（単票）の場合
// スタンドアローン構成の場合
var pdf = new IODoc("pdfd/tutorial/nouhinkensa.iod", "");
// 分散構成の場合
var pdf = new IODocRemote("pdfd/tutorial/nouhinkensa.iod", "");
```

```
// JavaEE開発モデル（単票）の場合
// スタンドアローン構成の場合
CSVDoc pdf = new CSVDoc("pdfd/tutorial/nouhinkensa.iod", "");
// 分散構成の場合
CSVDocRemote pdf = new CSVDocRemote("pdfd/tutorial/nouhinkensa.iod", "");
```

コラム

分散構成で intra-mart Accel Platform サーバと PDF帳票サーバ の間にロードバランサーを導入している場合、Factory クラスを使用することでセッションを維持できる可能性があります。

```
// このFactoryクラスで生成したオブジェクトは同一セッションになります
PdfdRemoteFactory factory = new PdfdRemoteFactory();

CSVDocRemote iodoc = factory.createCSVDocRemote("sample1.iod", null);
iodoc.makeIOD("sample1_temp.iod");

CSVDocRemote iodoc = factory.createCSVDocRemote("sample2.iod", null);
iodoc.makeIOD("sample2_temp.iod");

IOIntegrationRemote integration = factory.createIOIntegrationRemote();
integration.add("sample1_temp.iod");
integration.add("sample2_temp.iod");
integration.makePDF("result.pdf");
```

プログラム開発における注意点

項目

- 全般
  - ファイルサイズの大きいPDFファイルは、出力処理に時間がかかる場合があります
  - ファイルパスの指定方法について注意してください
  - ファイルの出力先の競合に注意してください
  - IM-PDFDesigner for Accel Platform で廃止されたAPIは利用しないでください
- 連票形式 IODBDOC
  - カラムデータとしてnullを指定しないでください
  - 定義したカラム数と同数のカラムデータを設定してください

全般

ファイルサイズの大きいPDFファイルは、出力処理に時間がかかる場合があります

ファイルサイズの大きいPDFファイルを作成する際、出力処理に時間がかかり、APIのレスポンスとPDFファイルがディスク上に完全に書き出されるタイミングが大きく異なる場合があります。

サイズの大きいPDFファイルを作成する場合は、出力処理の時間を考慮し、十分な時間が経過した後に、出力されたPDFファイルにアクセスするようにしてください。

ファイルパスの指定方法について注意してください

帳票レイアウトファイルやデータファイルは、< %PUBLIC_STORAGE_PATH% >からの相対パスで指定してください。

ただし、一部の項目では、相対パスではなく絶対パスでの指定が必要です。

次のメソッド、および、データファイル内で画像のファイルパスを指定する場合は、絶対パスで指定してください。

開発モデル	クラス	メソッド
スクリプト開発モデル	IODoc／IODocRemote	setData(id, data)
	IODBDoc／IODBDocRemote	setCol(value) setGlobal(name, value)
	IOCela／IOCelaRemote	setData(id, data)
JavaEE開発モデル	CSVDoc／CSVDocRemote	setData(id, data) setRecord(datarecord)
	DBDoc／DBDocRemote	setCol(value) setGlobal(name, value)
	CSVCela／CSVCelaRemote	setData(id, data)

ファイルの出力先の競合に注意してください

IODファイル、または、PDFファイルを作成する際、指定したパスにファイルが既に存在している場合でも、ファイルは上書きされ、エラーは発生しません。

ただし、ファイルを別プロセスで使用している場合は、エラーが発生します。

ファイルの出力先の競合を回避するためには、重複しない出力ファイル名を使用する等、上位アプリケーション側でファイルの排他制御を行ってください。

IM-PDFDesigner for Accel Platform で廃止されたAPIは利用しないでください

IM-PDF デザイナー Ver.7.x の一部のAPIは、 IM-PDFDesigner for Accel Platform では廃止されています。

IM-PDFDesigner for Accel Platform で廃止されたAPIは利用しないでください。

廃止されたAPIの詳細については「*IM-PDFDesigner for Accel Platform で廃止されたAPI*」を参照してください。

連票形式 IODBDOC

カラムデータとしてnullを指定しないでください

カラムデータを設定する際、setColに、nullを指定しないでください。

nullを指定した場合、PDFファイル作成時にエラーが発生します。

定義したカラム数と同数のカラムデータを設定してください

setColを使用する際は、帳票レイアウトファイルで定義したカラムと同数のカラムデータが設定されるように指定してください。

カラムとデータの数が一致していない場合、PDFファイル作成時にエラーが発生します。

帳票デザインツール IOWebDOC について

帳票デザインツール IOWebDOC について説明します。

帳票デザインツール概要

帳票デザインツール IOWebDOC は、帳票レイアウトファイルを作成するためのGUIアプリケーションです。

項目

- 帳票デザインツール一覧
  - 単票形式 IODOC
  - 連票形式 IODBDOC
  - 連票形式 IOCELA

帳票デザインツール一覧

帳票デザインツールでは、単票形式、連票形式の帳票レイアウトファイルを作成することが可能です。

各形式に対応した帳票デザインツールは、次の通りです。

帳票形式	帳票デザインツール名	ファイル拡張子
単票形式	IODOC	.dlf／.iod
連票形式	IODBDOC IOCELA	.ddl .clf／.def

コラム

連票形式 IOCELA のファイル拡張子は、.def／.clf の2種類があります。

DEFファイル、CLFファイルともに帳票レイアウトファイルです。

注意

単票形式 IODOC のファイル拡張子は、.dlf／.iod の2種類があります。

DLFファイルは、帳票レイアウト作成用のファイルです。

IODファイルは、DLFファイルを基にした帳票レイアウトファイルです。

帳票デザインツール IODOC で帳票レイアウトを編集する場合は、DLFファイルが必要です。

PDFファイルを作成する場合は、IODファイルが必要です。

注意

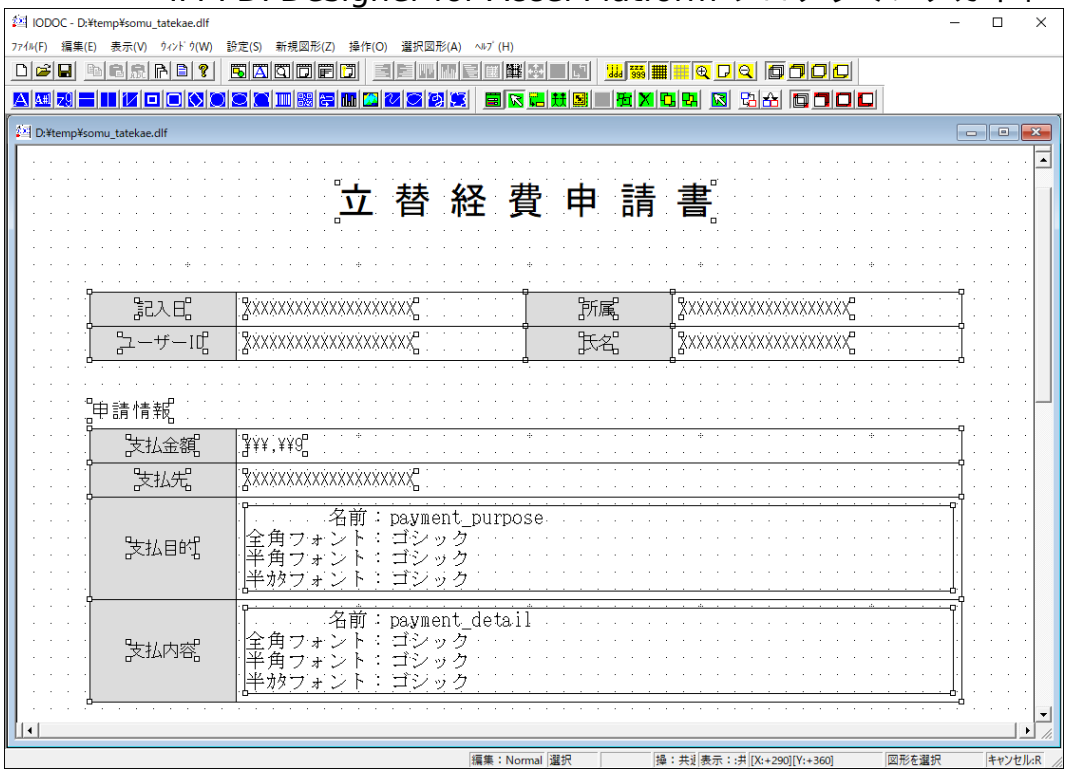
連票用の帳票デザインツールとして、 IODBDOC 、および、 IOCELA の2種類のツールがあります。

連票形式の帳票レイアウトファイルを新規で作成する場合は、 IODBDOC を利用してください。

IOCELA は互換性のために残されています。

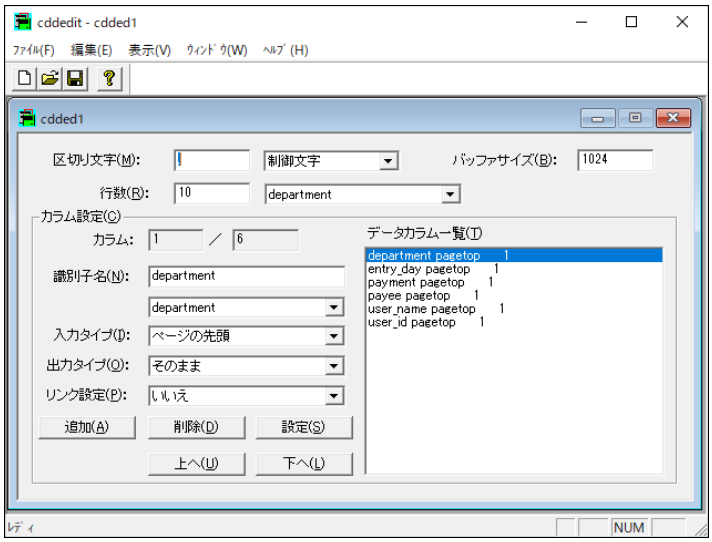
単票形式 IODOC

単票形式 IODOC は、単票形式の帳票レイアウトファイルを作成するためのツールです。



CSV形式のデータファイルを指定してPDFファイルを作成する場合、帳票レイアウトファイルとCSVファイルのデータを関連付ける定義ファイル（以下CDDファイルと記述します）が必要となります。

CDDファイルは、 帳票デザインツール IOWebDOC の CDDエディタ で作成します。



> **コラム**
> 帳票デザインツール（単票形式 IODOC ）の補足情報については、「 *帳票デザインツール IODOC 補足情報* 」を参照してください。

> **コラム**
> 帳票デザインツール（単票形式 IODOC ）、および、 CDDエディタ の使用方法については、「 *単票形式 IODOC* 」を参照してください。

> **注意**
> IM-PDFDesigner for Accel Platform を Linux 環境で利用する場合、単票形式のレイアウトを保存する際、保存形式は「V4.9形式」を選択してください。
>
> 「V.4.9.3形式」で保存した場合、処理実行時に「-100 ファイルアクセスエラー」になります。

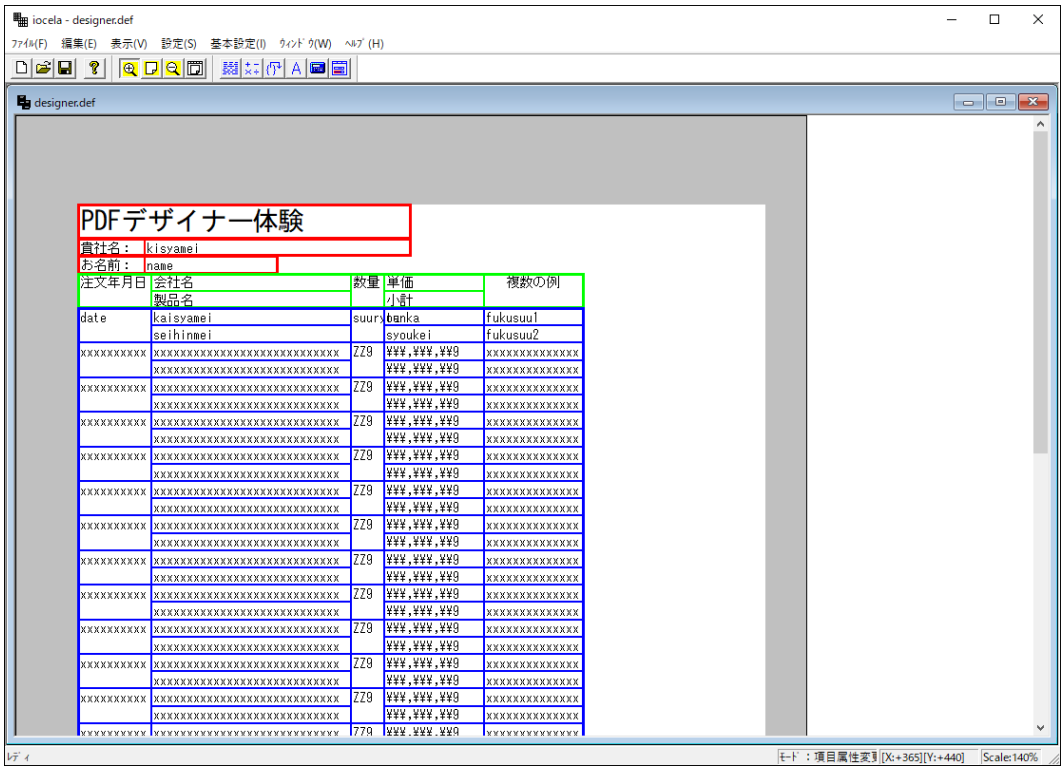
連票形式 IODBDOC

連票形式 IODBDOC は、連票形式の帳票レイアウトファイルを作成するためのツールです。

> **コラム**
> 帳票デザインツール（連票形式 IODBDOC ）の使用方法については、「*連票形式 IODBDOC* 」を参照してください。

連票形式 IOCELA

連票形式 IOCELA は、連票形式の帳票レイアウトファイルを作成するためのツールです。



> **コラム**
> 連票形式 IOCELA は、設定ファイルで出力を制御することが可能です。
>
> 詳細については、「*連票形式 IOCELA の出力カスタマイズ*」を参照してください。

コラム
帳票デザインツール（連票形式 IOCELA ）の使用方法については、「 *連票形式 IOCELA* 」を参照してください。

帳票デザインツールに関する注意点

帳票デザインツール IOWebDOC は、 Windows 環境で動作します

帳票デザインツール IOWebDOC は、 Windows 環境で動作します。

IM-PDFDesigner for Accel Platform を Linux 環境で利用する場合でも、 帳票デザインツール IOWebDOC は Windows 環境にインストールし、帳票レイアウトファイルを作成してください。

適切な余白を確保して帳票レイアウトファイルを作成してください

帳票レイアウトファイルを作成する際、適切な余白（最低5mm以上）を確保してください。

通常、プリンタには印字可能な範囲（プリンタのヘッドが動ける範囲＝余白）があります。

用紙サイズ目いっぱいにデザインすると、プリンタの余白の制限で印字されない部分が出る場合があります。

利用可能なフォントは、 固定幅フォントです

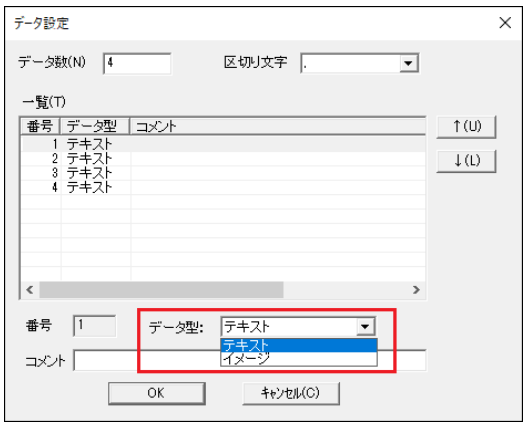
帳票デザインツールで利用可能なフォントは、 固定幅フォントです。

プロポーショナルフォントは利用できません。

IODBDOC のデータ設定画面でのデータ型は「テキスト」を選択してください

IODBDOC で帳票レイアウトファイルを作成する際、データ設定画面でのデータ型は、画像を埋め込む場合でも必ず「テキスト」を選択してください。

データ型の「イメージ」は、複数行に同じ画像を効率的に指定するためのものでしたが、現在は非推奨で、以降のバージョンから削除する可能性があります。



帳票デザインツールマニュアル

単票形式 IODOC

- **IODOCマニュアル**
- **IODOC簡易マニュアル**
- **CDDEDIT操作説明書**

連票形式 IODBDOC

- **IODBDOCマニュアル**
- **IODBDOC簡易マニュアル**

連票形式 IOCELA

- **IOCELAマニュアル**
- **IOCELA簡易マニュアル**

サンプル

サンプルプログラム

本製品には、 IM-PDFDesigner for Accel Platform のサンプルプログラムが同梱されています。

サンプルプログラムを利用するには、 IM-PDFDesigner for Accel Platform のセットアップ時、WARファイルを出力する際に「サンプルデータを含める」へのチェックが必要です。

サンプルプログラム・データの設置場所

サンプルを含むWARファイルのデプロイ、および、テナント環境のセットアップを行うことで、サンプルプログラム・データが設置されます。

セットアップに関する各手順については、次を参照してください。

- 「intra-mart Accel Platform セットアップガイド」-「WARファイルのデプロイ 」
- 「intra-mart Accel Platform セットアップガイド」-「テナント環境セットアップ 」

項目
- サンプルプログラム
  - スクリプト開発モデル
  - JavaEE開発モデル
- サンプルデータ

サンプルプログラム

サンプルプログラムが設置される場所は、次の通りです。

スクリプト開発モデル

- < %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/jssp/src/pdfd/sample/ >

帳票形式	帳票デザインツール	埋め込むデータの指定方法	ファイル名	説明
単票形式	IODOC	CSVファイル	doccsv.html doccsv.js	CSVファイルを用いて単票形式のPDFファイルを作成するサンプル
		DATファイル	docdat.html docdat.js	DATファイルを用いて単票形式のPDFファイルを作成するサンプル
		プログラム変数	docobj.html docobj.js	メモリデータを用いて単票形式のPDFファイルを作成するサンプル
連票形式	IODBDOC	プログラム変数	dbdocobj.html dbdocobj.js	メモリデータを用いて連票形式のPDFファイルを作成するサンプル
			dbdocmulti.html dbdocmulti.js	複数表形式のPDFファイルを作成するサンプル
	IOCELA	CSVファイル	celacsv.html celacsv.js	CSVファイルを用いて連票形式のPDFファイルを作成するサンプル
			celacsvdat.html celacsvdat.js	帳票レイアウトを重ね合わせ、CSVファイルを用いてPDFファイルを作成するサンプル
		プログラム変数	celarec.html celarec.js	レコードデータを用いて連票形式のPDFファイルを作成するサンプル
			celarecobj.html celarecobj.js	帳票レイアウトを重ね合わせ、メモリデータを用いてPDFファイルを作成するサンプル

帳票形式	帳票デザインツール	埋め込むデータの指定方法	ファイル名	説明
その他	IODOC ／ IOCELA	CSVファイル	integration.html integration.js	異なる形式の帳票レイアウトを結合し、PDFファイルを作成するサンプル

- < %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/jssp/src/pdfd/tutorial/ >

帳票形式	帳票デザインツール	埋め込むデータの指定方法	ファイル名	説明
単票形式	IODOC	CSVファイル	docsample.html docsample.js	帳票レイアウトを重ね合わせ、CSVファイルを用いてPDFファイルを作成するサンプル
連票形式	IODBDOC	プログラム変数	dbdocsample.html dbdocsample.js	メモリデータを用いて連票形式のPDFファイルを作成するサンプル
	IOCELA	CSVファイル	celasample.html celasample.js	CSVファイルを用いて連票形式のPDFファイルを作成するサンプル
その他	IODOC ／ IOCELA	CSVファイル	iointegration.html iointegration.js	異なる形式の帳票レイアウトを結合し、PDFファイルを作成するサンプル

JavaEE開発モデル

- < %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/sample/ >

帳票形式	帳票デザインツール	埋め込むデータの指定方法	ファイル名	説明
単票形式	IODOC	CSVファイル	doccsv.jsp doccsv_act.jsp	CSVファイルを用いて単票形式のPDFファイルを作成するサンプル
		DATファイル	docdat.jsp docdat_act.jsp	DATファイルを用いて単票形式のPDFファイルを作成するサンプル
		プログラム変数	docobj.jsp docobj_act.jsp	メモリデータを用いて単票形式のPDFファイルを作成するサンプル
連票形式	IODBDOC	プログラム変数	dbdocobj.jsp dbdocobj_act.jsp	メモリデータを用いて連票形式のPDFファイルを作成するサンプル
			dbdocmulti.jsp dbdocmulti_act.jsp	複数表形式のPDFファイルを作成するサンプル
	IOCELA	CSVファイル	celacsv.jsp celacsv_act.jsp	CSVファイルを用いて連票形式のPDFファイルを作成するサンプル
			celacsvdat.jsp celacsvdat_act.jsp	帳票レイアウトを重ね合わせ、CSVファイルを用いてPDFファイルを作成するサンプル
		プログラム変数	celarec.jsp celarec_act.jsp	レコードデータを用いて連票形式のPDFファイルを作成するサンプル
			celarecobj.jsp celarecobj_act.jsp	帳票レイアウトを重ね合わせ、メモリデータを用いてPDFファイルを作成するサンプル
その他	IODOC ／ IOCELA	CSVファイル	integration.jsp integration_act.jsp	異なる形式の帳票レイアウトを結合し、PDFファイルを作成するサンプル

- < %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/ >

帳票形式	帳票デザインツール	埋め込むデータの指定方法	ファイル名	説明
単票形式	IODOC	CSVファイル	docsample.jsp docsample_act.jsp	帳票レイアウトを重ね合わせ、CSVファイルを用いてPDFファイルを作成するサンプル
連票形式	IODBDOC	プログラム変数	dbdocsample.jsp dbdocsample_act.jsp	メモリデータを用いて連票形式のPDFファイルを作成するサンプル
	IOCELA	CSVファイル	celasample.jsp celasample_act.jsp	CSVファイルを用いて連票形式のPDFファイルを作成するサンプル
その他	IODOC ／ IOCELA	CSVファイル	iointegration.jsp iointegration_act.jsp	異なる形式の帳票レイアウトを結合し、PDFファイルを作成するサンプル

サンプルデータ

上記のサンプルプログラムで使用するサンプルデータ（帳票レイアウト、データファイル、および、画像ファイルなど）が設置される場所は、次の通りです。

- < %PUBLIC_STORAGE_PATH% /pdfd/ >

フォルダ名	説明
tutorial	チュートリアル用データー式
webdoc	単票形式（ IODOC ）のPDFファイルを作成するサンプル用データー式
webdbdoc	連票形式（ IODBDOC ）のPDFファイルを作成するサンプル用データー式
webcela	連票形式（ IOCELA ）のPDFファイルを作成するサンプル用データー式
integration	異なる形式の帳票レイアウトを結合し、PDFファイルを作成するサンプル用のデーター式

サンプルプログラムに関する注意点

サンプルプログラムは、スレッドセーフではありません

サンプルプログラムは、スレッドセーフではありません。

複数のスレッドで同時にサンプルプログラムを実行した場合、正しくPDFファイルを作成できない場合があります。

サンプルプログラムでは、エラー処理を単純化しています

プログラムの理解に重点を置いているため、サンプルプログラムでは、エラー処理を単純化して簡潔に記述しています。

PDFファイルに埋め込むデータは固定値です

サンプルプログラムでは、PDFファイルに埋め込むデータは固定値を使用しています。

出力内容の変化を確認したい場合は、サンプルプログラムの処理を変更してください。

指定したパスにPDFファイルが既に存在している場合、PDFファイルが上書きされます

一部のサンプルプログラムでは、指定したパスにPDFファイルが既に存在している場合、元のPDFファイルは失われ、新規作成したPDFファイルに上書きされます。

PDFファイルを上書きさせないためには、重複しない出力ファイル名を使用する等、サンプルプログラムの処理を変更してください。

バイナリ形式のコンテンツを扱う場合、実運用ではサーブレットを使用してください

説明を簡素化するため、サンプルプログラムではJSPからファイルダウンロードを行っていますが、実運用ではサーブレットを使用して

ください。

JSPはテキスト形式のコンテンツを扱う方式のため、画像やファイルのダウンロードといったバイナリ形式のコンテンツを扱う処理は、サーブレットを使っての実装が推奨されます。

IOWebDOC 1.x の場合、データファイルの文字コードを Shift_JIS に変更してください

連携エンジン IOWebDOC 1.x が対応しているデータファイルの文字コードは Shift_JIS です。

サンプルプログラムのデータファイル（CSVファイル、DATファイル）は UTF-8（BOMあり） で作成しているため、連携エンジン IOWebDOC 1.x を利用する場合は、データファイルの文字コードを Shift_JIS に変更してください。

サンプルプログラムの実行方法

IM-PDFDesigner for Accel Platform のサンプルプログラムは、 IM-PDFDesigner for Accel Platform のセットアップ後、サンプルデータをセットアップすることで実行できます。

サンプルデータのセットアップ、および、実行方法は、「 IM-PDFDesigner for Accel Platform セットアップガイド」-「 サンプルデータ 」を参照してください。

> **コラム**
> サンプルプログラムは、画面プログラムとして作成されていますが、バッチプログラム内でも同様に（本製品で提供されている）PDFファイル作成用APIを利用することができます。

> **注意**
> サンプルプログラムを利用する上での、 IM-PDFDesigner for Accel Platform のセットアップの詳細については、「IM-PDFDesigner for Accel Platform セットアップガイド 」を参照してください。

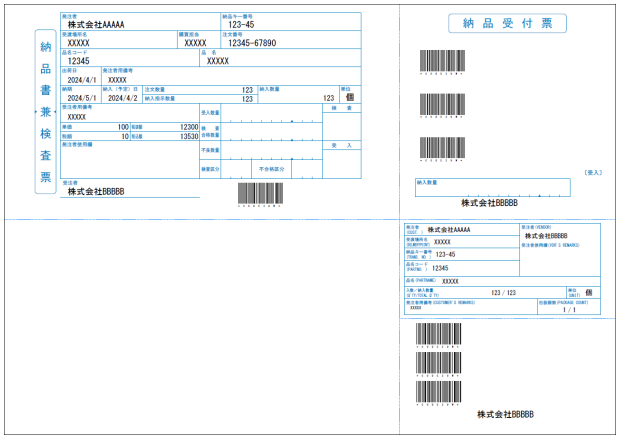
サンプル帳票レイアウト

IM-PDFDesigner for Accel Platform のサンプル帳票レイアウトを紹介します。

単票形式 IODOC

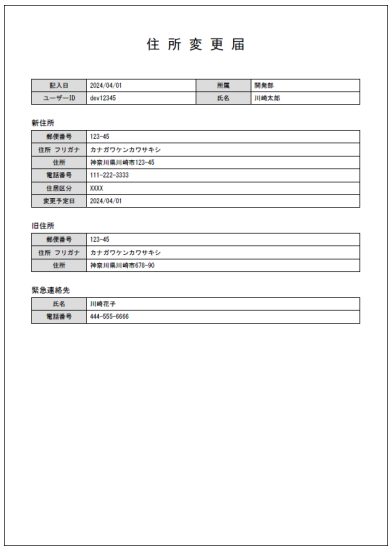
納品書

- **帳票レイアウト作成ファイル**
- **帳票レイアウトファイル**
- **PDF帳票**



住所変更届

- **帳票レイアウト作成ファイル**
- **帳票レイアウトファイル**
- **PDF帳票**

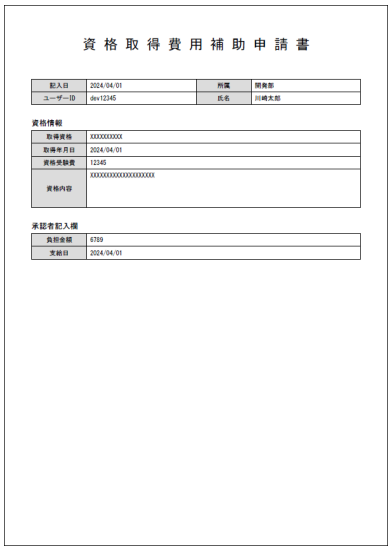


給与・賞与受領口座申告書

- **帳票レイアウト作成ファイル**
- **帳票レイアウトファイル**
- **PDF帳票**

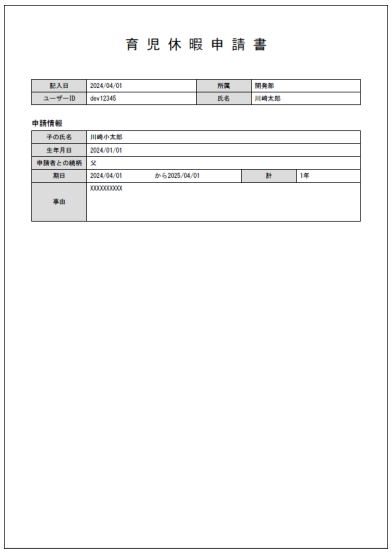
資格取得費用補助申請書

- **帳票レイアウト作成ファイル**
- **帳票レイアウトファイル**
- **PDF帳票**



育児休暇申請書

- 帳票レイアウト作成ファイル
- 帳票レイアウトファイル
- PDF帳票


育 児 休 暇 申 請 書

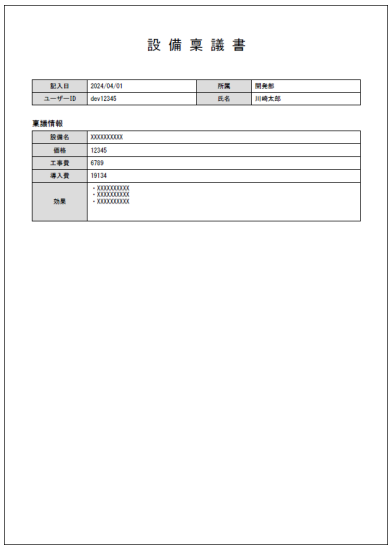
記入日	2024/04/01	所属	開発部
ユーザーID	dev12345	氏名	川崎太郎

申請情報

子の氏名	川崎小太郎		
生年月日	2024/01/01		
申請者との続柄	父		
期日	2024/04/01	から2025/04/01	計 1年
事由	XXXXXXXXXX		

設備稟議書

- 帳票レイアウト作成ファイル
- 帳票レイアウトファイル
- PDF帳票


設 備 稟 議 書

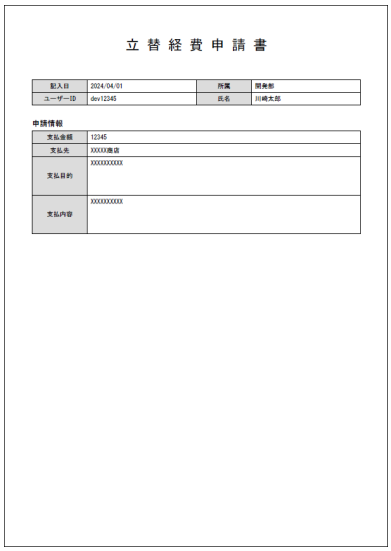
記入日	2024/04/01	所属	開発部
ユーザーID	dev12345	氏名	川崎太郎

稟議情報

設備名	XXXXXXXXXX
価格	12345
工事費	6789
導入費	19134
効果	・XXXXXXXXXX ・XXXXXXXXXX ・XXXXXXXXXX

立替経費申請書

- 帳票レイアウト作成ファイル
- 帳票レイアウトファイル
- PDF帳票


立 替 経 費 申 請 書

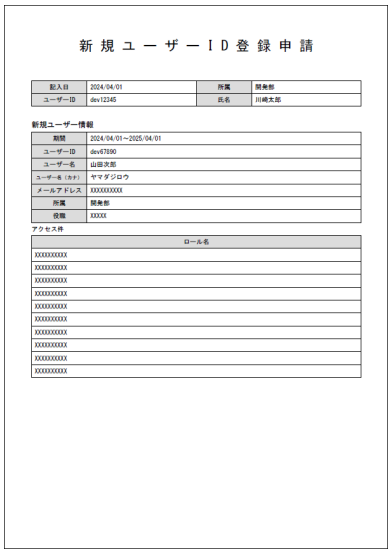
記入日	2024/04/01	所属	開発部
ユーザーID	dev12345	氏名	川崎太郎

申請情報

支払金額	12345
支払先	XXXXX商店
支払目的	XXXXXXXXXX
支払内容	XXXXXXXXXX

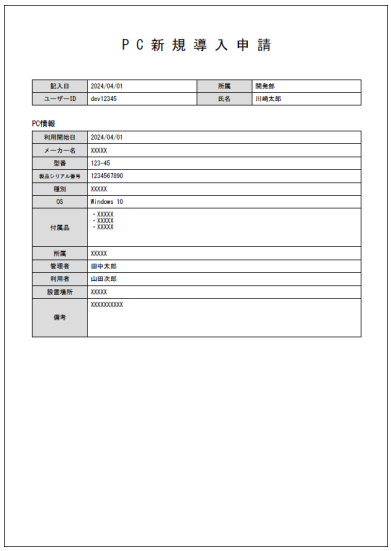
新規ユーザーID登録申請

- **帳票レイアウト作成ファイル**
- **帳票レイアウトファイル**
- **PDF帳票**



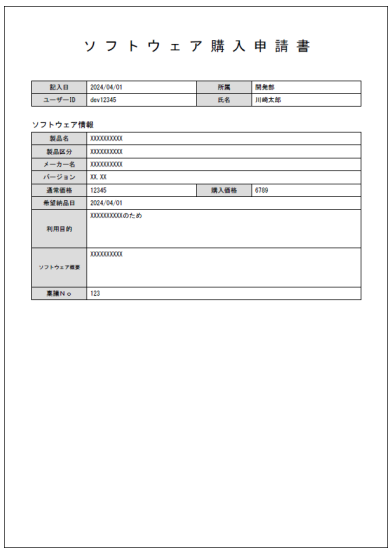
PC新規導入申請

- **帳票レイアウト作成ファイル**
- **帳票レイアウトファイル**
- **PDF帳票**



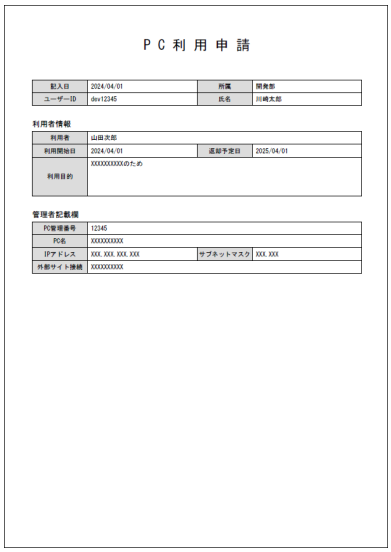
ソフトウェア購入申請

- **帳票レイアウト作成ファイル**
- **帳票レイアウトファイル**
- **PDF帳票**



PC利用申請

- **帳票レイアウト作成ファイル**
- **帳票レイアウトファイル**
- **PDF帳票**



P C 利 用 申 請

記入日	2024/04/01	所属	開発部
ユーザーID	dev12345	氏名	川崎太郎

利用者情報

利用者	山田次郎		
利用開始日	2024/04/01	返却予定日	2025/04/01
利用目的	XXXXXXXXXXのため		

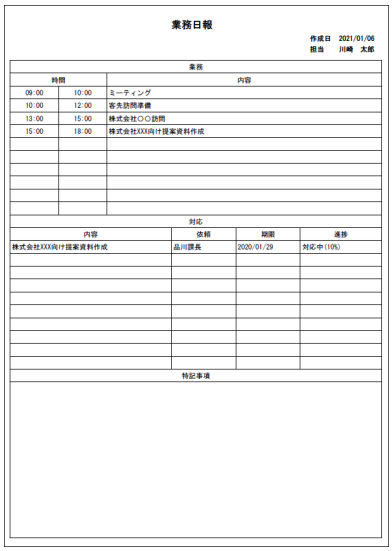
管理者記載欄

PC管理番号	12345		
PC名	XXXXXXXXX		
IPアドレス	XXX.XXX.XXX.XXX	サブネットマスク	XXX.XXX
外部サイト接続	XXXXXXXXX		

連票形式 IODBDOC

業務日報

- **帳票レイアウトファイル**
- **帳票レイアウトファイル（レイアウト切り替え用）**
- **PDF帳票**



業務日報

作成日　2021/01/06
担当　川崎　太郎

業務

時間		内容
09:00	10:00	ミーティング
10:00	12:00	客先訪問準備
13:00	15:00	株式会社○○訪問
15:00	18:00	株式会社XXX向け提案資料作成

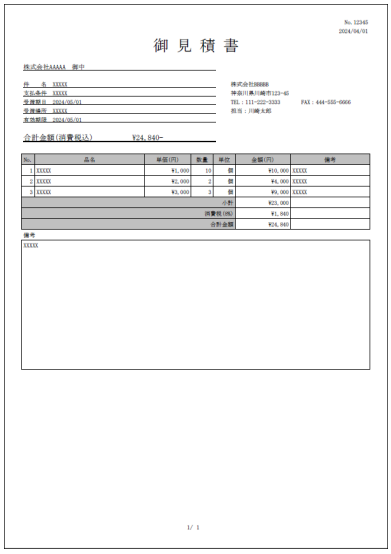
対応

内容	依頼	期限	進捗
株式会社XXX向け提案資料作成	品川課長	2020/01/29	対応中(10%)

特記事項

見積書

- **帳票レイアウトファイル**
- **PDF帳票**



No. 12345
2024/04/01

御 見 積 書

株式会社AAAAA　御中

件　名　XXXXX
支払条件　XXXXX
有効期日　2024/05/01
受渡場所　XXXXX
有効期限　2024/05/01

株式会社SSSSS
神奈川県川崎市123-45
TEL：111-222-3333　FAX：444-555-6666
担当：川崎太郎

合計金額(消費税込)　¥24,840-

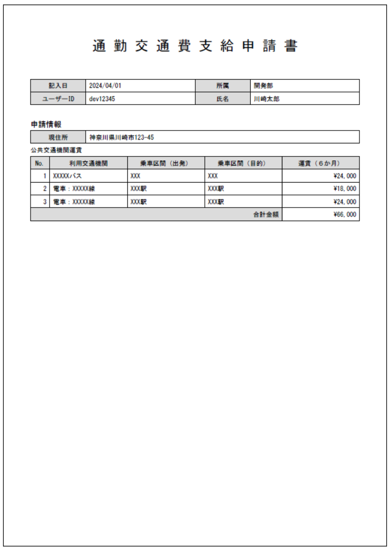
No.	品名	単価(円)	数量	単位	金額(円)	備考
1	XXXXX	¥1,000	10	個	¥10,000	XXXXX
2	XXXXX	¥2,000	2	個	¥4,000	XXXXX
3	XXXXX	¥3,000	3	個	¥9,000	XXXXX
					小計	¥23,000
					消費税(8%)	¥1,840
					合計金額	¥24,840

備考
XXXXX

1/ 1

通勤交通費支給申請書

- 帳票レイアウトファイル
- PDF帳票



通 勤 交 通 費 支 給 申 請 書

記入日	2024/04/01	所属	開発部
ユーザーID	dev12345	氏名	川崎太郎

申請情報

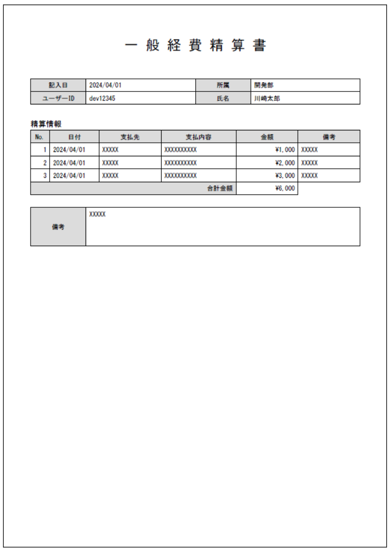
現住所	神奈川県川崎市123-45

公共交通機関運賃

No.	利用交通機関	乗車区間（出発）	乗車区間（目的）	運賃（６か月）
1	XXXXXバス	XXX	XXX	¥24, 000
2	電車：XXXXX線	XXX駅	XXX駅	¥18, 000
3	電車：XXXXX線	XXX駅	XXX駅	¥24, 000
			合計金額	¥66, 000

一般経費精算書

- 帳票レイアウトファイル
- PDF帳票



一 般 経 費 精 算 書

記入日	2024/04/01	所属	開発部
ユーザーID	dev12345	氏名	川崎太郎

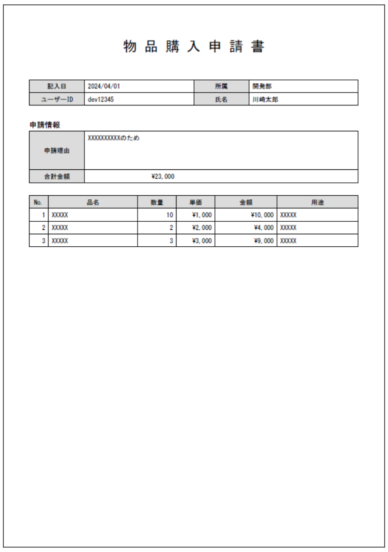
精算情報

No.	日付	支払先	支払内容	金額	備考
1	2024/04/01	XXXXX	XXXXXXXXXX	¥1, 000	XXXXX
2	2024/04/01	XXXXX	XXXXXXXXXX	¥2, 000	XXXXX
3	2024/04/01	XXXXX	XXXXXXXXXX	¥3, 000	XXXXX
			合計金額	¥6, 000	

備考	XXXXX

物品購入申請書

- 帳票レイアウトファイル
- PDF帳票



物 品 購 入 申 請 書

記入日	2024/04/01	所属	開発部
ユーザーID	dev12345	氏名	川崎太郎

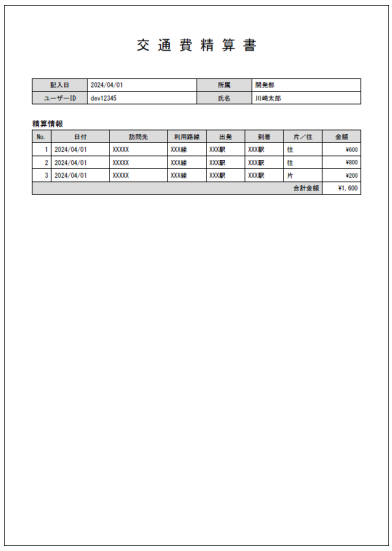
申請情報

申請理由	XXXXXXXXXXのため
合計金額	¥23, 000

No.	品名	数量	単価	金額	用途
1	XXXXX	10	¥1, 000	¥10, 000	XXXXX
2	XXXXX	2	¥2, 000	¥4, 000	XXXXX
3	XXXXX	3	¥3, 000	¥9, 000	XXXXX

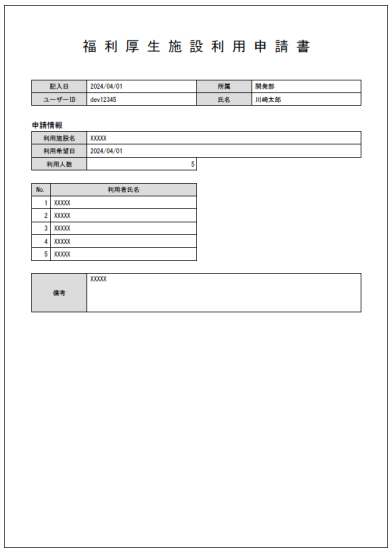
交通費精算書

- **帳票レイアウトファイル**
- **PDF帳票**



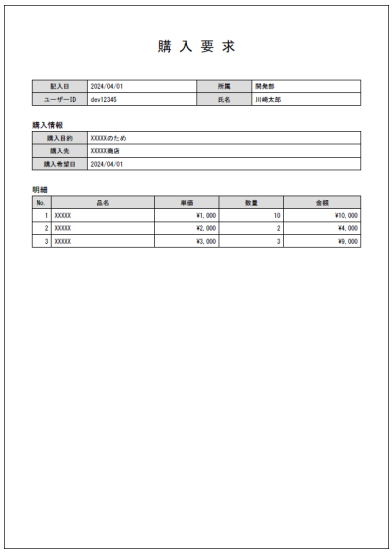
福利厚生施設利用申請書

- **帳票レイアウトファイル**
- **PDF帳票**



購入要求

- **帳票レイアウトファイル**
- **PDF帳票**



チュートリアル

基本

- 概要
- 前提条件
- 実践

概要

本チュートリアルでは、プログラミング言語を使用したPDFファイルの基本的な作成方法を説明します。

前提条件

本チュートリアルを進めるにあたり、次の事前準備が行われていることが前提となります。

- IM-PDFDesigner for Accel Platform 、または、 IM-PDFDesigner FullPack for Accel Platform のセットアップが完了していること。
- intra-mart Accel Platform のテナント環境セットアップが完了していること。

実践

単票形式 IODOC

本チュートリアルでは単票形式のPDFファイルを作成します。

チュートリアルを実施するにあたり、次のzipファイルをダウンロードし、解凍してください。

< **tutorial_basic_iodoc.zip** >

解凍したファイルの構成は、次の通りです。

フォルダ名／ファイル名	説明
layout/	帳票レイアウト関連フォルダ
somu_tatekae.dlf	帳票レイアウト作成用ファイル
somu_tatekae_sample.dlf	帳票レイアウト作成用ファイル（参考用）
somu_tatekae_sample.iod	帳票レイアウトファイル（参考用）
somu_tatekae_sample.dat	帳票レイアウトファイルに埋め込むデータファイル（参考用）
jssp/	スクリプト開発モデル 用フォルダ
iodoc.html iodoc.js	スクリプト開発モデル 用プログラム
pdfd-tutorial_basic_iodoc.xml	スクリプト開発モデル 用設定ファイル
javaee/	JavaEE開発モデル 用フォルダ
iodoc.jsp iodoc_act.jsp download.jsp	JavaEE開発モデル 用プログラム
service-config-pdfd_tutorial_basic_iodoc.xml pdfd-tutorial_basic_iodoc.xml	JavaEE開発モデル 用設定ファイル

次の手順に沿って、チュートリアルを進めます。

帳票レイアウトファイルの作成

帳票デザインツール IODOC を使用し、PDFファイルの雛形を作成する作業です。

本項目では、未完成の帳票レイアウト作成用ファイルに識別子を設定後、帳票レイアウトファイルを作成します。

識別子とは、帳票レイアウトファイル上に、文字列や画像などのデータを埋め込むための項目です。

PDFファイル作成時、識別子とデータを紐づけることで、帳票レイアウトファイル上に設定した識別子にデータが埋め込まれます。

未完成の帳票レイアウト作成用ファイル< layout/somu_tatekae.dlf >で、識別子が未設定の項目は、次の通りです。

- ユーザーID
- 氏名
- 支払金額
- 支払先

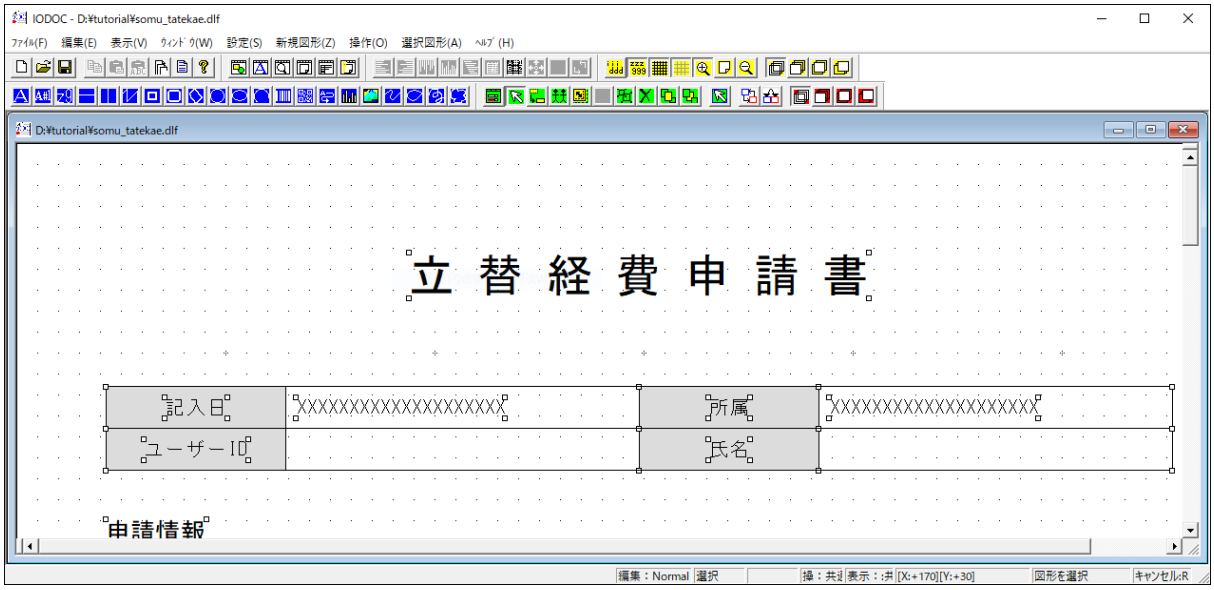
> **コラム**
> 帳票デザインツール（単票形式 IODOC ）の詳しい使用方法については、「 *単票形式 IODOC* 」を参照してください。

手順

- 帳票レイアウト作成用ファイルを開く
- 識別子を追加する
- 識別子を編集する
- 識別子の位置を調整する
- 残りの識別子を全て設定する
- 名前を付けて保存する
- 作成した帳票レイアウトファイルを設置する

帳票レイアウト作成用ファイルを開く

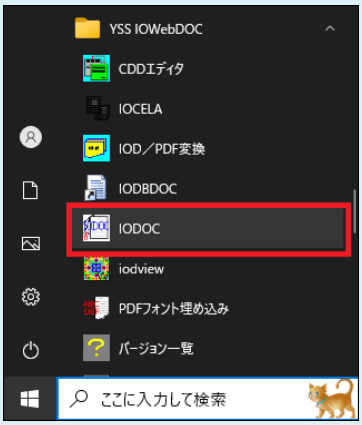
1. 帳票デザインツール IOWebDOC をインストールしているPCに< layout/somu_tatekae.dlf >をコピーします。
2. コピーした< somu_tatekae.dlf >をダブルクリックします。
   帳票デザインツール IODOC が起動し、ファイルが開きます。



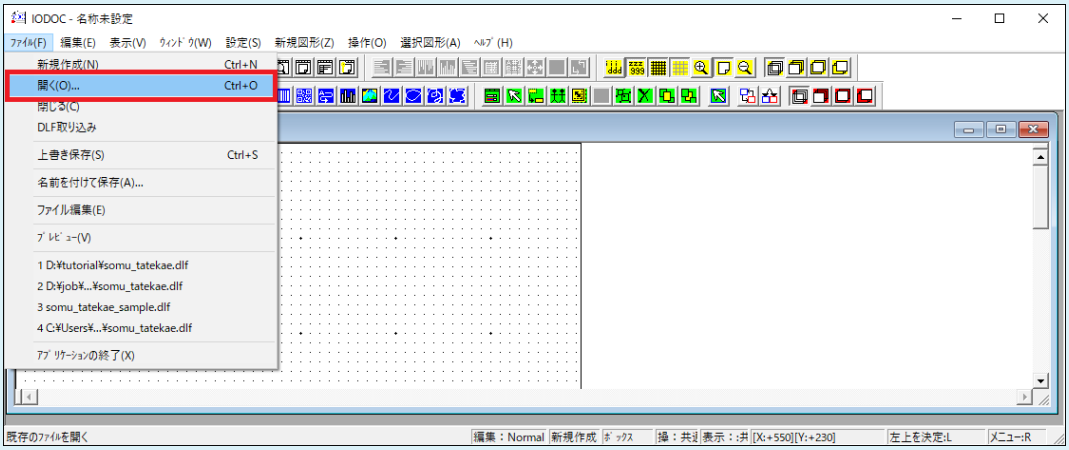
コラム

上記の手順以外に、次の手順でもファイルを開くことが可能です。

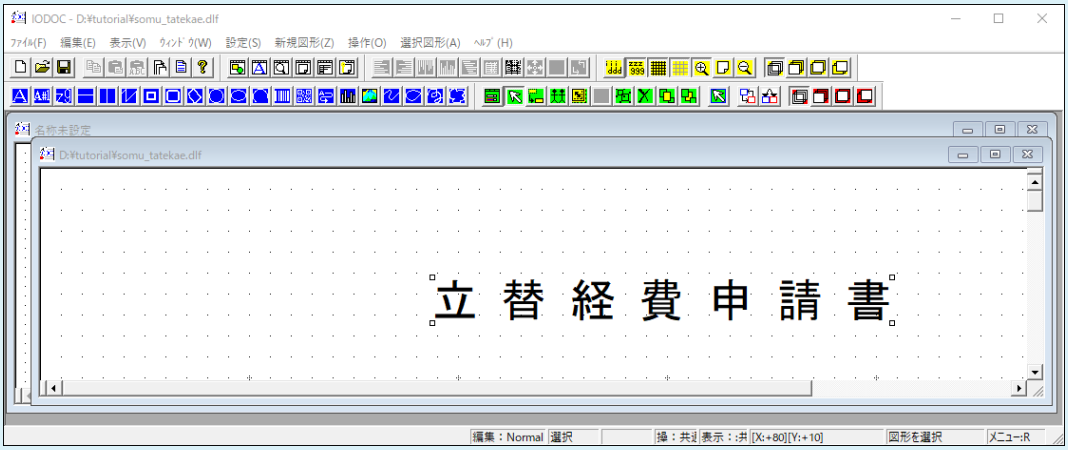
1. 「スタート」-「 YSS IOWebDOC 」-「 IODOC 」を選択します。



2. 起動した帳票デザインツール IODOC のメニューバーから「ファイル」-「開く」を選択し、< somu_tatekae.dlf >を選択します。

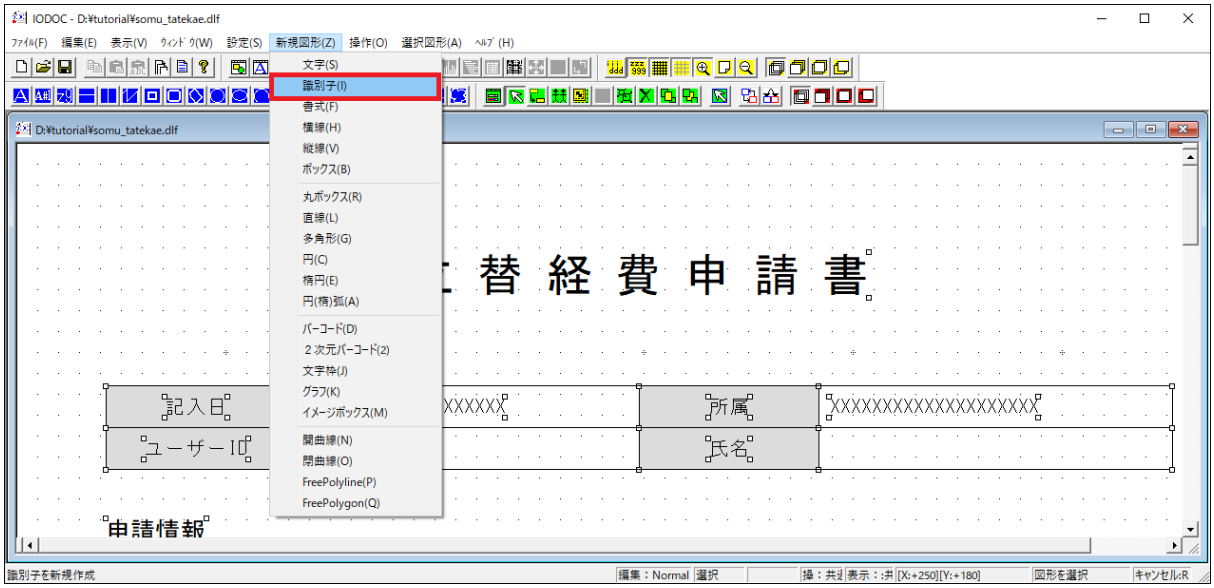


選択したファイルが開きます。

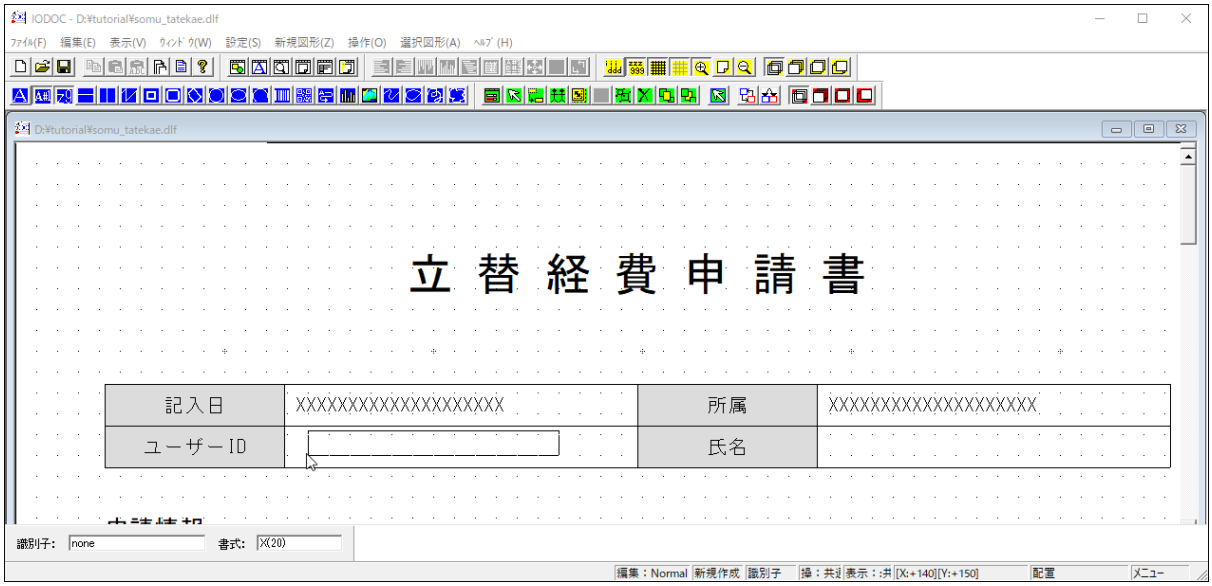


識別子を追加する

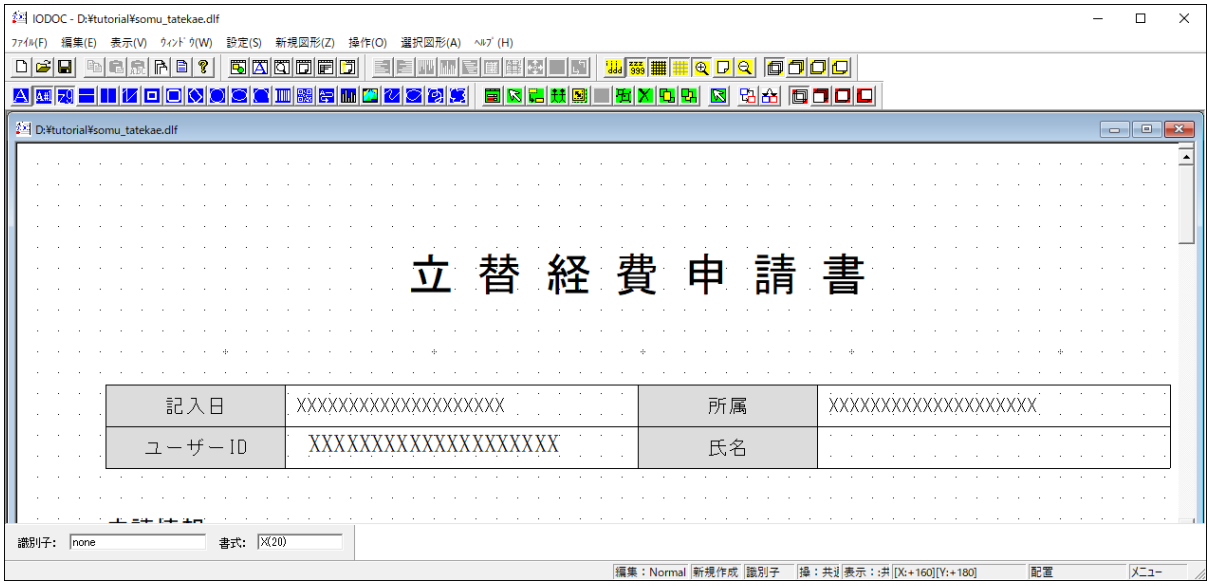
1. メニューバーから「新規図形」-「識別子」を選択します。



2. マウスカーソルに表示される四角いボックスを、レイアウト内の項目「ユーザーID」の横に合わせ、クリックします。

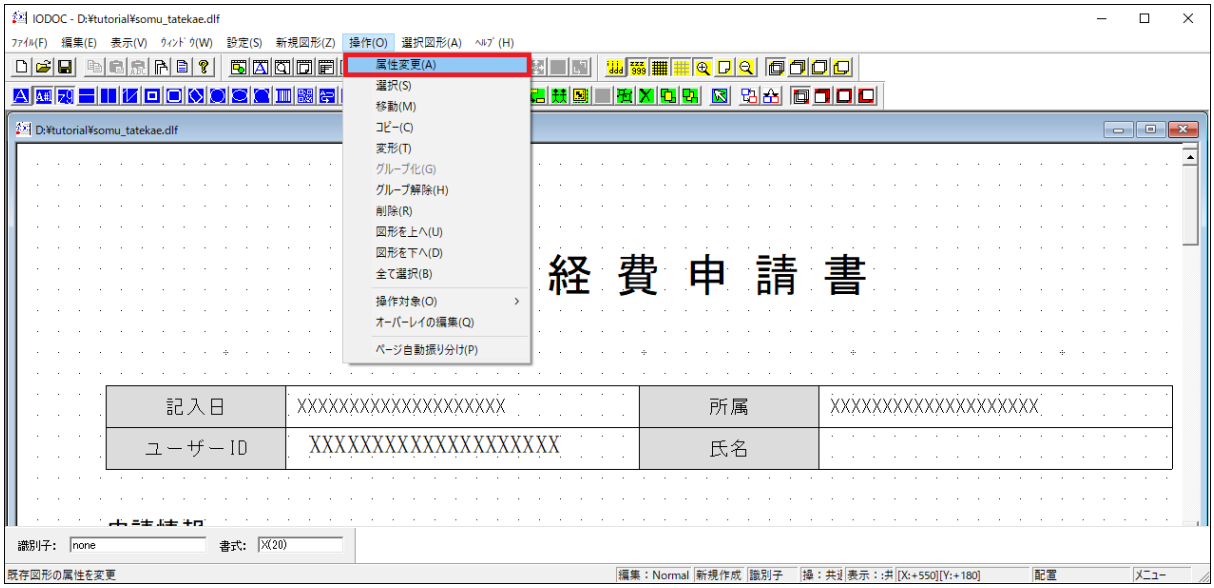


クリックした箇所に識別子が追加されます。

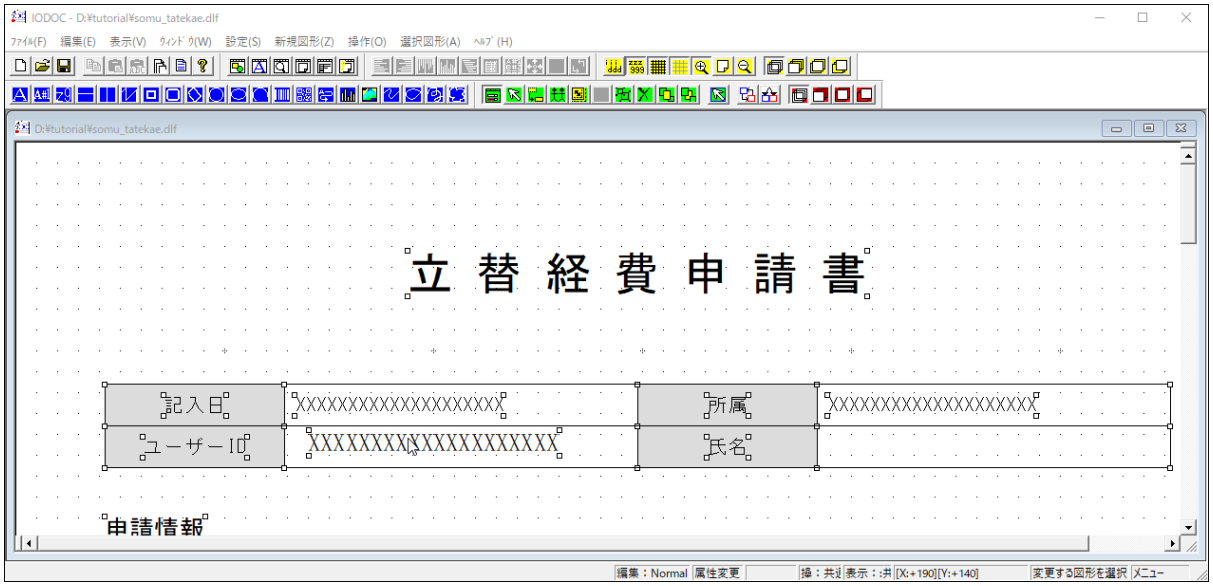


識別子を編集する

1. メニューバーから「操作」-「属性変更」を選択します。

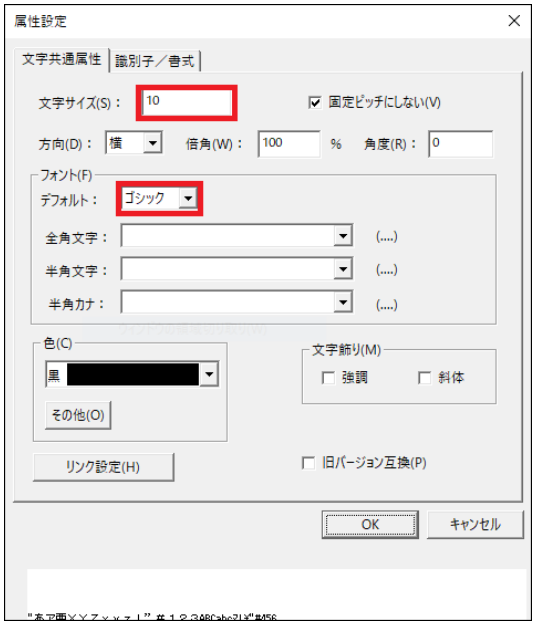


2. 追加した識別子をクリックします。



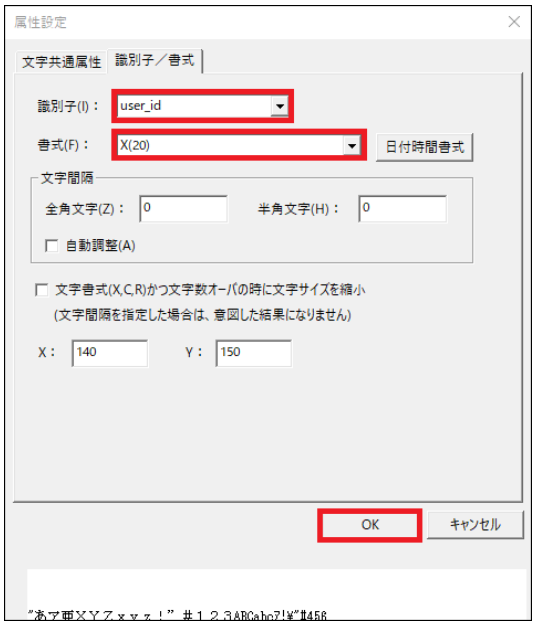
3. 「属性設定」画面の「文字共通属性」タブを開き、次の値を設定します。

項目	値
文字サイズ	10
フォント - デフォルト	ゴシック

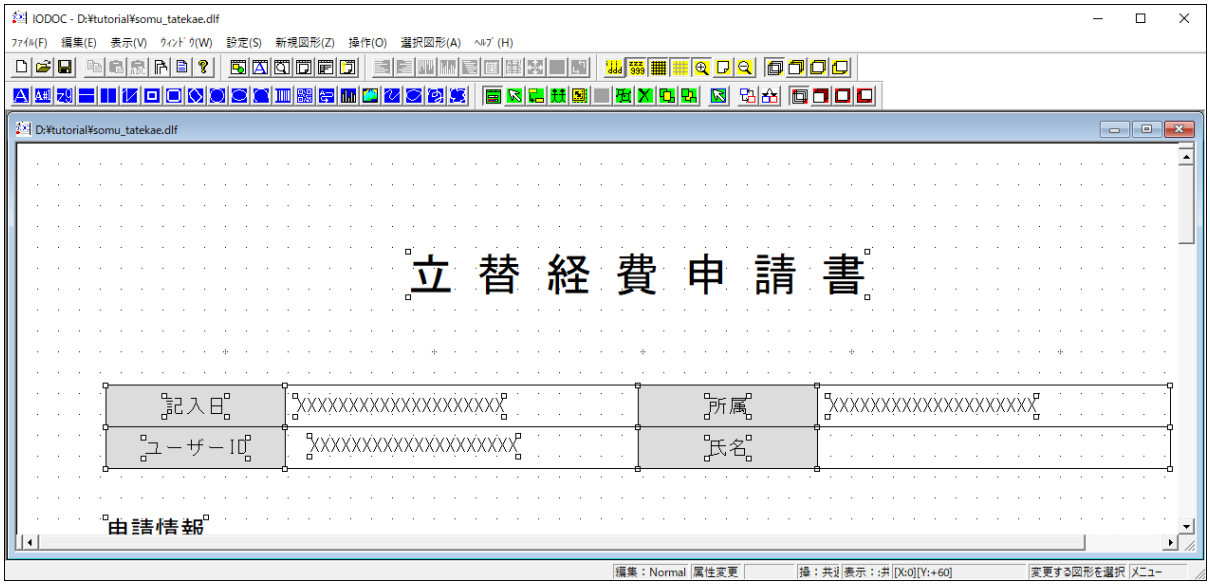


4. 「識別子／書式」タブを開き、次の値を設定し、「OK」をクリックします。

項目	値
識別子	user_id
書式	X(20)

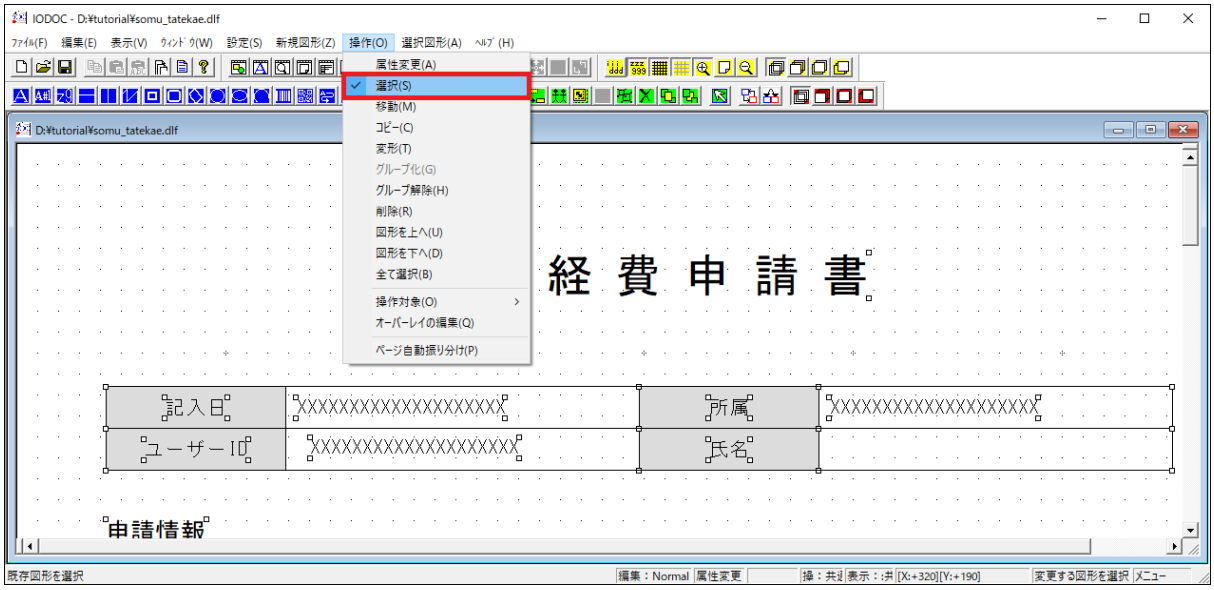


文字サイズ、および、フォントの設定が画面上に反映されます。

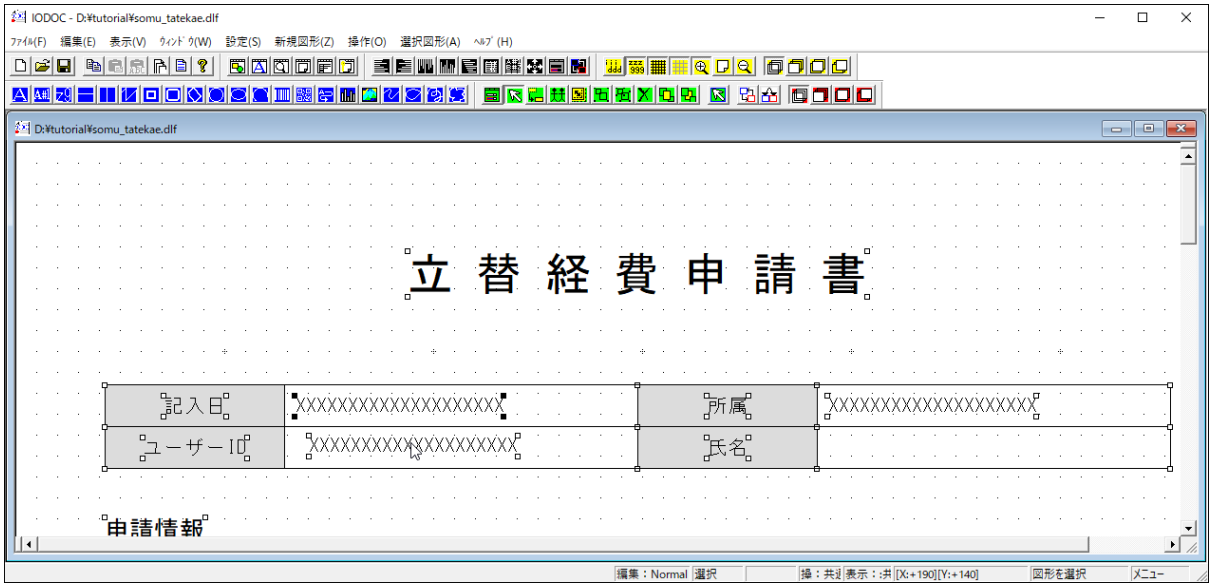


識別子の位置を調整する

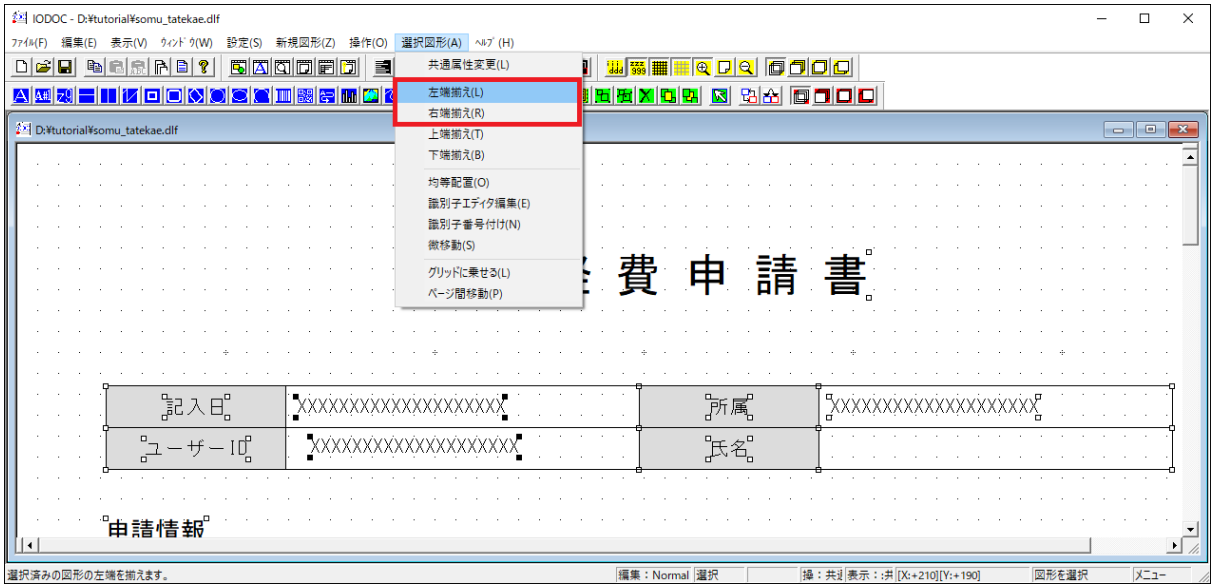
1. メニューバーから「操作」-「選択」を選択します。



2. レイアウト内の項目「記入日」の識別子と、「ユーザーID」の識別子をクリックします。

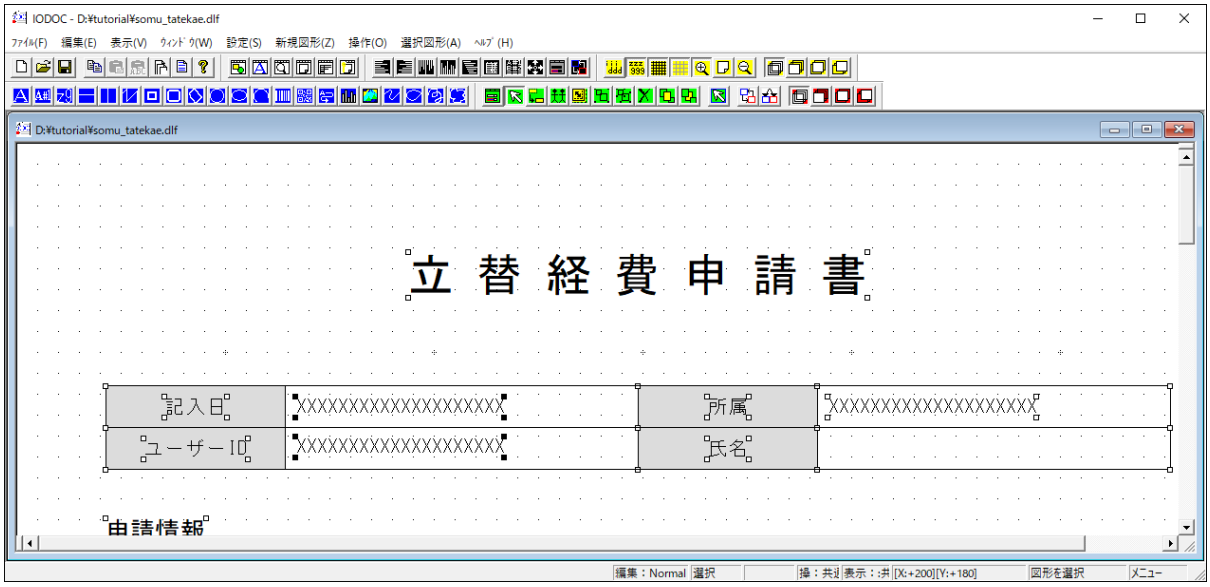


3. 「記入日」の識別子の位置に合わせるように、メニューバーから「選択図形」-「左端揃え」、または、「右端揃え」を選択します。

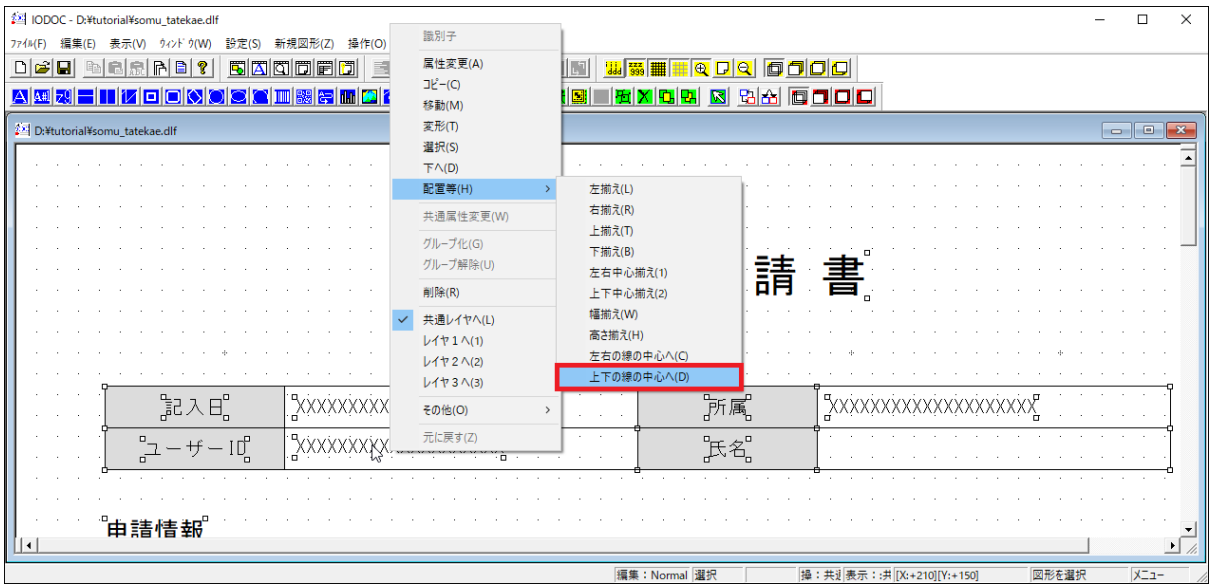


> **コラム**
> 「ユーザーID」の識別子が、「記入日」の識別子より左側に配置されている場合は「右端揃え」、右側に配置されている場合は「左端揃え」を選択します。

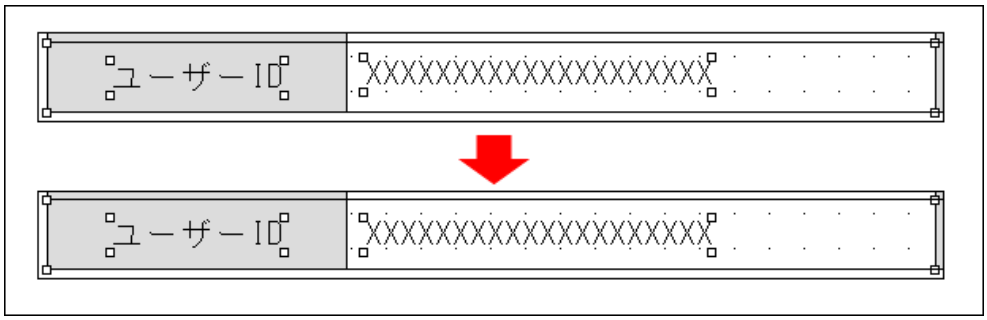
「ユーザーID」の識別子が、「記入日」の識別子の位置と同じ位置に移動します。



4. 「ユーザーID」の識別子を右クリックし、右クリックメニューから「配置等」-「上下の線の中心へ」を選択します。



「ユーザーID」の識別子が、上下の線の中心に移動します。



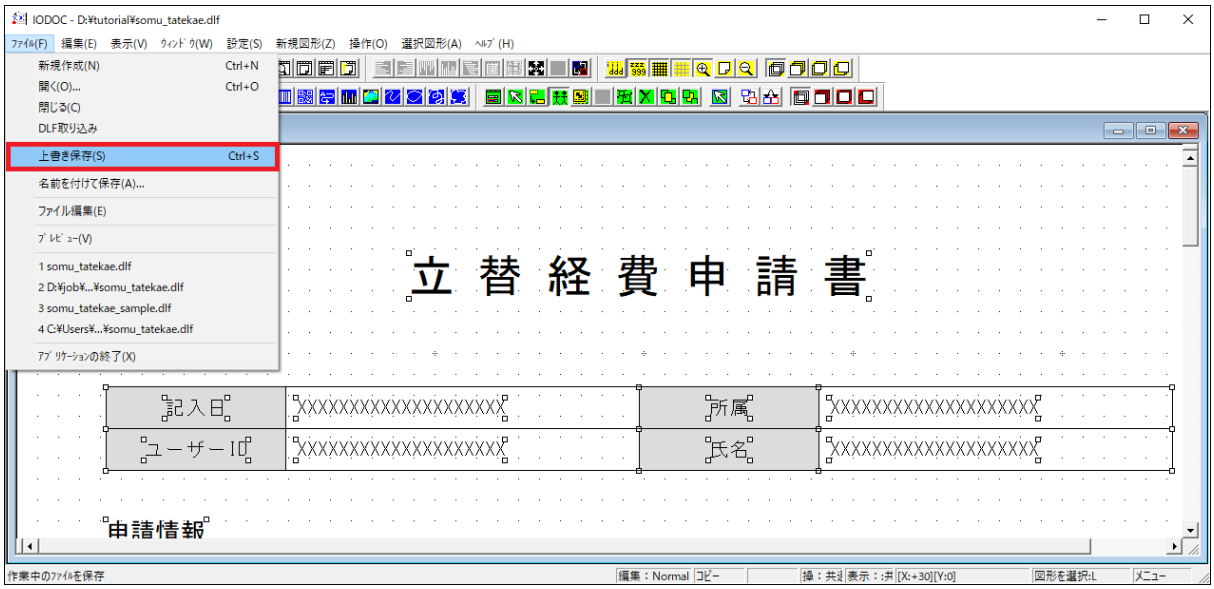
残りの識別子を全て設定する

上記の手順、および、< layout/somu_tatekae_sample.dlf >を参考に、次の識別子を全て設定します。

項目	識別子名	書式
氏名	user_name	X(20)
支払金額	payment	\\\,\\9
支払先	payee	X(20)

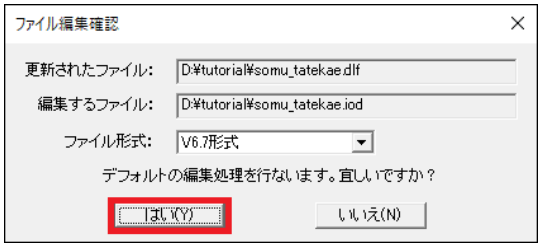
名前を付けて保存する

1. メニューバーから「ファイル」-「上書き保存」を選択します。



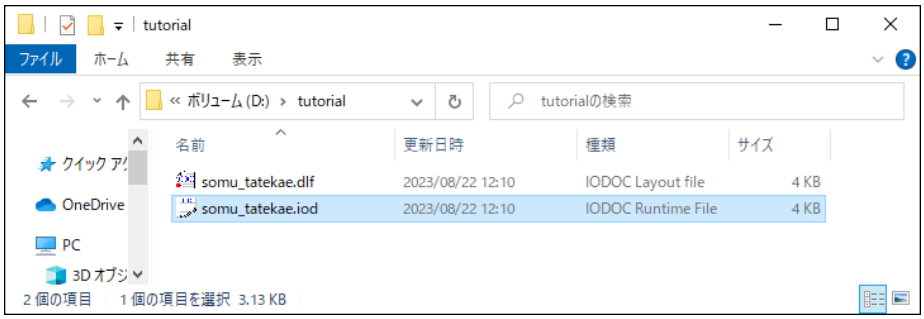
> **コラム**
> Ctrl + S でも保存可能です。

2. 「ファイル編集確認」画面の「ファイル形式」を必要に応じて変更し、「はい」をクリックします。



> **注意**
> Linux 環境の場合は「V4.9形式」に変更してください。
>
> 「V.4.9.3形式」で保存した場合、処理実行時に「-100 ファイルアクセスエラー」になります。

< somu_tatekae.dlf >は上書き保存され、同じ階層に< somu_tatekae.iod >が出力されます。



作成した帳票レイアウトファイルを設置する

1. < %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >ディレクトリを作成します。
2. < somu_tatekae.iod >を< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >配下に設置します。

帳票レイアウトファイルに渡すデータの準備

帳票レイアウトファイルに埋め込むデータを準備する作業です。

埋め込むデータの指定方法については、「 *埋め込むデータの指定方法について* 」を参照してください。

本項目では、帳票デザインツール IODOC の、データファイルの自動作成機能を使用し、DATファイルを作成します。

手順

- 帳票レイアウト作成用ファイルを開く
- DATファイルを作成する
- 作成したデータファイルを設置する

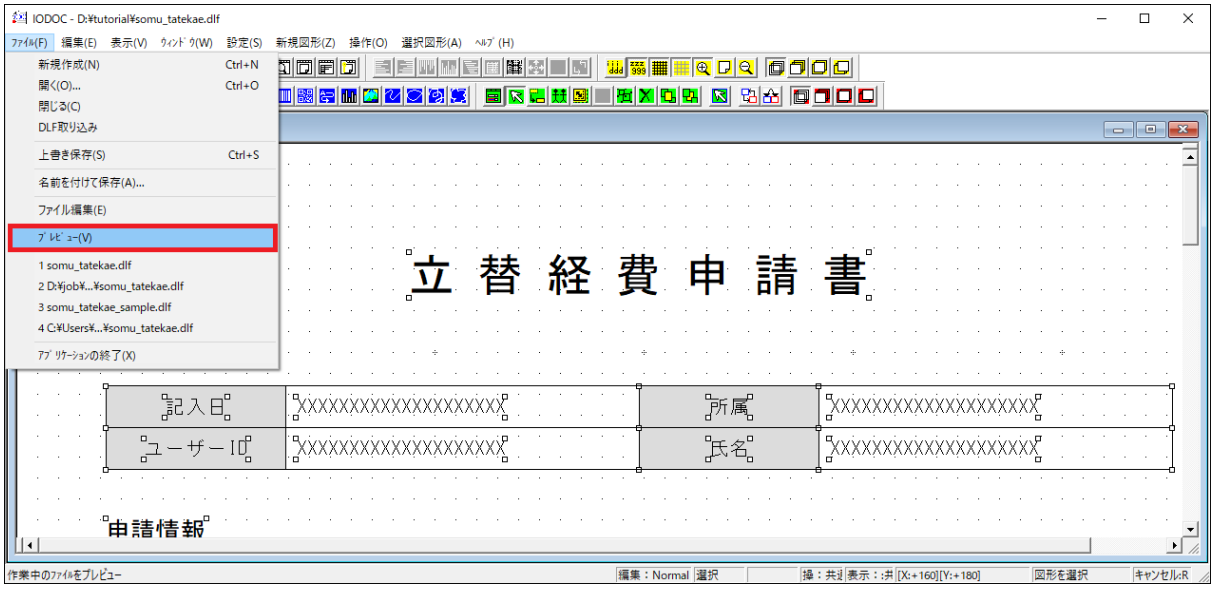
帳票レイアウト作成用ファイルを開く

1. 帳票デザインツール IOWebDOC をインストールしているPCに、前章で作成した< somu_tatekae.dlf >をコピーします。
2. コピーした< somu_tatekae.dlf >を開きます。

> **コラム**
> ファイルを開く詳細な手順については、「 *帳票レイアウト作成用ファイルを開く* 」を参照してください。

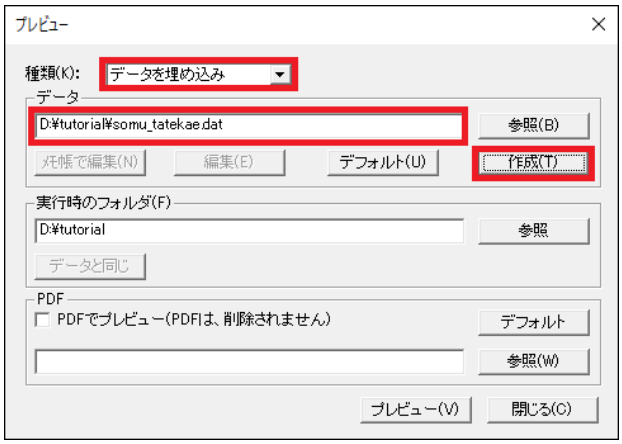
DATファイルを作成する

1. メニューバーから「ファイル」-「プレビュー」を選択します。



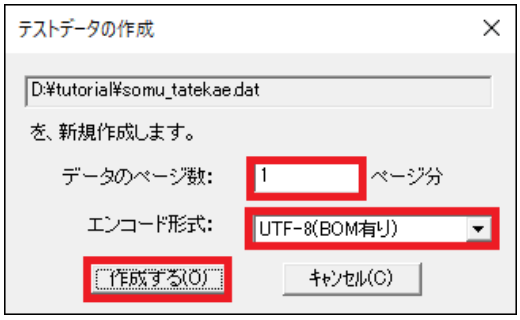
2. 「プレビュー」画面で次の値を設定し、「作成」をクリックします。

項目	値
種類	データを埋め込み
データ	{データファイルの出力先}/somu_tatekae.dat

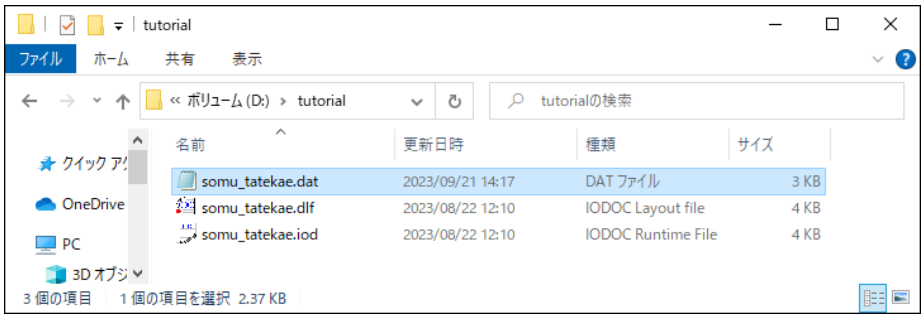


3. 「テストデータの作成」画面で次の値を設定し、「作成する」をクリックします。

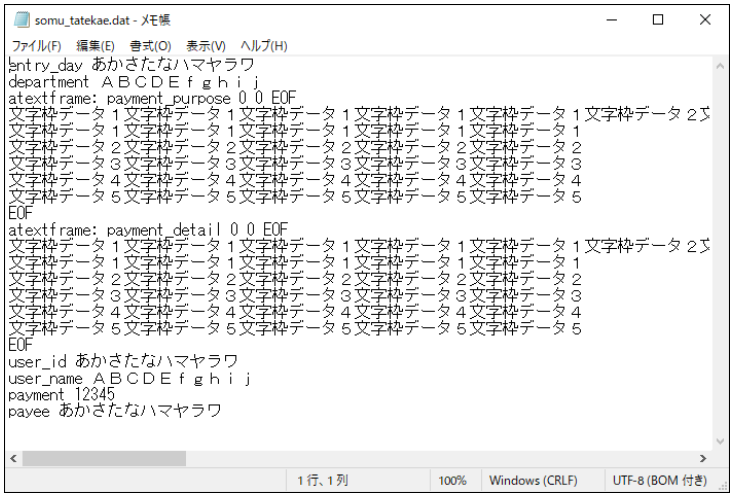
項目	値
データのページ数	1
エンコード形式	UTF-8 (BOM有り)



指定した出力先に< somu_tatekae.dat >が出力されます。



4. < somu_tatekae.dat >をテキストエディタで開きます。



> **コラム**
>
> 帳票レイアウトファイル上の識別子の記載順と、出力したデータファイル内の識別子の記載順が異なる場合があります。
>
> 記載順が異なっていても、識別子名が一致していれば問題はありません。
>
> また、本チュートリアルで追加した識別子は、データファイルの末尾に記載されている可能性があります。

5. < layout/somu_tatekae_sample.dat >を参考に、識別子に設定する値を、適切な値に修正し、上書き保存します。

> **注意**
> 文字コードを UTF-8（BOMあり） にして保存してください。

作成したデータファイルを設置する

< somu_tatekae.dat >を< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >配下に設置します。

PDFファイルの作成

IM-PDFDesigner for Accel Platform を使用し、帳票レイアウトファイルと埋め込むデータを組み合わせて、PDFファイルを作成する作業です。

スクリプト開発モデル 、または、 JavaEE開発モデル を使用し、プログラムを作成します。

スクリプト開発モデル

本項目では、 スクリプト開発モデル 用APIを使用したプログラムを作成し、そのプログラムを実行することで、PDFファイルを作成します。

手順

- プログラムを作成する
  - 実行画面用のHTMLファイルを設置する
  - PDFファイル出力処理用のJSファイルを作成する
- 設定ファイルを設置する
- プログラムを登録する
  - 認可を設定する
  - メニューを設定する
- プログラムを実行・確認する

プログラムを作成する

実行画面用のHTMLファイルを設置する

1. < %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/jssp/src/pdfd/tutorial/basic >ディレクトリを作成します。
2. < jssp/iodoc.html >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/jssp/src/pdfd/tutorial/basic >配下に設置します。

PDFファイル出力処理用のJSファイルを作成する

1. < jssp/iodoc.js >をテキストエディタで開きます。
2. 3行目を次のように修正し、帳票レイアウトファイルを指定します。

```
const iodPath = "/pdfd/tutorial/basic/somu_tatekae.iod";
```

3. 4行目を次のように修正し、DATファイルを指定します。

```
const datPath = "/pdfd/tutorial/basic/somu_tatekae.dat";
```

4. 10行目を次のように修正し、PDFファイルの出力先フォルダを指定します。

```
const dirPath = "/pdfd/tutorial/basic/";
```

5. 11行目を次のように修正し、PDFファイル名の接頭文字を指定します。

```
const prefix = "somu_tatekae";
```

6. 19行目を次のように修正し、クラスを指定します。
   - スタンドアローン構成 の場合

```
const pdf = new IODoc(iodPath, "");
```

   - 分散構成 の場合

```
const pdf = new IODocRemote(iodPath, "");
```

7. 上書き保存します。

> **注意**
> 文字コードを UTF-8 にして保存してください。

8. < jssp/iodoc.js >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/jssp/src/pdfd/tutorial/basic >配下に設置します。

設定ファイルを設置する

< jssp/pdfd-tutorial_basic_iodoc.xml >を< %RESIN_HOME%/webapps/{アプリケーション名}/WEB-INF/conf/routing-jssp-config >配下に設置します。

プログラムを登録する

設置したプログラム、および、設定ファイルを環境に適用するため、 Web Application Server を再起動します。

再起動後、プログラムを認可とメニューに設定します。

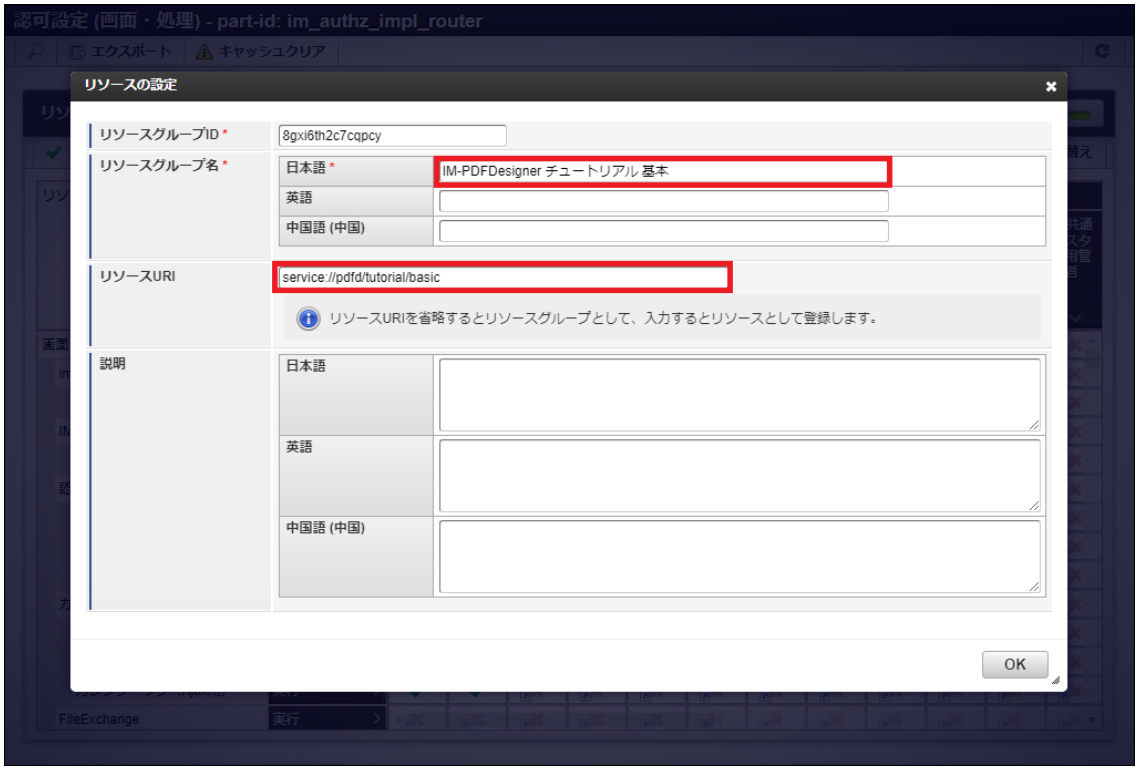
> **コラム**
>
> テナント管理に関する詳細な手順については、次を参照してください。
>
> - 「テナント管理者操作ガイド」-「認可を設定する」
> - 「テナント管理者操作ガイド」-「メニューを設定する」

> **注意**
>
> 同一の認可やメニューグループ、メニューフォルダを既に設定している場合、該当する手順は不要です。

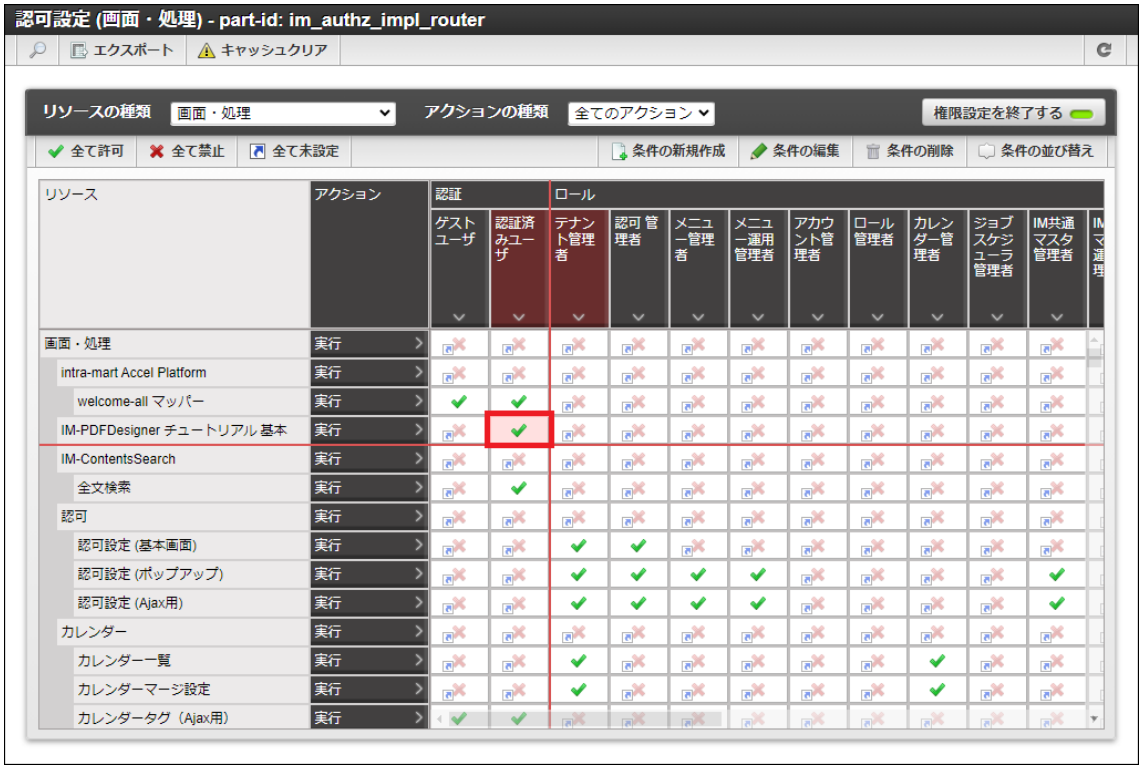
認可を設定する

1. テナント管理の権限を持つユーザで intra-mart Accel Platform にログインします。
2. リソース「画面・処理」配下に、次のリソースを作成します。

項目	値
リソースグループ名 - 日本語	IM-PDFDesigner チュートリアル 基本
リソースURI	*service://pdfd/tutorial/basic*



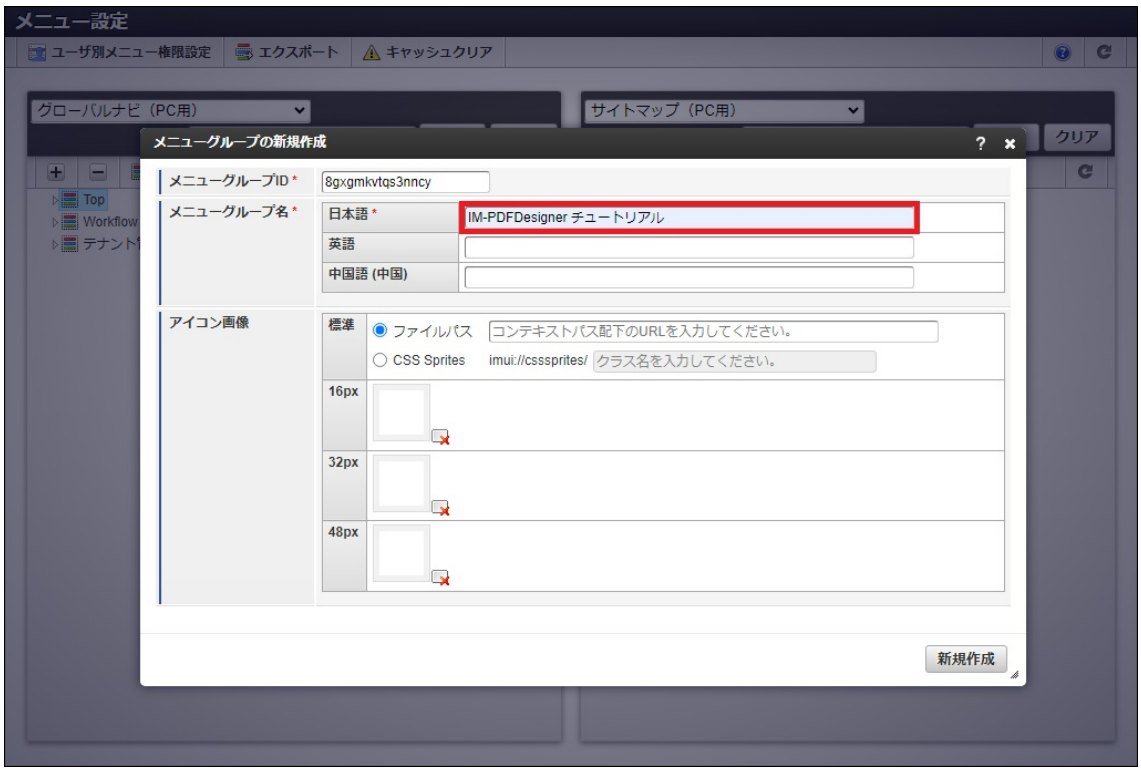
3. リソース「 IM-PDFDesigner チュートリアル 基本 」の「認証済みユーザ」に「全て許可」を付与します。



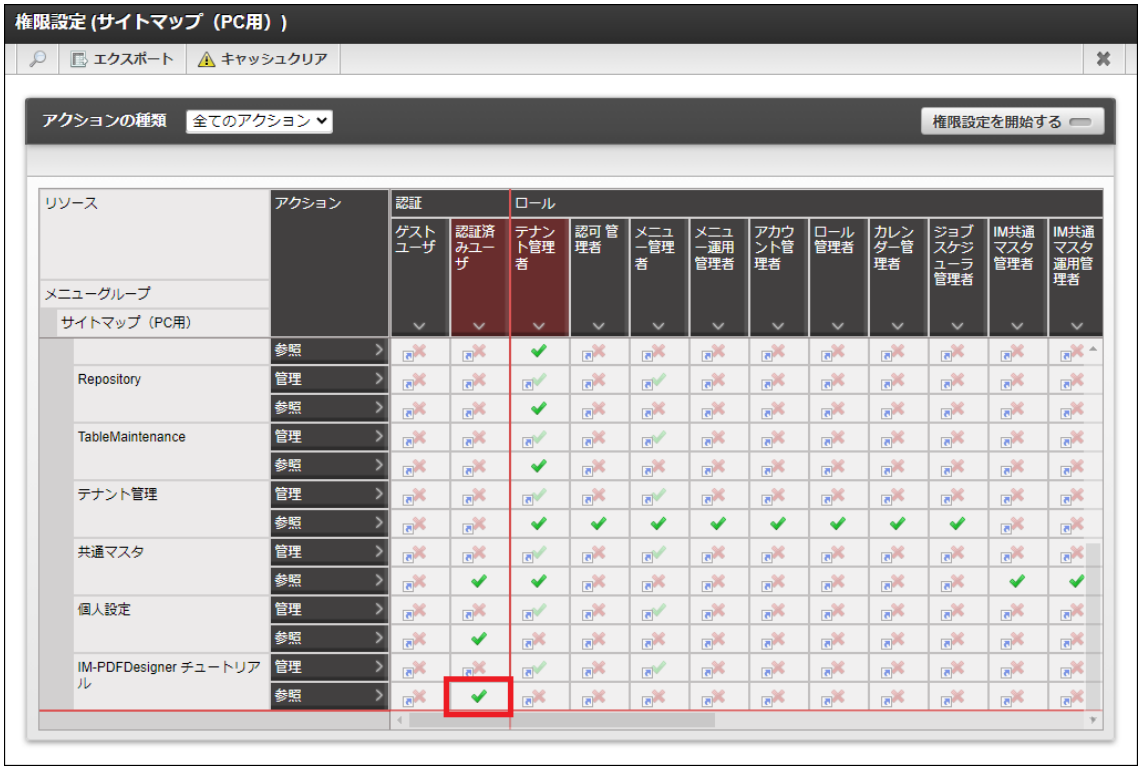
メニューを設定する

1. テナント管理の権限を持つユーザで intra-mart Accel Platform にログインします。
2. 「サイトマップ（PC用）」のトップ階層に、次のメニューグループを作成します。

項目	値
メニューグループ名 - 日本語	IM-PDFDesigner チュートリアル

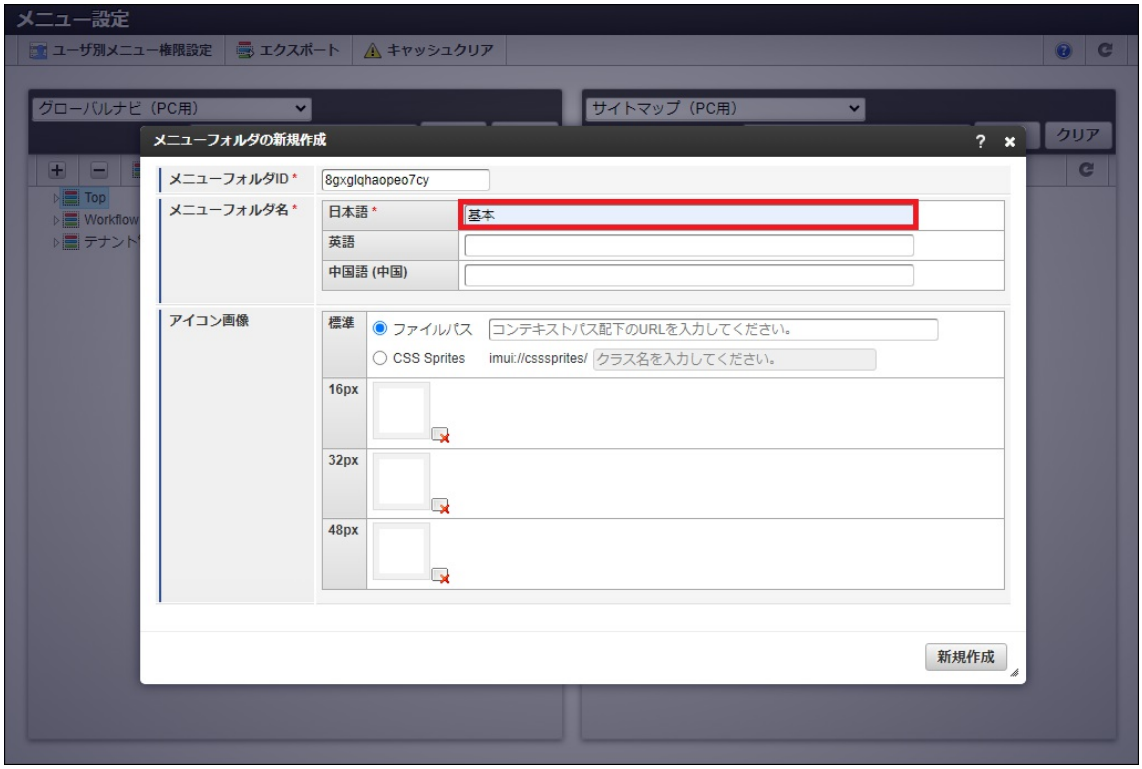


3. メニューグループ「 IM-PDFDesigner チュートリアル 」-「参照」の、「認証済みユーザ」に「全て許可」を付与します。



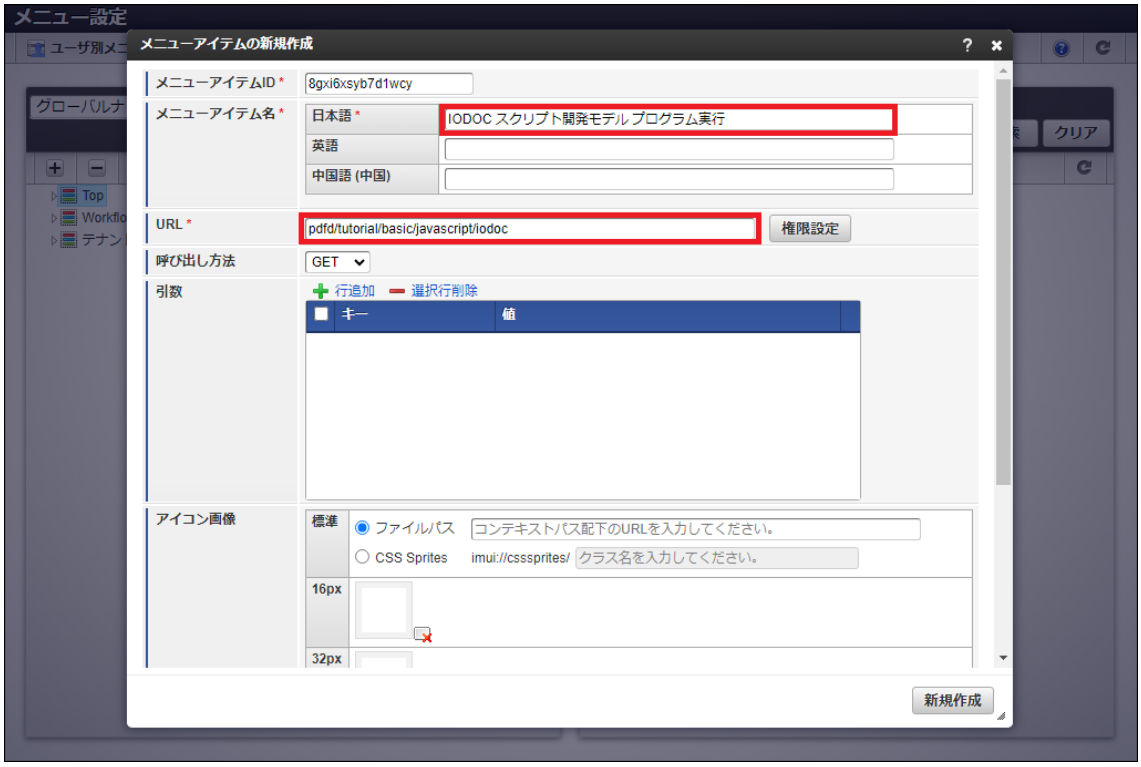
4. メニューグループ「 IM-PDFDesigner チュートリアル 」配下に、次のメニューフォルダを作成します。

項目	値
メニューフォルダ名 - 日本語	基本



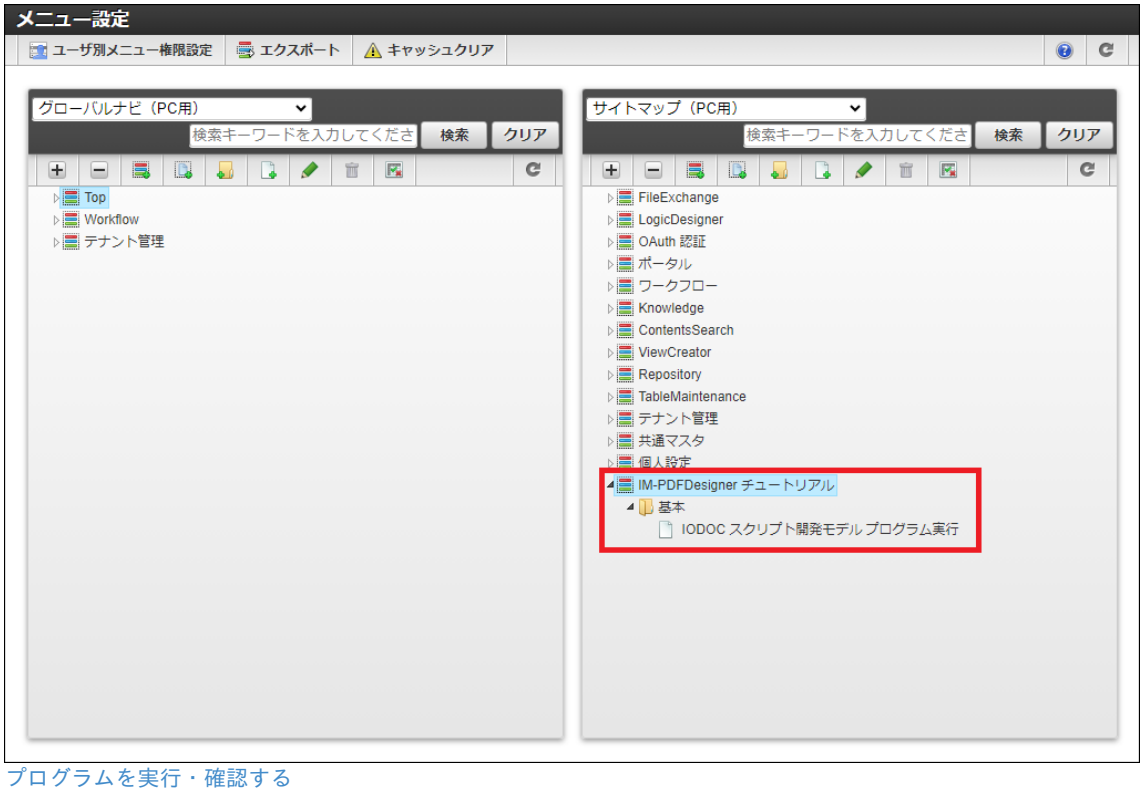
5. メニューフォルダ「基本」配下に、次のメニューアイテムを作成します。

項目	値
メニューアイテム名 - 日本語	IODOC スクリプト開発モデル プログラム実行
URL	pdfd/tutorial/basic/javascript/iodoc



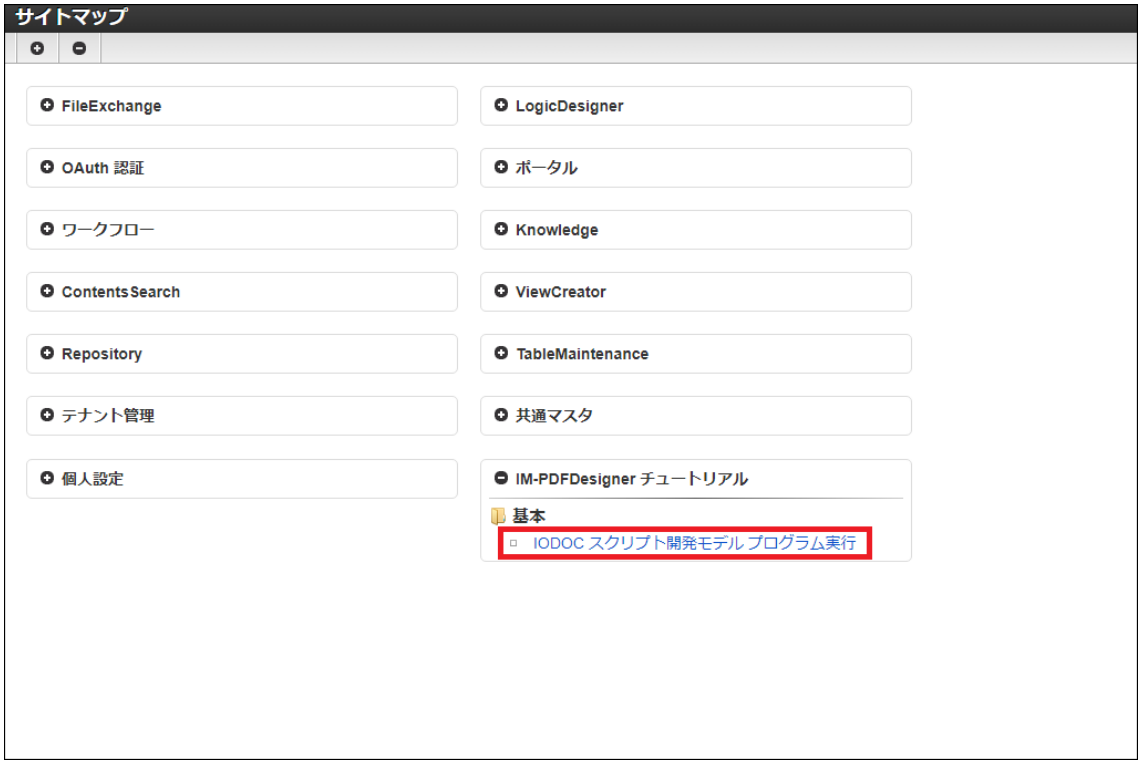
メニューの設定ができました。メニュー構成は、次の通りです。

- メニューグループ： IM-PDFDesigner チュートリアル
- メニューフォルダ： 基本
- メニューアイテム： IODOC スクリプト開発モデル プログラム実行



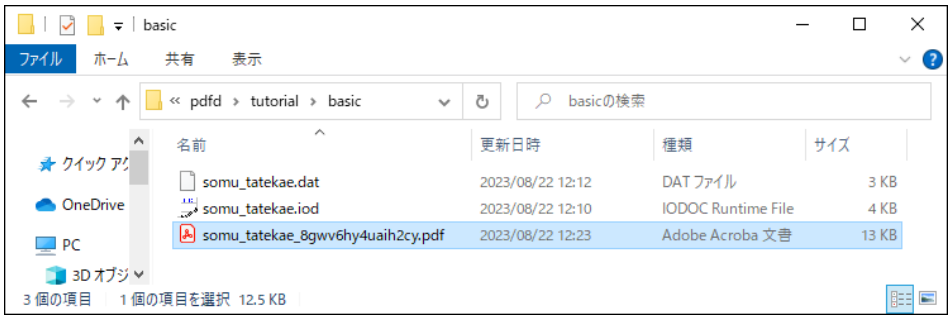
プログラムを実行・確認する

1. 一般ユーザで intra-mart Accel Platform にログインします。
2. 「サイトマップ」-「 IM-PDFDesigner チュートリアル 」-「基本」-「 IODOC スクリプト開発モデル プログラム実行 」をクリックします。



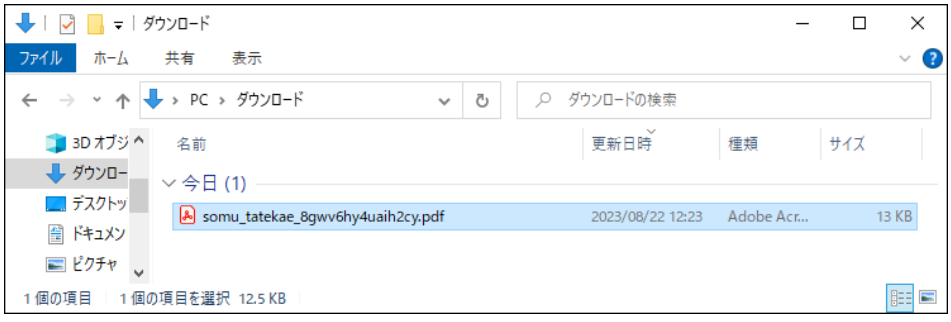
3. 表示された実行画面の「PDF作成」をクリックします。

プログラムが実行され、< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >にPDFファイルが出力されます。



> **コラム**
> エラーが発生した場合は、エラーメッセージの内容に従いファイルの修正等を行ってください。

また、処理終了後にPDFファイルがダウンロードされます。



4. 出力されたPDFファイルをPDFビューア（ Adobe Acrobat Reader など）で開き、正しく表示されることを確認します。

JavaEE開発モデル

本項目では、 JavaEE開発モデル 用APIを使用したプログラムを作成し、そのプログラムを実行することで、PDFファイルを作成します。

手順

- プログラムを作成する
  - 実行画面用のJSPファイルを設置する
  - PDFファイル出力処理用のJSPファイルを作成する
  - ダウンロード処理用のJSPファイルを設置する
- 設定ファイルを設置する
- プログラムを登録する
  - 認可を設定する
  - メニューを設定する
- プログラムを実行・確認する

プログラムを作成する

実行画面用のJSPファイルを設置する

1. < %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/basic >ディレクトリを作成します。

2. < javaee/iodoc.jsp >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/basic >配下に設置します。

PDFファイル出力処理用のJSPファイルを作成する

1. < javaee/iodoc_act.jsp >をテキストエディタで開きます。

2. 4行目を次のように修正し、クラスを指定します。
   - スタンドアローン構成 の場合

```
<%@ page import="jp.co.intra_mart.product.pdfmaker.net.CSVDoc" %>
```

   - 分散構成 の場合

```
<%@ page import="jp.co.intra_mart.product.pdfmaker.net.rest.CSVDocRemote" %>
```

3. 8行目を次のように修正し、帳票レイアウトファイルを指定します。

```
String iodPath = "pdfd/tutorial/basic/somu_tatekae.iod";
```

4. 9行目を次のように修正し、DATファイルを指定します。

```
String datPath = "pdfd/tutorial/basic/somu_tatekae.dat";
```

5. 17行目を次のように修正し、PDFファイルの出力先フォルダを指定します。

```
String dirPath = "pdfd/tutorial/basic/";
```

6. 18行目を次のように修正し、PDFファイル名の接頭文字を指定します。

```
String prefix = "somu_tatekae";
```

7. 26行目を次のように修正し、クラスを指定します。
   - スタンドアローン構成 の場合

```
CSVDoc pdf = new CSVDoc(iodPath, "");
```

   - 分散構成 の場合

```
CSVDocRemote pdf = new CSVDocRemote(iodPath, "");
```

8. 上書き保存します。

> **注意**
> 文字コードを UTF-8 にして保存してください。

9. < javaee/iodoc_act.jsp >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/basic >配下に設置します。

ダウンロード処理用のJSPファイルを設置する

< javaee/download.jsp >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/basic >配下に設置します。

> **注意**
> 同一のファイルを既に設置している場合、該当する手順は不要です。

設定ファイルを設置する

1. < javaee/service-config-pdfd_tutorial_basic_iodoc.xml >を< %RESIN_HOME%/webapps/{アプリケーション名}/WEB-INF/classes >配下に設置します。
2. < javaee/pdfd-tutorial_basic_iodoc.xml >を< %RESIN_HOME%/webapps/{アプリケーション名}/WEB-INF/conf/routing-service-config >配下に設置します。

プログラムを登録する

設置したプログラム、および、設定ファイルを環境に適用するため、 Web Application Server を再起動します。

再起動後、プログラムを認可とメニューに設定します。

> **コラム**
> テナント管理に関する詳細な手順については、次を参照してください。
>
> - 「テナント管理者操作ガイド」-「認可を設定する」
> - 「テナント管理者操作ガイド」-「メニューを設定する」

> **注意**
> 同一の認可やメニューグループ、メニューフォルダを既に設定している場合、該当する手順は不要です。

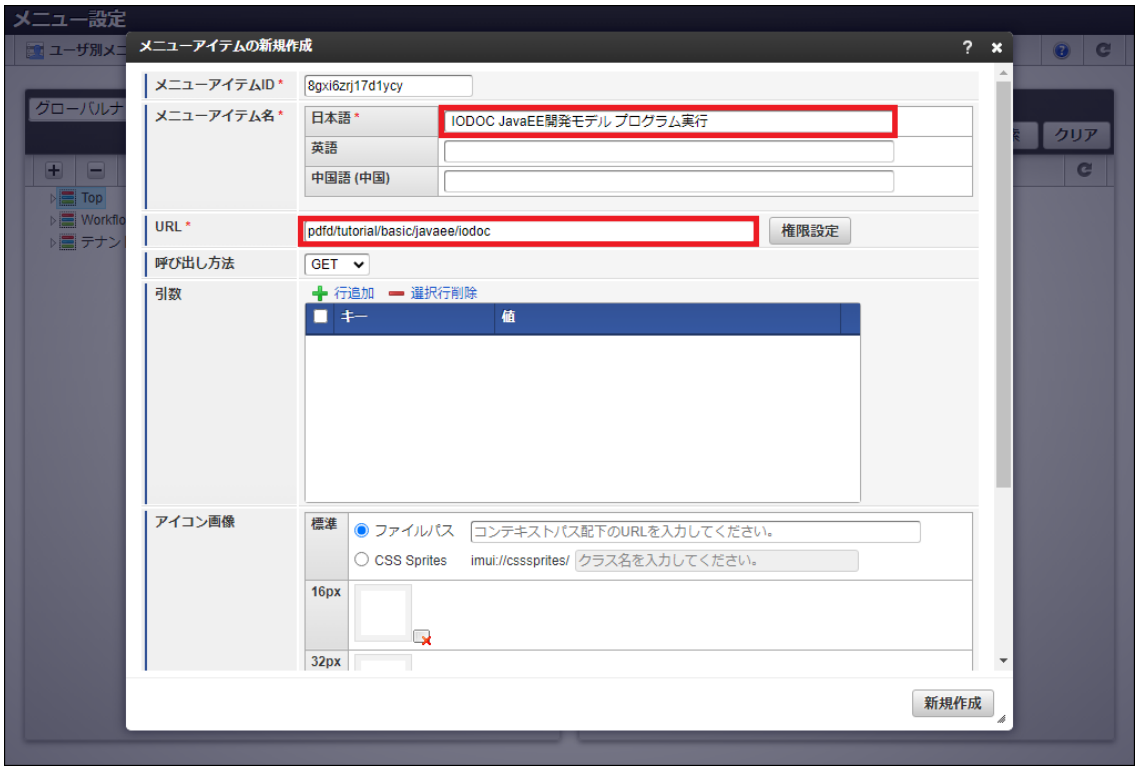
認可を設定する

「*認可を設定する*」を参照し、認可を設定してください。

メニューを設定する

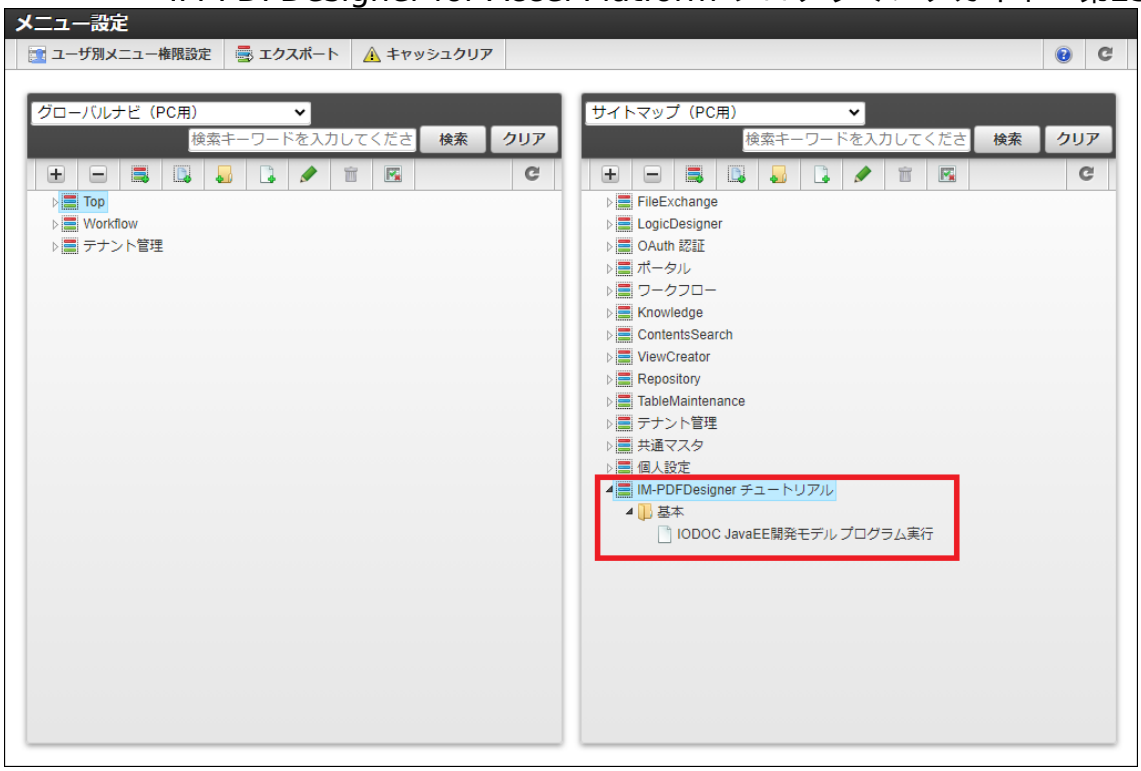
1. テナント管理の権限を持つユーザで intra-mart Accel Platform にログインします。
2. 「*メニューを設定する*」を参照し、メニューグループ、および、メニューフォルダを設定します。
3. メニューフォルダ「基本」配下に、次のメニューアイテムを作成します。

項目	値
メニューアイテム名 - 日本語	IODOC JavaEE開発モデル プログラム実行
URL	pdfd/tutorial/basic/javaee/iodoc



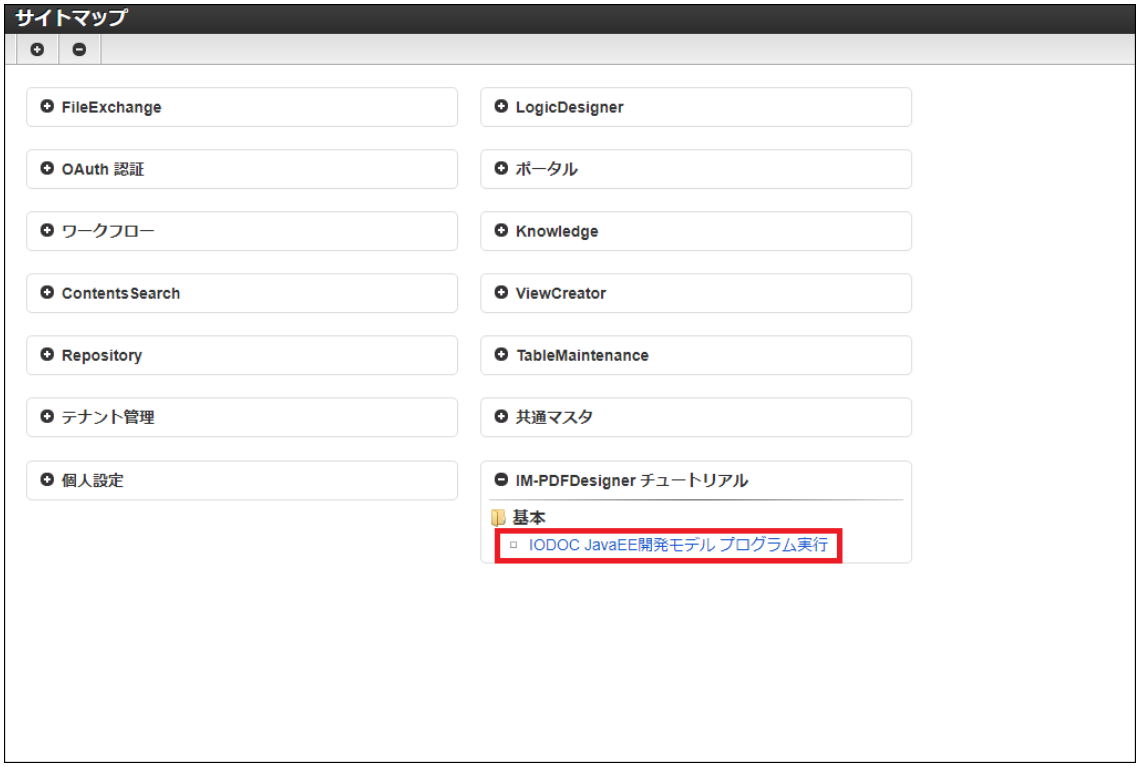
メニューの設定ができました。メニュー構成は、次の通りです。

- メニューグループ： IM-PDFDesigner チュートリアル
- メニューフォルダ： 基本
- メニューアイテム： IODOC JavaEE開発モデル プログラム実行



プログラムを実行・確認する

1. 一般ユーザで intra-mart Accel Platform にログインします。
2. 「サイトマップ」-「 IM-PDFDesigner チュートリアル 」-「基本」-「 IODOC JavaEE開発モデル プログラム実行 」をクリックします。



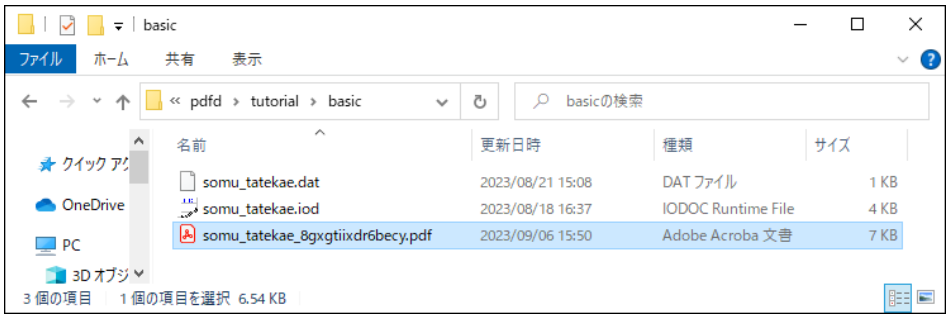
3. 表示された実行画面の「PDF作成」をクリックします。

プログラムが実行され、実行結果の画面に遷移します。

コラム

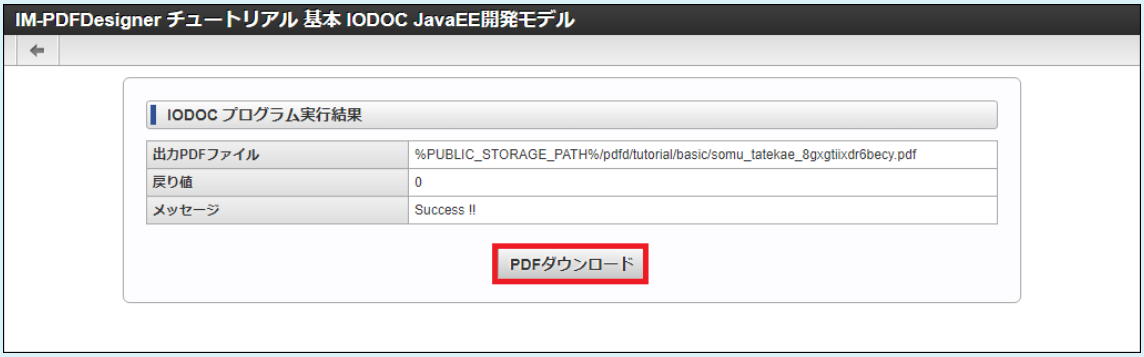
エラーが発生した場合は、エラーメッセージの内容に従いファイルの修正等を行ってください。

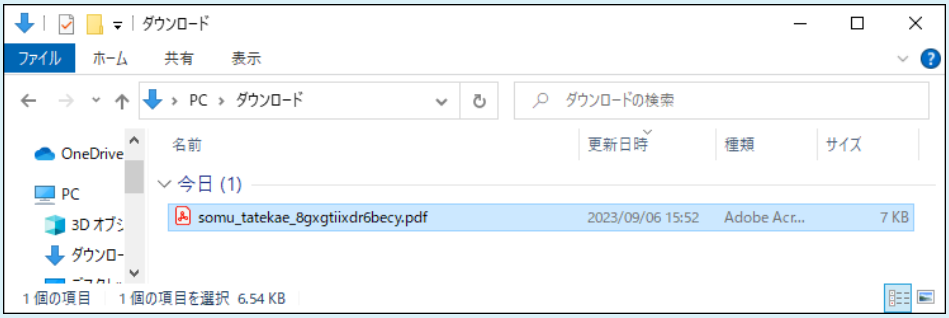
また、< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >にPDFファイルが出力されます。



コラム

実行結果画面内の「PDFダウンロード」をクリックすると、PDFファイルがダウンロードされます。





4. 出力されたPDFファイルをPDFビューア（ Adobe Acrobat Reader など）で開き、正しく表示されることを確認します。

連票形式 IODBDOC

本チュートリアルでは連票形式のPDFファイルを作成します。

チュートリアルを実施するにあたり、次のzipファイルをダウンロードし、解凍してください。

< **tutorial_basic_iodbdoc.zip** >

解凍したファイルの構成は、次の通りです。

フォルダ名／ファイル名	説明
layout/	帳票レイアウト関連フォルダ

フォルダ名／ファイル名	説明
somu_kotsuhi.ddl	帳票レイアウトファイル
somu_kotsuhi_sample.ddl	帳票レイアウトファイル（参考用）
somu_kotsuhi_sample.csv somu_kotsuhi_sample.dat	帳票レイアウトファイルに埋め込むデータファイル（参考用）
jssp/	スクリプト開発モデル 用フォルダ
iodbdoc.html iodbdoc.js	スクリプト開発モデル 用プログラム
pdfd-tutorial_basic_iodbdoc.xml	スクリプト開発モデル 用設定ファイル
javaee/	JavaEE開発モデル 用フォルダ
iodbdoc.jsp iodbdoc_act.jsp download.jsp	JavaEE開発モデル 用プログラム
service-config-pdfd_tutorial_basic_iodbdoc.xml pdfd-tutorial_basic_iodbdoc.xml	JavaEE開発モデル 用設定ファイル

次の手順に沿って、チュートリアルを進めます。

帳票レイアウトファイルの作成

帳票デザインツール IODBDOC を使用し、PDFファイルの雛形を作成する作業です。

本項目では、未完成の帳票レイアウトファイルに外部変数、および、データ項目変数を設定後、帳票レイアウトファイルを作成します。

外部変数、および、データ項目変数とは、帳票レイアウトファイル上に、文字列や画像などのデータを埋め込むための項目です。

外部変数は、該当ページのタイトルやヘッダ部分に一度だけ出力する場合に使用します。

データ項目変数は、データを明細行として出力する場合に使用します。

PDFファイル作成時、外部変数、および、データ項目変数とデータを紐づけることで、帳票レイアウトファイル上に設定した各変数にデータが埋め込まれます。

帳票レイアウトファイル< layout/somu_kotsuhi.ddl >は未完成です。

本チュートリアルで、次の項目を設定していきます。

- 外部変数
  - ユーザーID
  - 氏名
- データ項目変数
  - 到着
  - 片／往
  - 金額

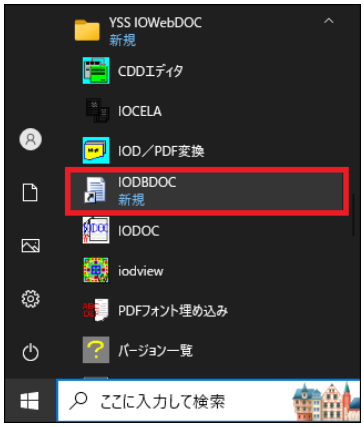
コラム
帳票デザインツール（連票形式 IODBDOC ）の詳しい使用方法については、「 *連票形式 IODBDOC* 」を参照してください。

手順

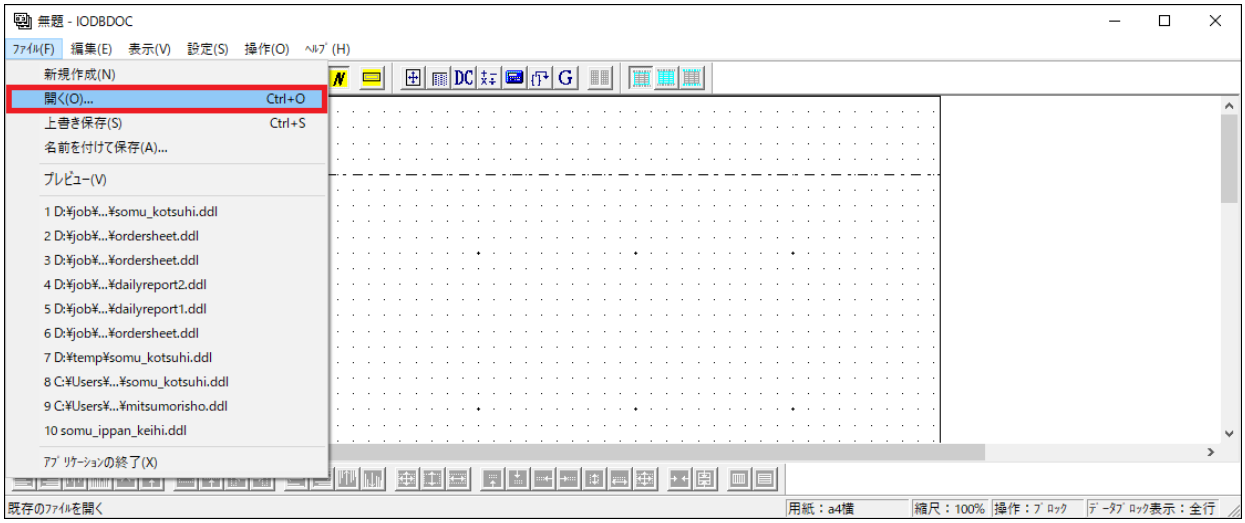
- 帳票レイアウトファイルを開く
- 外部変数を設定する
  - 外部変数を追加する
  - 外部変数を編集する
  - 外部変数の位置を調整する
  - 残りの外部変数を全て設定する
- データ項目変数を設定する
  - データ項目変数を追加する
  - データ項目変数を編集する
  - データ項目変数の位置を調整する
  - 残りのデータ項目変数を全て設定する
- 名前を付けて保存する
- 作成した帳票レイアウトファイルを設置する

帳票レイアウトファイルを開く

1. 帳票デザインツール IOWebDOC をインストールしているPCに< layout/somu_kotsuhi.ddl >をコピーします。
2. 「スタート」-「 YSS IOWebDOC 」-「 IODBDOC 」を選択します。



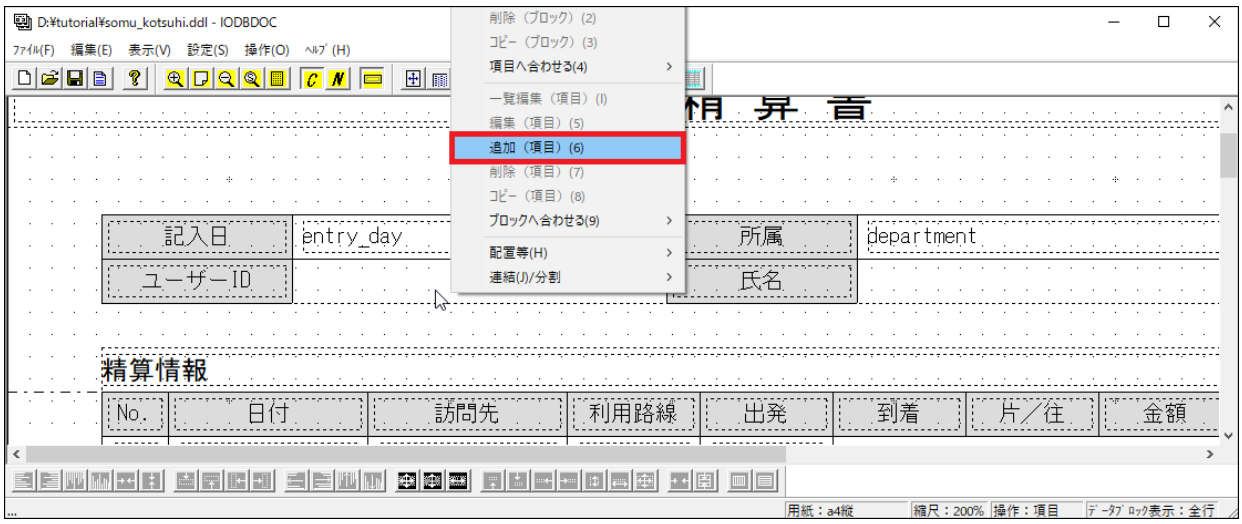
3. 起動した帳票デザインツール IODBDOC のメニューバーから「ファイル」-「開く」を選択し、< somu_kotsuhi.ddl >を選択します。



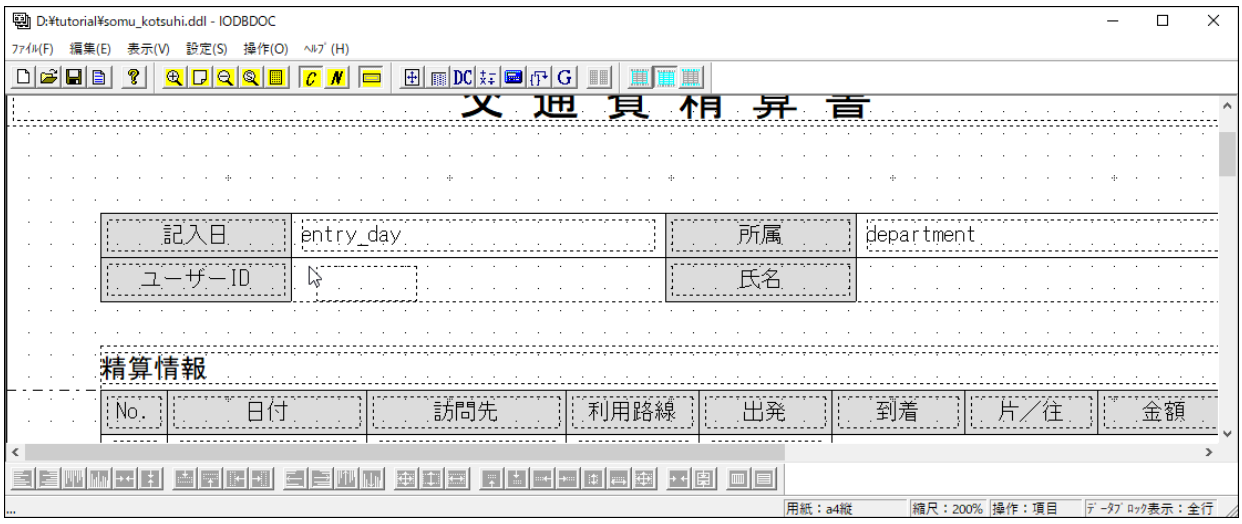
選択したファイルが開きます。

外部変数を設定する
外部変数を追加する

1. レイアウト内の項目「ユーザーID」の横を右クリックし、右クリックメニューから「追加（項目）」を選択します。



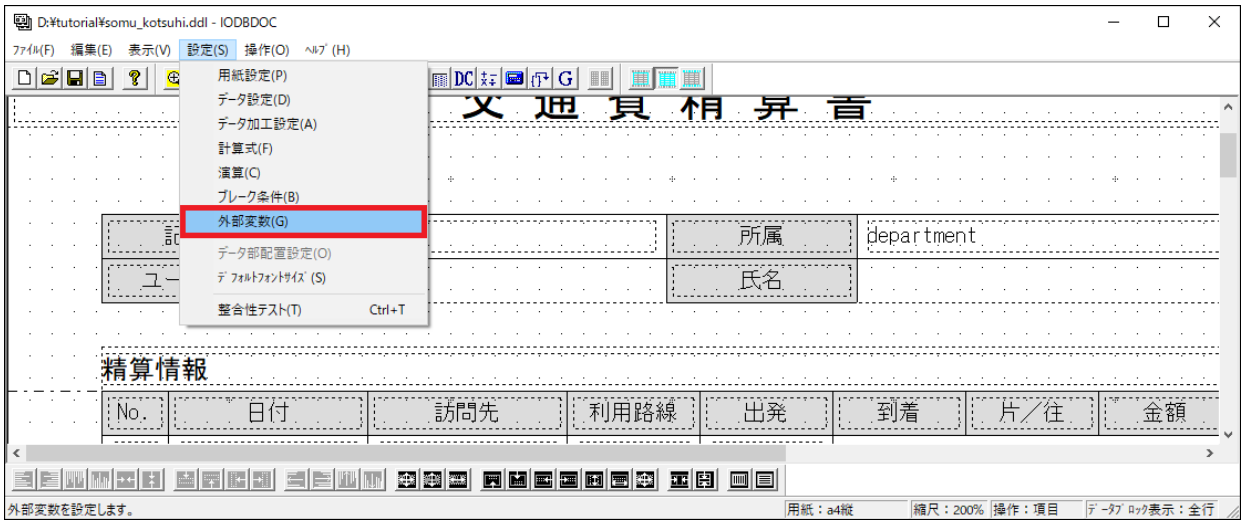
2. マウスカーソルに表示される四角いボックスを、「ユーザーID」の横に合わせ、クリックします。



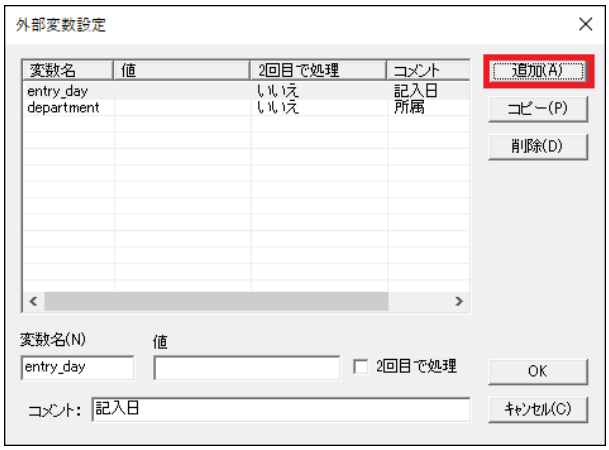
クリックした箇所に項目が追加されます。

外部変数を編集する

1. メニューバーから「設定」-「外部変数」を選択します。



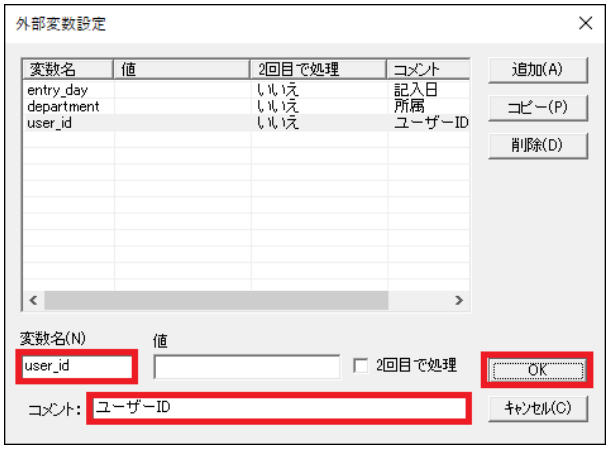
2. 「外部変数設定」画面の「追加」をクリックします。



変数が追加されます。

3. 追加された変数を選択後、画面下の入力欄に、次の値を設定し、「OK」をクリックします。

項目	値
変数名	user_id
コメント	ユーザーID

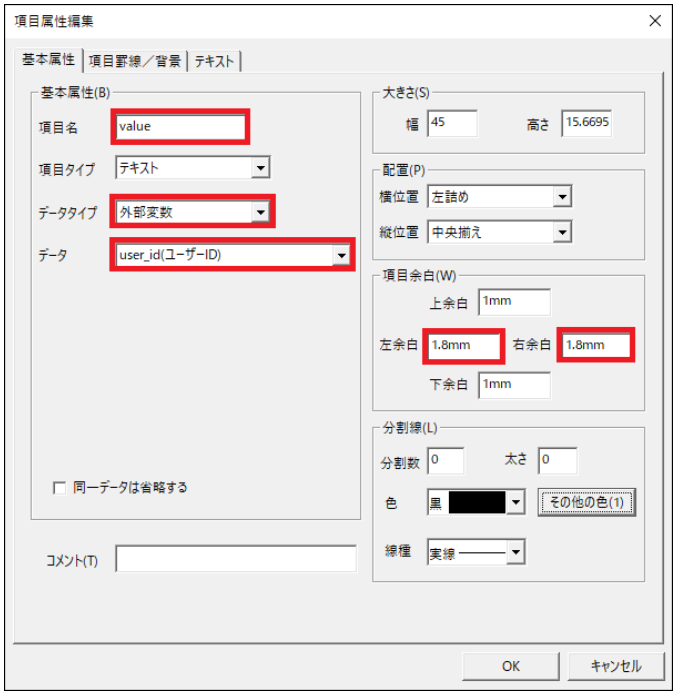


4. 追加した項目のみを選択状態にします。

5. 追加した項目を右クリックし、右クリックメニューから「編集（項目）」を選択します。

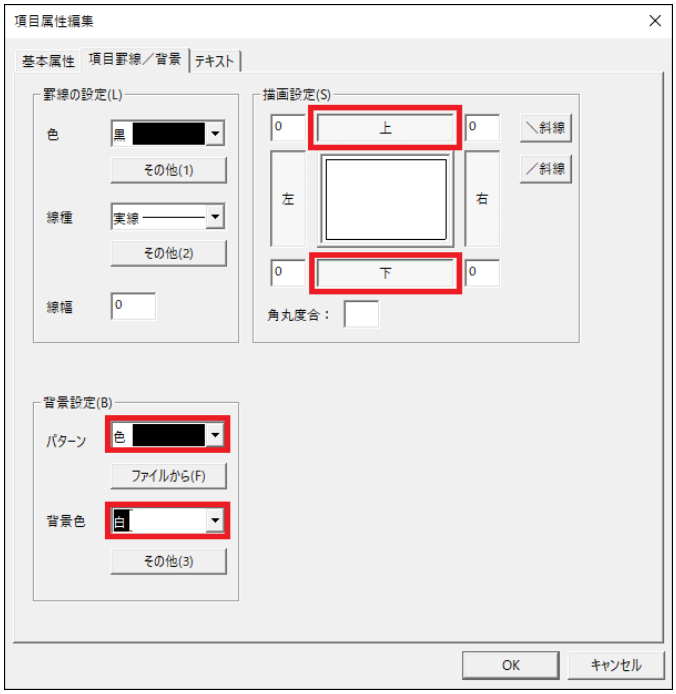
6. 「項目属性編集」画面の「基本属性」タブを開き、次の値を設定します。

項目	値
基本属性 - 項目名	value
基本属性 - データタイプ	外部変数
基本属性 - データ	user_id(ユーザーID)
項目余白 - 左余白	1.8mm
項目余白 - 右余白	1.8mm



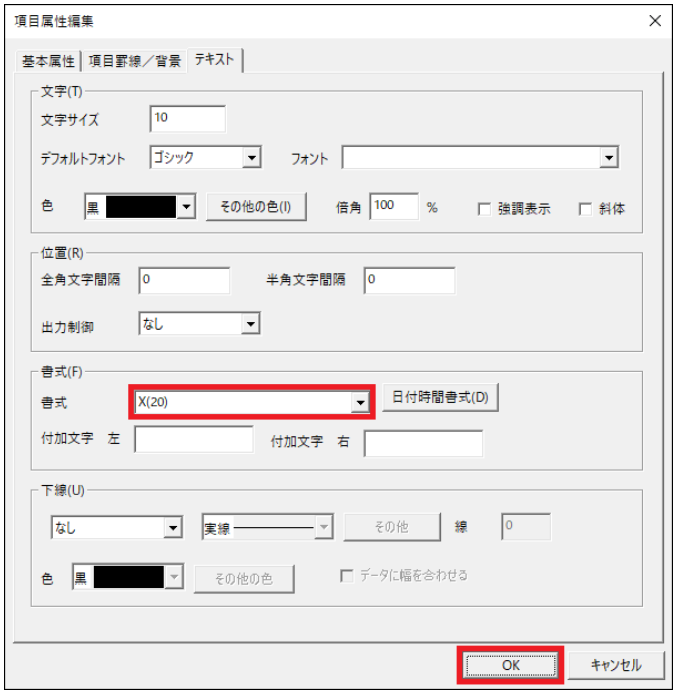
7. 「項目罫線／背景」タブを開き、次の値を設定します。

項目	値
描画設定 - 上	表示（ボタンが窪んでいる状態）
描画設定 - 下	表示（ボタンが窪んでいる状態）
背景設定 - パターン	色
背景設定 - 背景色	白



8. 「テキスト」タブを開き、次の値を設定し、「OK」をクリックします。

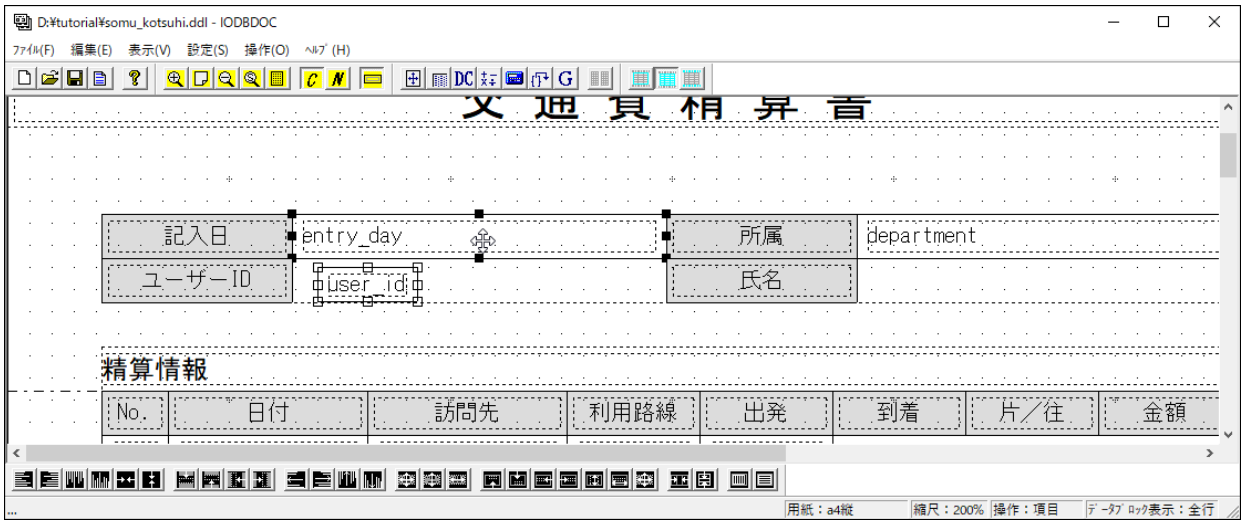
項目	値
書式 - 書式	X(20)



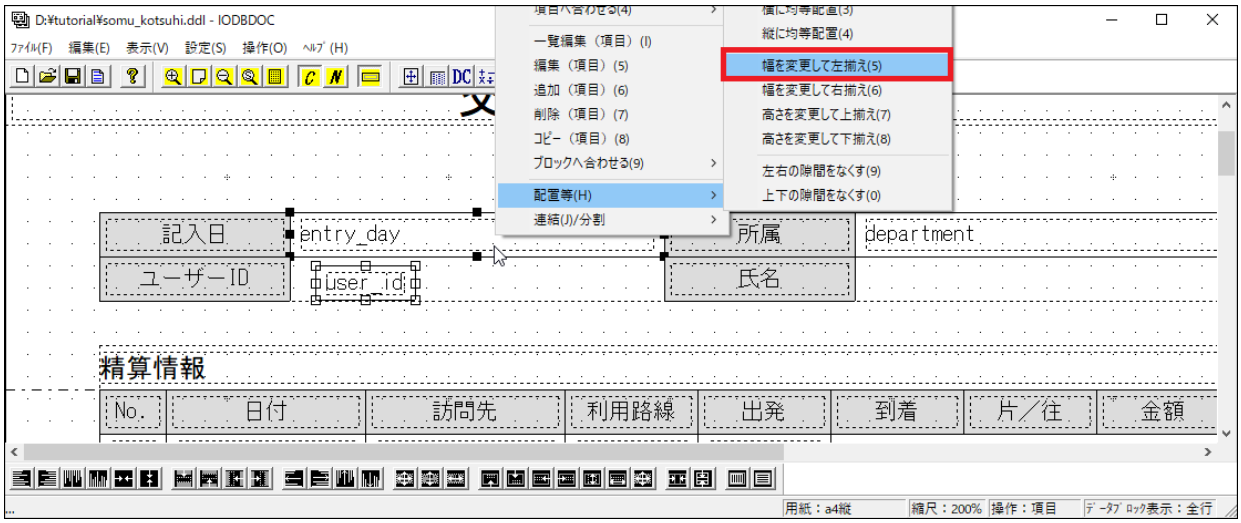
設定が画面上に反映されます。

外部変数の位置を調整する

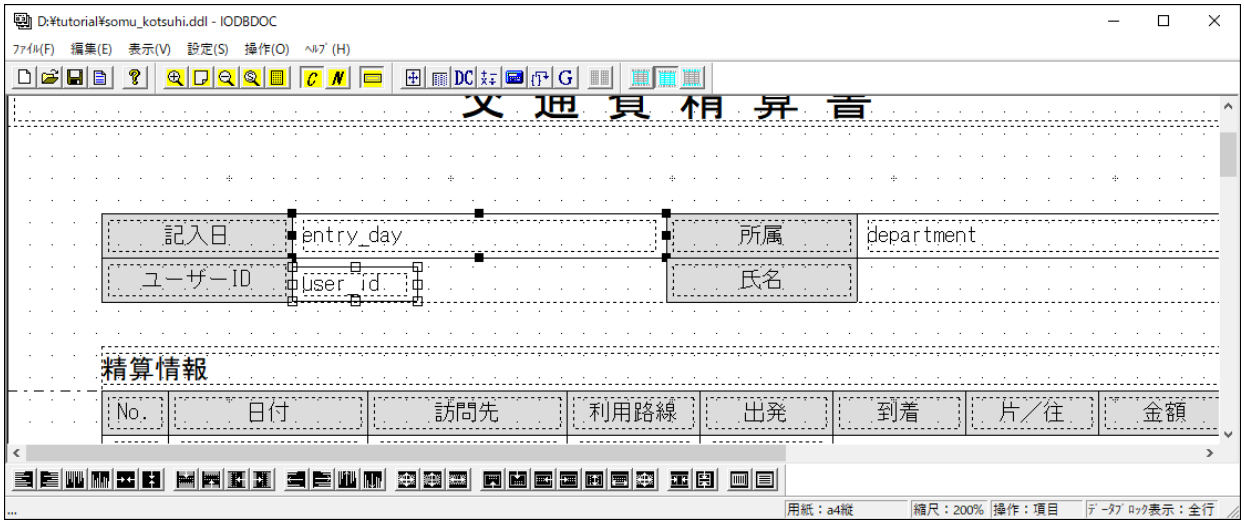
1. 「ユーザーID」の外部変数と、「記入日」の外部変数をクリックします。



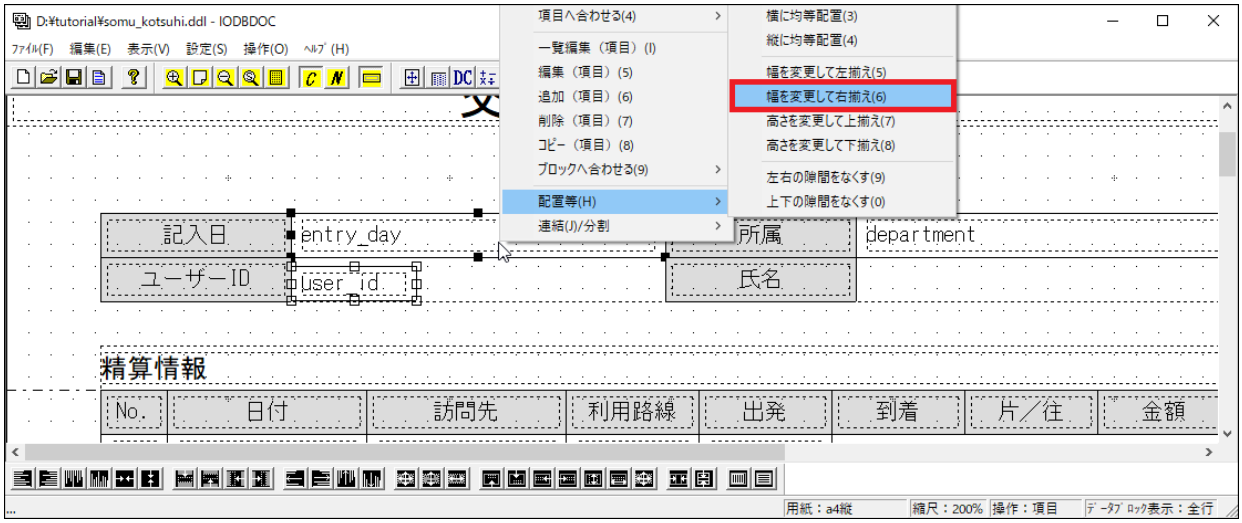
2. 「記入日」の外部変数を右クリックし、右クリックメニューから「配置等」-「幅を変更して左揃え」を選択します。



「ユーザーID」の外部変数の左幅が、「記入日」の外部変数の左幅に合わせて変更されます。

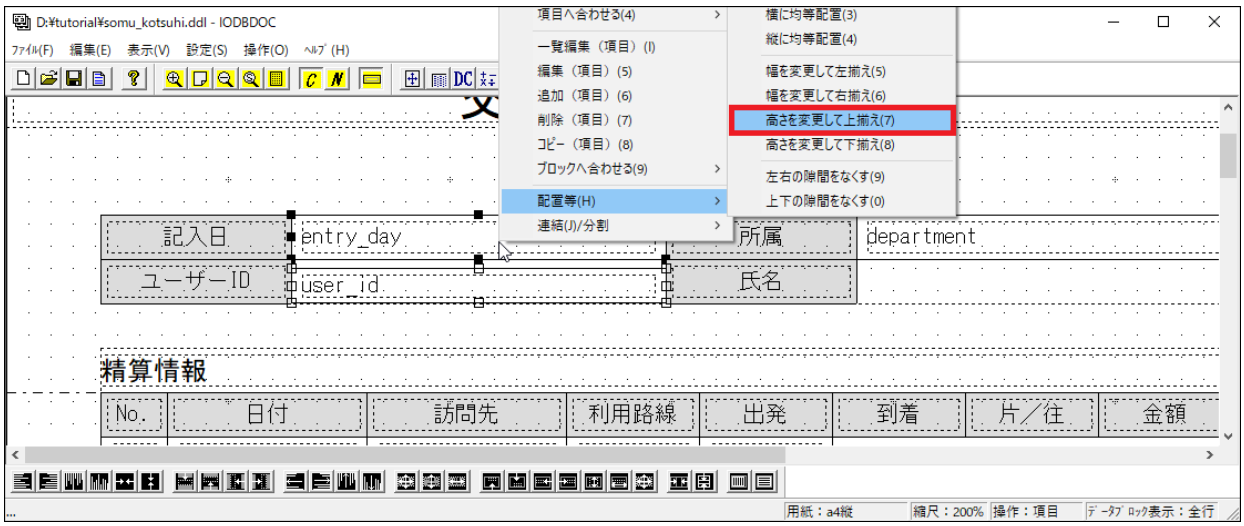


3. 「記入日」の外部変数を右クリックし、右クリックメニューから「配置等」-「幅を変更して右揃え」を選択します。

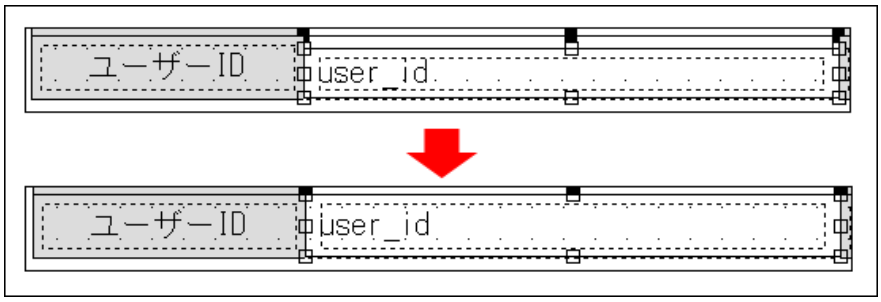


「ユーザーID」の外部変数の右幅が、「記入日」の外部変数の右幅に合わせて変更されます。

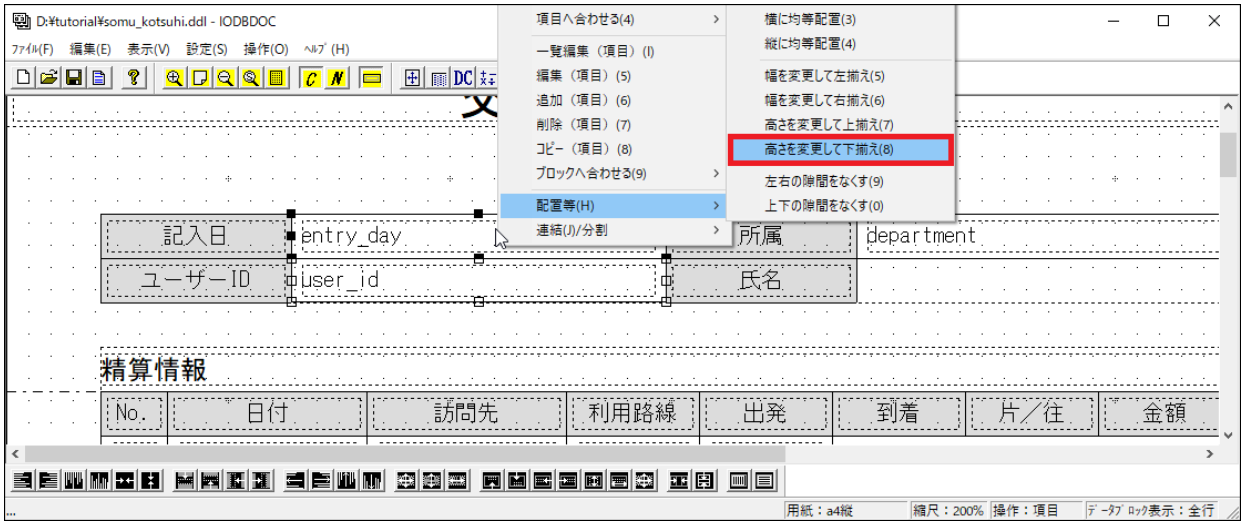
4. 「記入日」の外部変数を右クリックし、右クリックメニューから「配置等」-「高さを変更して上揃え」を選択します。



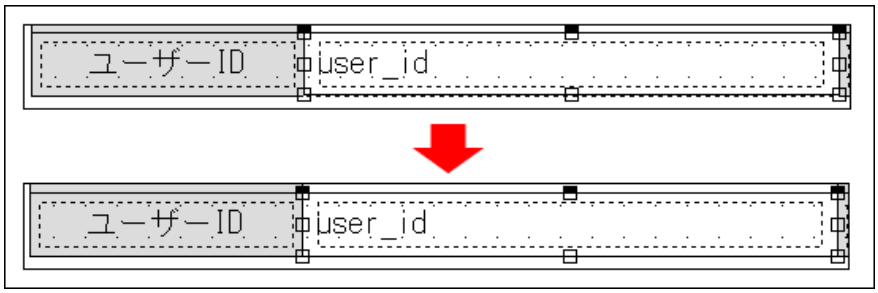
「ユーザーID」の外部変数の上の高さが、「記入日」の外部変数の上の高さに合わせて変更されます。



5. 「記入日」の外部変数を右クリックし、右クリックメニューから「配置等」-「高さを変更して下揃え」を選択します。



「ユーザーID」の外部変数の下の高さが、「記入日」の外部変数の下の高さに合わせて変更されます。



残りの外部変数を全て設定する

上記の手順、および、< layout/somu_kotsuhi_sample.ddl >を参考に、次の外部変数を全て設定します。

- 氏名

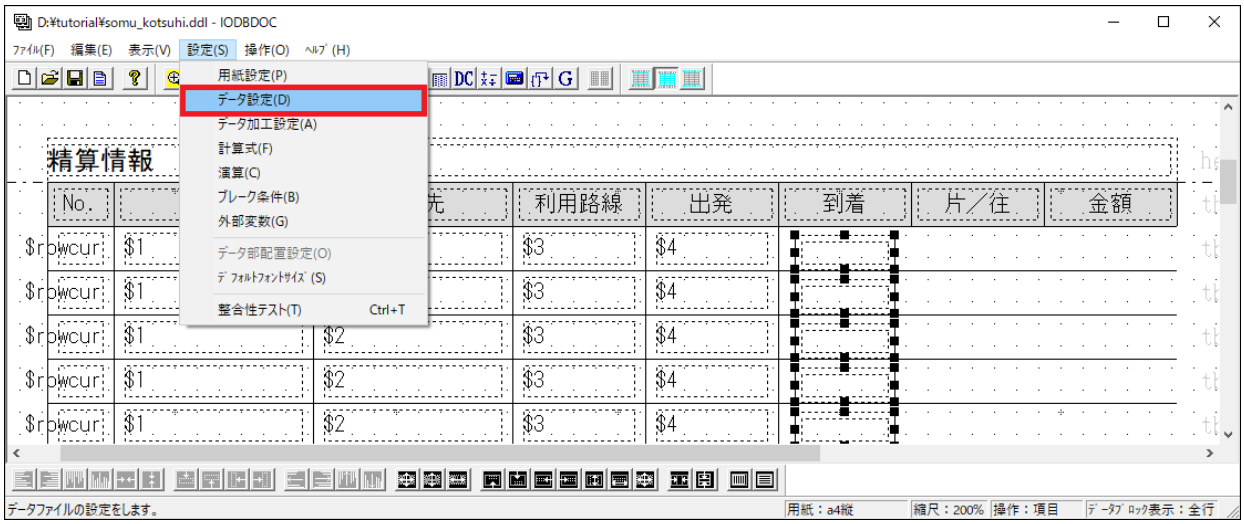
画面	項目	値
外部変数設定	変数名	user_name
	コメント	氏名
項目属性編集 - 基本属性タブ	基本属性 - データ	user_name(氏名)

データ項目変数を設定する
データ項目変数を追加する

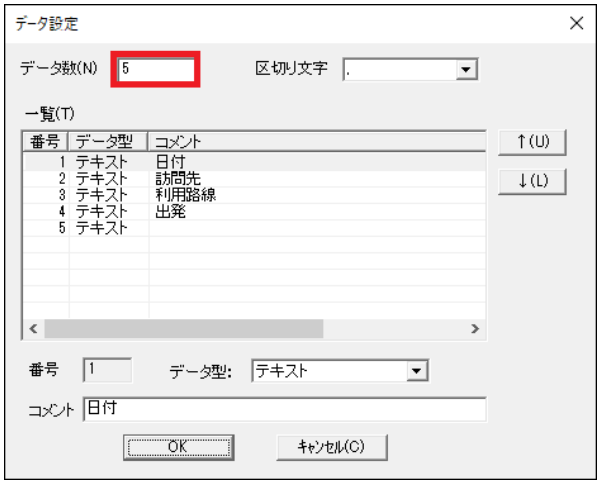
1. 「 *外部変数を追加する* 」を参考に、「到着」の下に項目を追加します。

データ項目変数を編集する

1. メニューバーから「設定」-「データ設定」を選択します。



2. 「データ設定」画面の「データ数」に「5」と入力します。
   入力したタイミングで一覧にデータが追加されます。



3. 追加されたデータを選択後、画面下の入力欄に次の値を設定し、「OK」をクリックします。

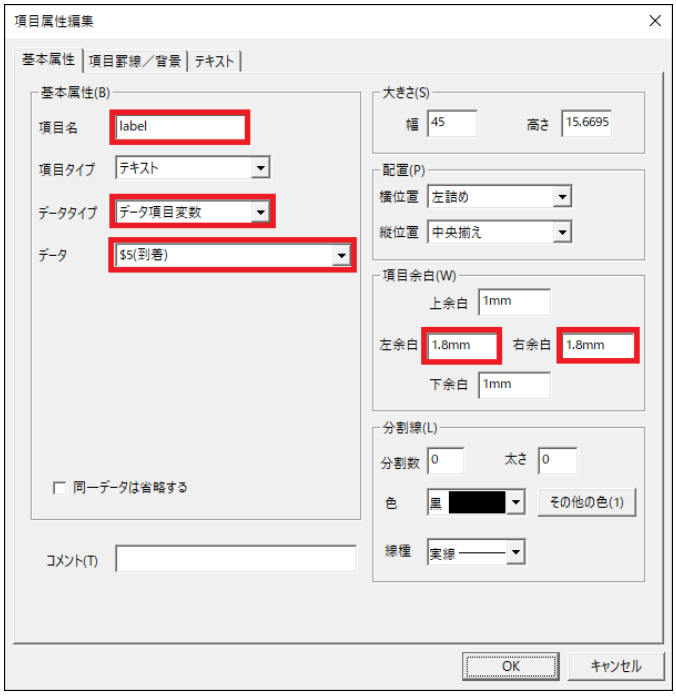
項目	値
コメント	到着

4. 追加した項目のみを選択状態にします。

5. 追加した項目を右クリックし、右クリックメニューから「編集（項目）」を選択します。

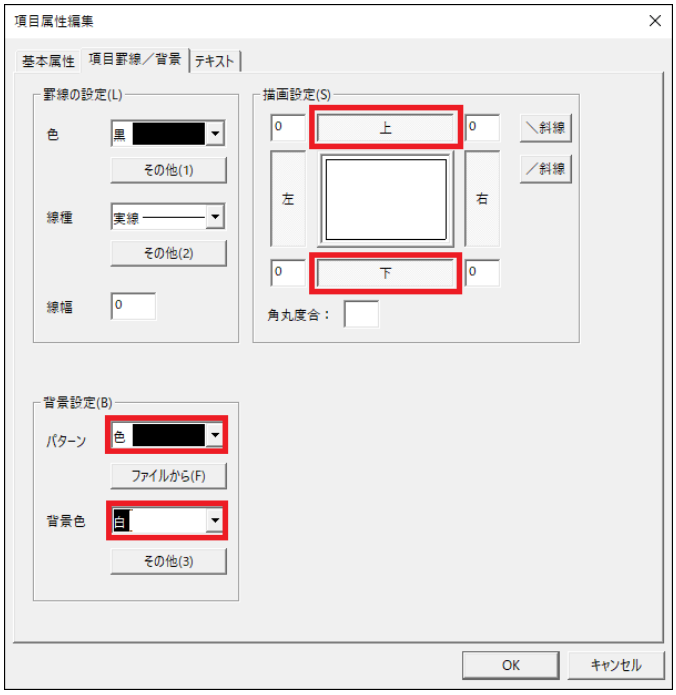
6. 「項目属性編集」画面の「基本属性」タブを開き、次の値を設定します。

項目	値
基本属性 - 項目名	label
基本属性 - データタイプ	データ項目変数
基本属性 - データ	$5(到着)
項目余白 - 左余白	1.8mm
項目余白 - 右余白	1.8mm



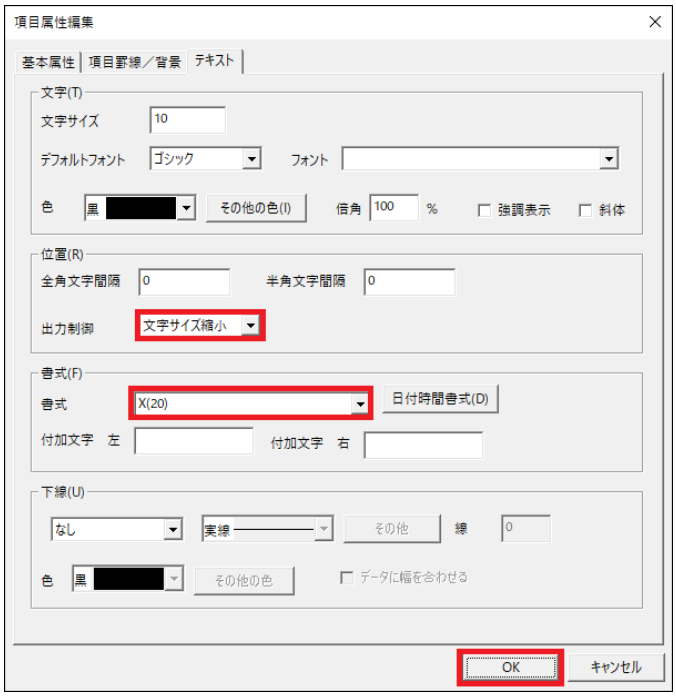
7. 「項目罫線／背景」タブを開き、次の値を設定します。

項目	値
描画設定 - 上	表示（ボタンが窪んでいる状態）
描画設定 - 下	表示（ボタンが窪んでいる状態）
背景設定 - パターン	色
背景設定 - 背景色	白



8. 「テキスト」タブを開き、次の値を設定し、「OK」をクリックします。

項目	値
位置 - 出力制御	文字サイズ縮小
書式 - 書式	X(20)



設定が画面上に反映されます。

データ項目変数の位置を調整する

1. 「 *外部変数の位置を調整する* 」を参考に、「到着」の固定文字の位置に合わせるよう、「到着」のデータ項目変数を調整します。

残りのデータ項目変数を全て設定する

上記の手順、および、< layout/somu_kotsuhi_sample.ddl >を参考に、次のデータ項目変数を全て設定します。

- 片／往

画面	項目	値
データ設定 - 番号6	コメント	片／往
項目属性編集 - 基本属性タブ	基本属性 - データ	$6(片／往)

- 金額

画面	項目	値
データ設定 - 番号7	コメント	金額
項目属性編集 - 基本属性タブ	基本属性 - データ	$7(金額)
項目属性編集 - テキストタブ	書式 - 書式	\\\,\\\,\\9

名前を付けて保存する

1. メニューバーから「ファイル」-「上書き保存」を選択します。
   < somu_kotsuhi.ddl >が上書き保存されます。

> **コラム**
> Ctrl + S でも保存可能です。

作成した帳票レイアウトファイルを設置する

1. < %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >ディレクトリを作成します。
2. < somu_kotsuhi.ddl >を< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >配下に設置します。

帳票レイアウトファイルに渡すデータの準備

帳票レイアウトファイルに埋め込むデータを準備する作業です。

埋め込むデータの指定方法については、「 *埋め込むデータの指定方法について* 」を参照してください。

本項目では、帳票デザインツール IODBDOC の、データファイルの自動作成機能を使用し、CSVファイル、および、DATファイルを作成します。

作成したデータファイルは、プログラム内部で読み込み、プログラム変数を使用してデータを指定します。

手順

- 帳票レイアウトファイルを開く
- CSVファイルを作成する
- DATファイルを作成する
- 作成したデータファイルを設置する

帳票レイアウトファイルを開く

1. 帳票デザインツール IOWebDOC をインストールしているPCに、前章で作成した< somu_kotsuhi.ddl >をコピーします。
2. コピーした< somu_kotsuhi.ddl >を開きます。

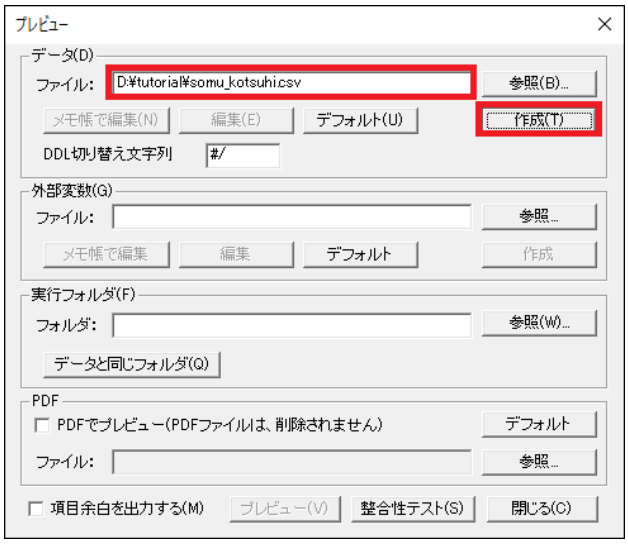
> **コラム**
> ファイルを開く詳細な手順については、「 *帳票レイアウトファイルを開く* 」を参照してください。

CSVファイルを作成する

1. メニューバーから「ファイル」-「プレビュー」を選択します。

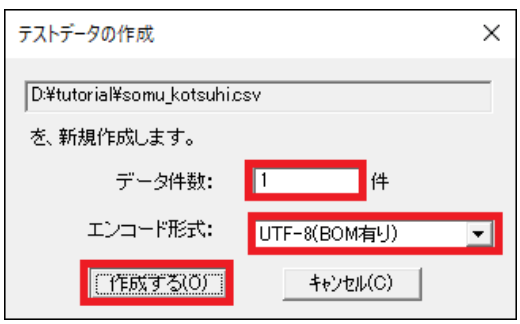
2. 「プレビュー」画面で次の値を設定し、「作成」をクリックします。

項目	値
データ - ファイル	{データファイルの出力先}/somu_kotsuhi.csv

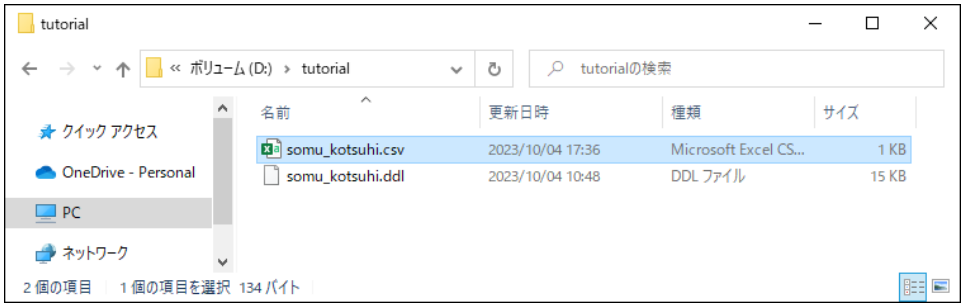


3. 「テストデータの作成」画面で次の値を設定し、「作成する」をクリックします。

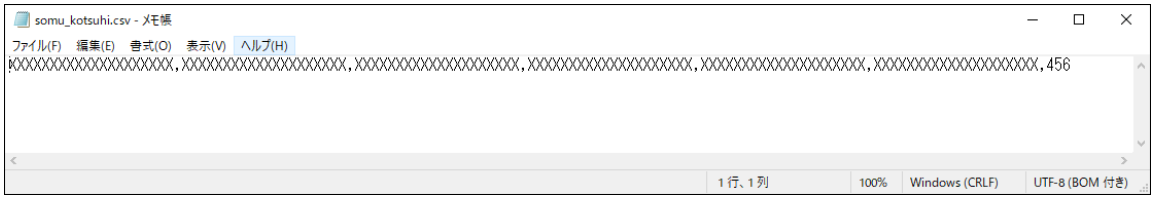
項目	値
データのページ数	1
エンコード形式	UTF-8 (BOM有り)



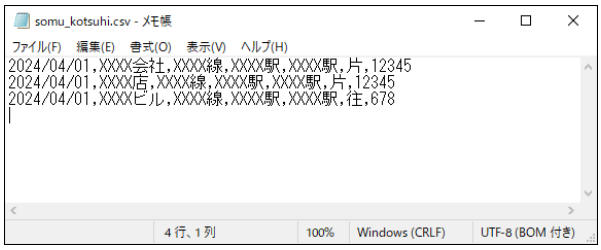
指定した出力先に< somu_kotsuhi.csv >が出力されます。



4. < somu_kotsuhi.csv >をテキストエディタで開きます。



5. < layout/somu_kotsuhi_sample.csv >を参考に、データ項目変数に設定する値を、適切な値に修正し、上書き保存します。



> **注意**
> 文字コードを UTF-8 にして保存してください。

> **注意**
> カンマ区切りでデータを記述してください。
>
> また、データ内にカンマは含めないでください。

> **注意**
> 帳票レイアウトファイルで定義したカラム数と、CSVファイルのカラム数を同じにしてください。

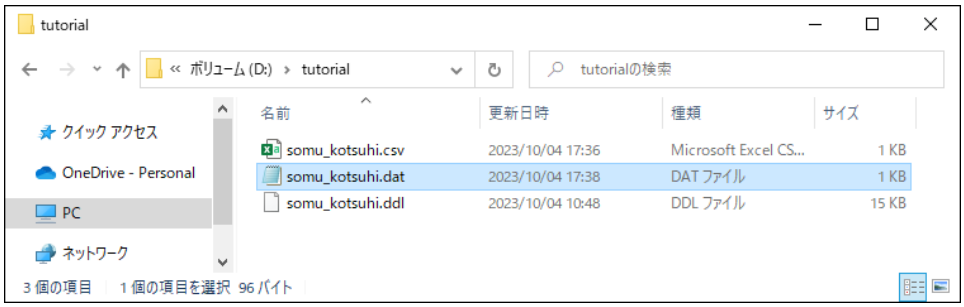
DATファイルを作成する

1. メニューバーから「ファイル」-「プレビュー」を選択します。

2. 「プレビュー」画面で次の値を設定し、「作成」をクリックします。

項目	値
外部変数 - ファイル	{データファイルの出力先}/somu_kotsuhi.dat

指定した出力先に< somu_kotsuhi.dat >が出力されます。



3. < somu_kotsuhi.dat >をテキストエディタで開きます。

> **コラム**
>
> 帳票レイアウトファイル上の外部変数の記載順と、出力したデータファイル内の外部変数の記載順が異なる場合があります。
>
> 記載順が異なっていても、変数名が一致していれば問題はありません。
>
> また、本チュートリアルで追加した外部変数は、データファイルの末尾に記載されている可能性があります。

4. < layout/somu_kotsuhi_sample.dat >を参考に、外部変数に設定する値を、適切な値に修正し、上書き保存します。

> **注意**
> 文字コードを UTF-8 にして保存してください。

> **注意**
> 次のようにデータを記述してください。
>
> ```
> 変数名=値
> entry_day=2024/04/01
> ```

作成したデータファイルを設置する

< somu_kotsuhi.csv >、および、< somu_kotsuhi.dat >を< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >配下に設置します。

PDFファイルの作成

IM-PDFDesigner for Accel Platform を使用し、帳票レイアウトファイルと埋め込むデータを組み合わせて、PDFファイルを作成する作業です。

スクリプト開発モデル 、または、 JavaEE開発モデル を使用し、プログラムを作成します。

スクリプト開発モデル

本項目では、 スクリプト開発モデル 用APIを使用したプログラムを作成し、そのプログラムを実行することで、PDFファイルを作成します。

手順

- プログラムを作成する
  - 実行画面用のHTMLファイルを設置する
  - PDFファイル出力処理用のJSファイルを作成する
- 設定ファイルを設置する
- プログラムを登録する
  - 認可を設定する
  - メニューを設定する
- プログラムを実行・確認する

プログラムを作成する
実行画面用のHTMLファイルを設置する

1. < %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/jssp/src/pdfd/tutorial/basic >ディレクトリを作成します。
2. < jssp/iodbdoc.html >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/jssp/src/pdfd/tutorial/basic >配下に設置します。

PDFファイル出力処理用のJSファイルを作成する

1. < jssp/iodbdoc.js >をテキストエディタで開きます。
2. 3行目を次のように修正し、帳票レイアウトファイルを指定します。

```
const ddlPath = "/pdfd/tutorial/basic/somu_kotsuhi.ddl";
```

3. 4行目を次のように修正し、CSVファイルを指定します。

```
const colDataFilePath = "/pdfd/tutorial/basic/somu_kotsuhi.csv";
```

4. 5行目を次のように修正し、DATファイルを指定します。

```
const globalDataFilePath = "/pdfd/tutorial/basic/somu_kotsuhi.dat";
```

5. 11行目を次のように修正し、PDFファイルの出力先フォルダを指定します。

```
const dirPath = "/pdfd/tutorial/basic/";
```

6. 12行目を次のように修正し、PDFファイル名の接頭文字を指定します。

```
const prefix = "somu_kotsuhi";
```

7. 20行目を次のように修正し、クラスを指定します。
   - スタンドアローン構成 の場合

```
const pdf = new IODBDoc(ddlPath);
```

   - 分散構成 の場合

```
const pdf = new IODBDocRemote(ddlPath);
```

8. 上書き保存します。

> **注意**
> 文字コードを UTF-8 にして保存してください。

9. < jssp/iodbdoc.js >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/jssp/src/pdfd/tutorial/basic >配下に設置します。

設定ファイルを設置する

< jssp/pdfd-tutorial_basic_iodbdoc.xml >を< %RESIN_HOME%/webapps/{アプリケーション名}/WEB-INF/conf/routing-jssp-config >配下に設置します。

プログラムを登録する

設置したプログラム、および、設定ファイルを環境に適用するため、 Web Application Server を再起動します。

再起動後、プログラムを認可とメニューに設定します。

> **コラム**
> テナント管理に関する詳細な手順については、次を参照してください。
>
> - 「 テナント管理者操作ガイド 」-「 認可を設定する 」
> - 「 テナント管理者操作ガイド 」-「 メニューを設定する 」

> **注意**
> 同一の認可やメニューグループ、メニューフォルダを既に設定している場合、該当する手順は不要です。

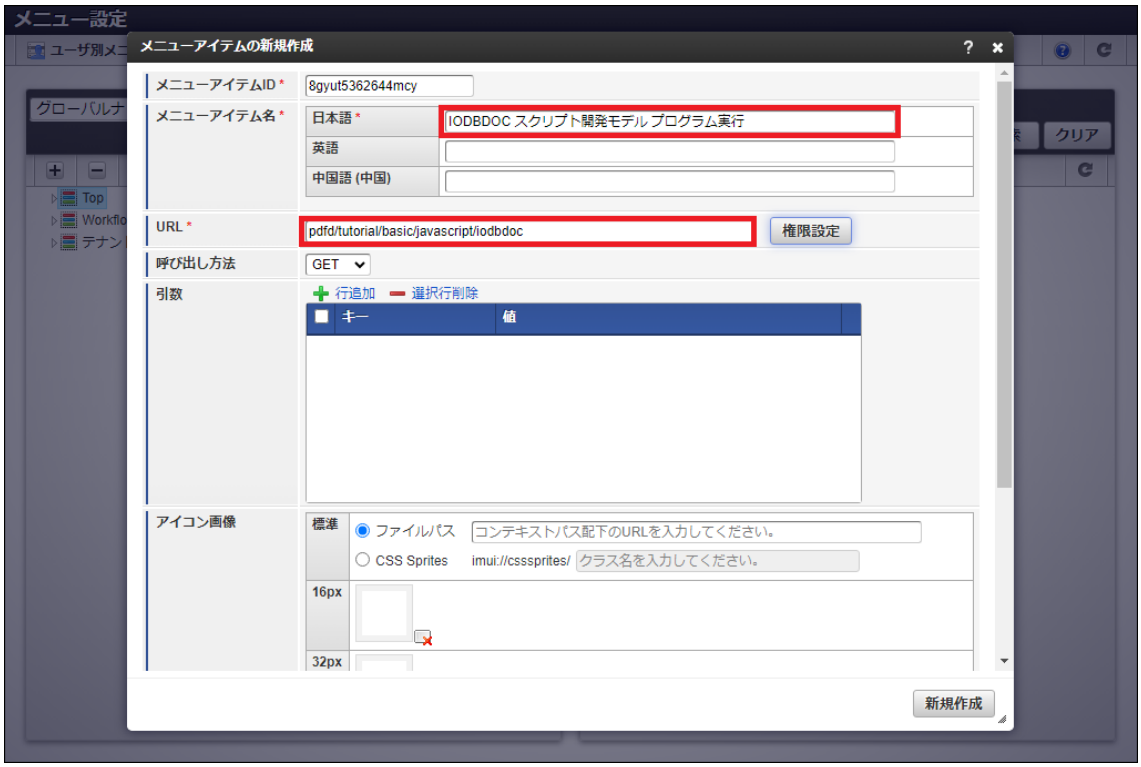
認可を設定する

「 *認可を設定する* 」を参照し、認可を設定してください。

メニューを設定する

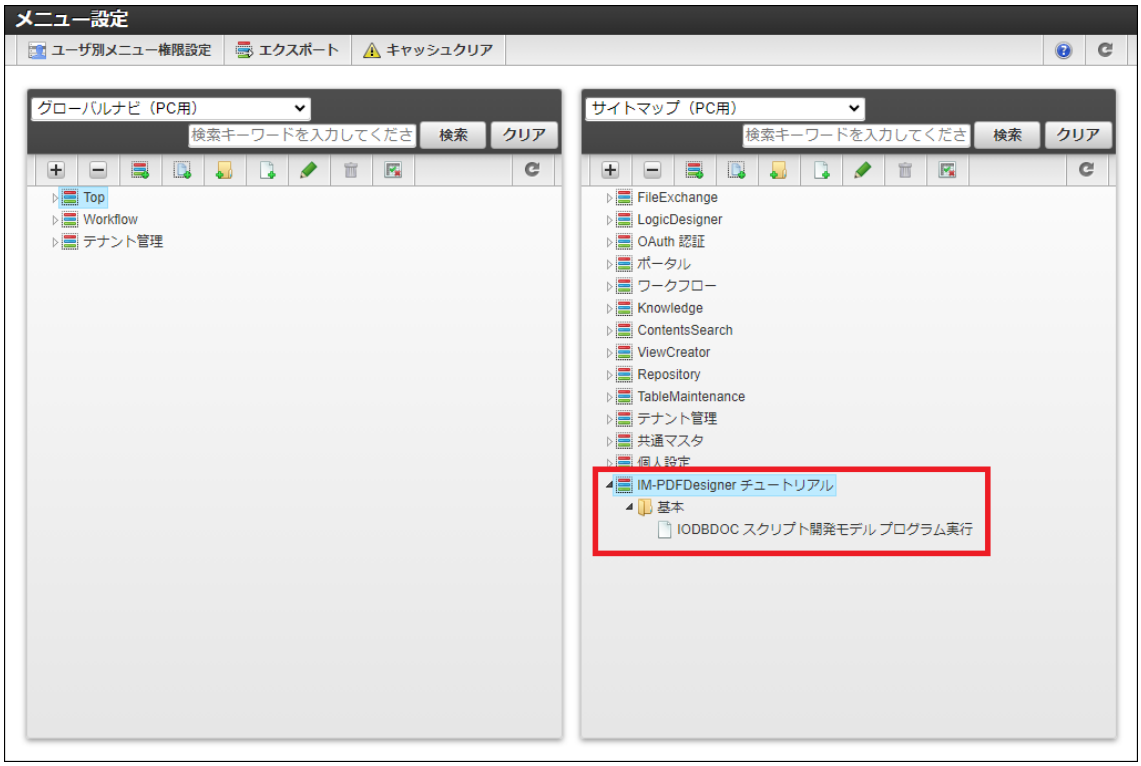
1. テナント管理の権限を持つユーザで intra-mart Accel Platform にログインします。
2. 「 *メニューを設定する* 」を参照し、メニューグループ、および、メニューフォルダを設定します。
3. メニューフォルダ「基本」配下に、次のメニューアイテムを作成します。

項目	値
メニューアイテム名 - 日本語	IODBDOC スクリプト開発モデル プログラム実行
URL	pdfd/tutorial/basic/javascript/iodbdoc



メニューの設定ができました。メニュー構成は、次の通りです。

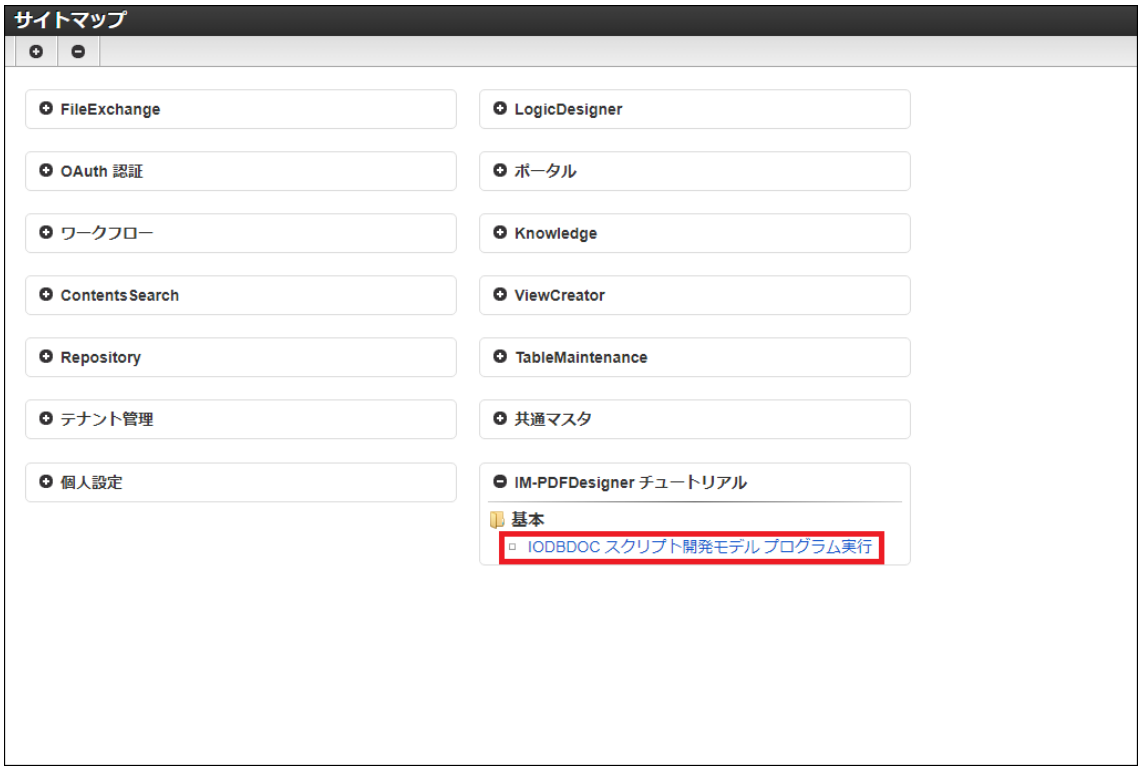
- メニューグループ： IM-PDFDesigner チュートリアル
- メニューフォルダ： 基本
- メニューアイテム： IODBDOC スクリプト開発モデル プログラム実行



プログラムを実行・確認する

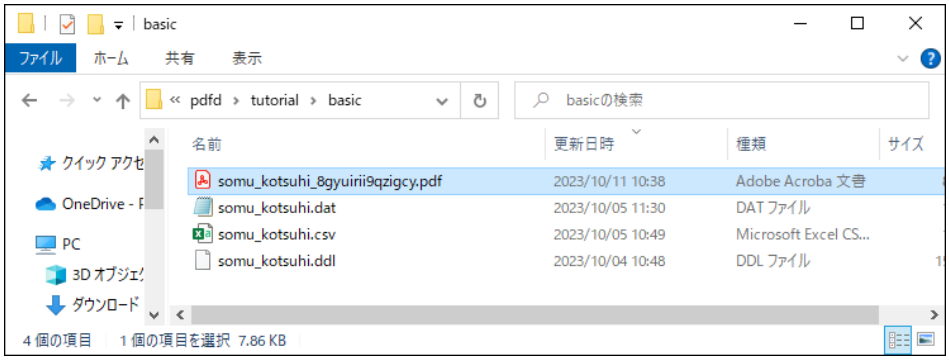
1. 一般ユーザで intra-mart Accel Platform にログインします。
2. 「サイトマップ」-「 IM-PDFDesigner チュートリアル 」-「基本」-「 IODBDOC スクリプト開発モデル プログラム実行 」をク

リックします。



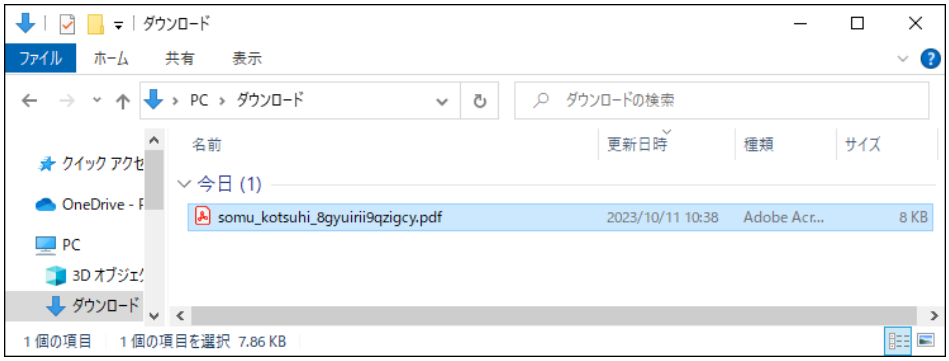
3. 表示された実行画面の「PDF作成」をクリックします。

プログラムが実行され、< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >にPDFファイルが出力されます。



> **コラム**
> エラーが発生した場合は、エラーメッセージの内容に従いファイルの修正等を行ってください。

また、処理終了後にPDFファイルがダウンロードされます。



4. 出力されたPDFファイルをPDFビューア（ Adobe Acrobat Reader など）で開き、正しく表示されることを確認します。

JavaEE開発モデル

本項目では、 JavaEE開発モデル 用APIを使用したプログラムを作成し、そのプログラムを実行することで、PDFファイルを作成します。

手順

- プログラムを作成する
  - 実行画面用のJSPファイルを設置する
  - PDFファイル出力処理用のJSPファイルを作成する
  - ダウンロード処理用のJSPファイルを設置する
- 設定ファイルを設置する
- プログラムを登録する
  - 認可を設定する
  - メニューを設定する
- プログラムを実行・確認する

プログラムを作成する

実行画面用のJSPファイルを設置する

1. < %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/basic >ディレクトリを作成します。

2. < javaee/iodbdoc.jsp >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/basic >配下に設置します。

PDFファイル出力処理用のJSPファイルを作成する

1. < javaee/iodbdoc_act.jsp >をテキストエディタで開きます。
2. 12行目を次のように修正し、クラスを指定します。
   - スタンドアローン構成 の場合

     ```
     <%@ page import="jp.co.intra_mart.product.pdfmaker.net.DBDoc" %>
     ```

   - 分散構成 の場合

     ```
     <%@ page import="jp.co.intra_mart.product.pdfmaker.net.rest.DBDocRemote" %>
     ```

3. 16行目を次のように修正し、帳票レイアウトファイルを指定します。

   ```
   String ddlPath = "pdfd/tutorial/basic/somu_kotsuhi.ddl";
   ```

4. 17行目を次のように修正し、CSVファイルを指定します。

   ```
   String colDataFilePath = "pdfd/tutorial/basic/somu_kotsuhi.csv";
   ```

5. 18行目を次のように修正し、DATファイルを指定します。

   ```
   String globalDataFilePath = "pdfd/tutorial/basic/somu_kotsuhi.dat";
   ```

6. 26行目を次のように修正し、PDFファイルの出力先フォルダを指定します。

   ```
   String dirPath = "pdfd/tutorial/basic/";
   ```

7. 27行目を次のように修正し、PDFファイル名の接頭文字を指定します。

   ```
   String prefix = "somu_kotsuhi";
   ```

8. 35行目を次のように修正し、クラスを指定します。
   - スタンドアローン構成 の場合

     ```
     DBDoc pdf = new DBDoc(ddlPath);
     ```

   - 分散構成 の場合

     ```
     DBDocRemote pdf = new DBDocRemote(ddlPath);
     ```

9. 上書き保存します。

   > **注意**
   > 文字コードを UTF-8 にして保存してください。

10. < javaee/iodbdoc_act.jsp >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/basic >配下に設置します。

ダウンロード処理用のJSPファイルを設置する

< javaee/download.jsp >を< %RESIN_HOME%/webapps/{warファイルと同名のディレクトリ}/WEB-INF/view/pdfd/tutorial/basic >配下に設置します。

> **注意**
> 同一のファイルを既に設置している場合、該当する手順は不要です。

設定ファイルを設置する

1. < javaee/service-config-pdfd_tutorial_basic_iodbdoc.xml >を< %RESIN_HOME%/webapps/{アプリケーション

名}/WEB-INF/classes >配下に設置します。

2. < javaee/pdfd-tutorial_basic_iodbdoc.xml >を< %RESIN_HOME%/webapps/{アプリケーション名}/WEB-INF/conf/routing-service-config >配下に設置します。

プログラムを登録する

設置したプログラム、および、設定ファイルを環境に適用するため、 Web Application Server を再起動します。

再起動後、プログラムを認可とメニューに設定します。

> **コラム**
>
> テナント管理に関する詳細な手順については、次を参照してください。
>
> - 「テナント管理者操作ガイド」-「認可を設定する」
> - 「テナント管理者操作ガイド」-「メニューを設定する」

> **注意**
>
> 同一の認可やメニューグループ、メニューフォルダを既に設定している場合、該当する手順は不要です。

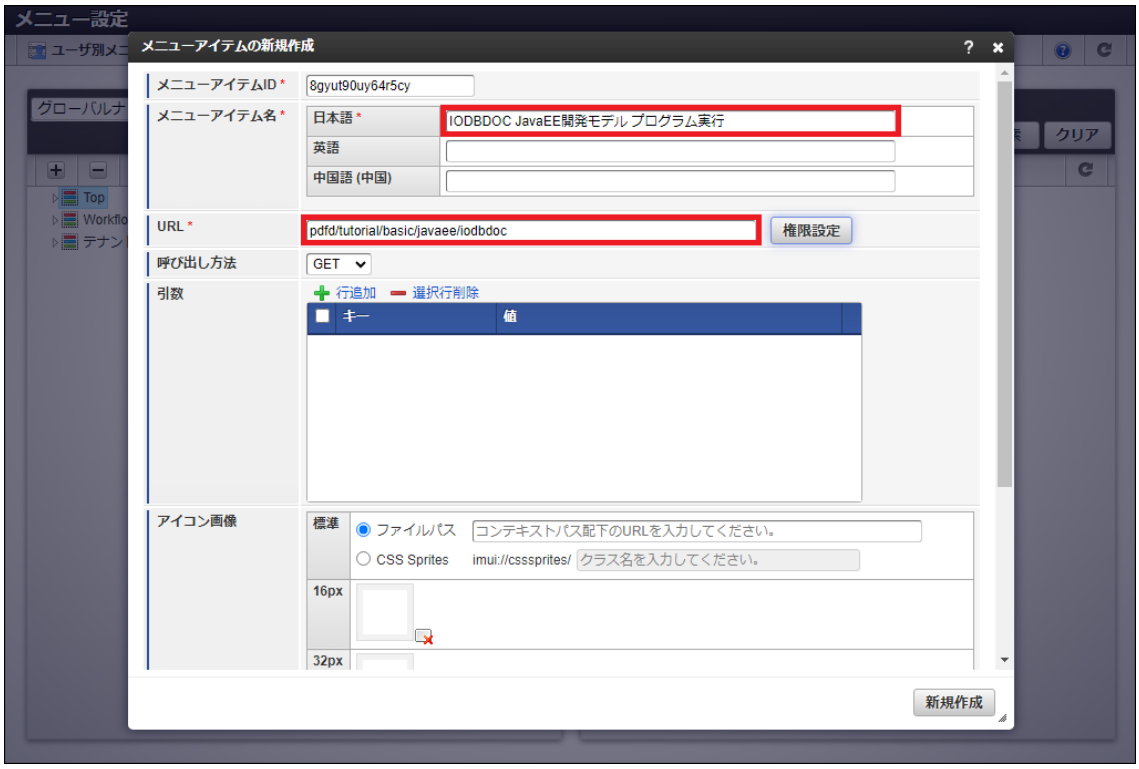
認可を設定する

「*認可を設定する*」を参照し、認可を設定してください。

メニューを設定する

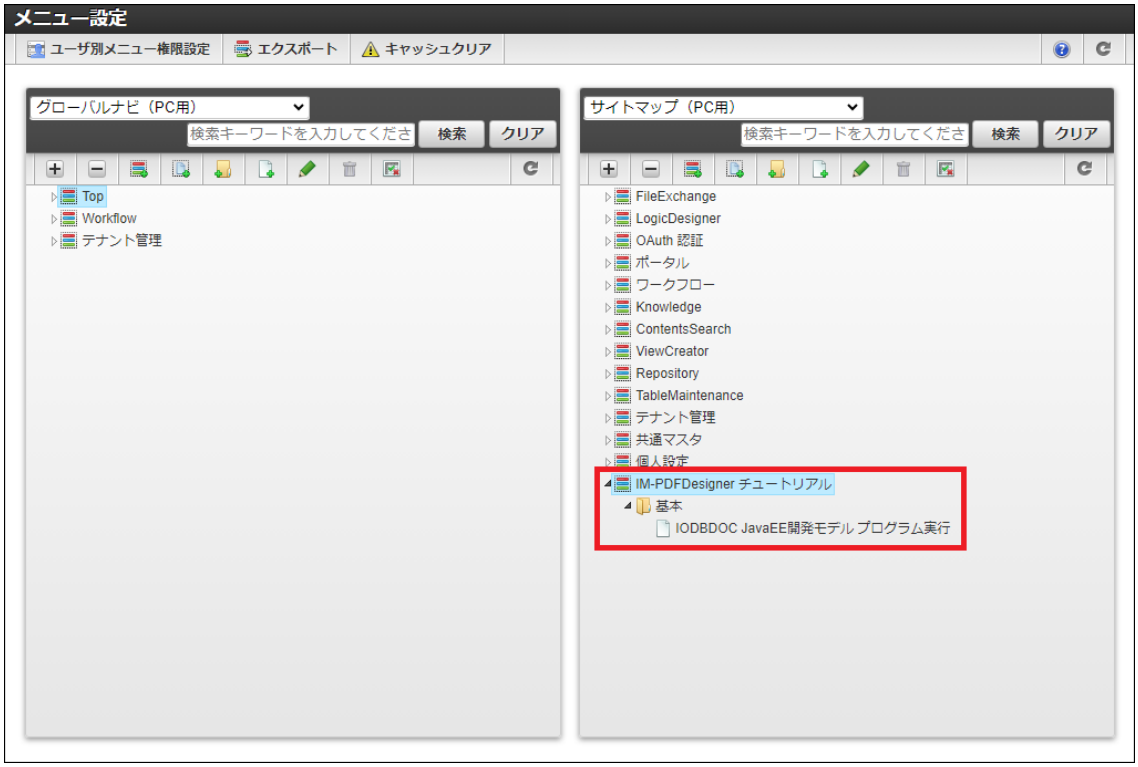
1. テナント管理の権限を持つユーザで intra-mart Accel Platform にログインします。
2. 「*メニューを設定する*」を参照し、メニューグループ、および、メニューフォルダを設定します。
3. メニューフォルダ「基本」配下に、次のメニューアイテムを作成します。

項目	値
メニューアイテム名 - 日本語	IODBDOC JavaEE開発モデル プログラム実行
URL	pdfd/tutorial/basic/javaee/iodbdoc



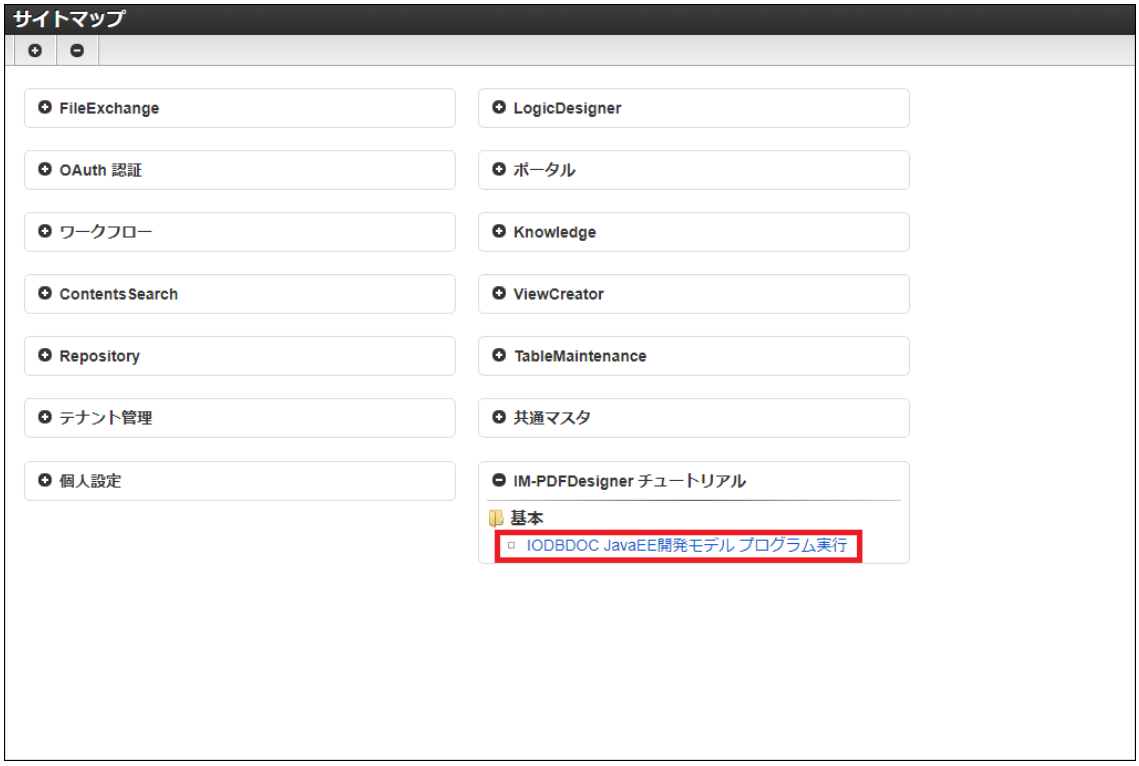
メニューの設定ができました。メニュー構成は、次の通りです。

- メニューグループ： IM-PDFDesigner チュートリアル
- メニューフォルダ： 基本
- メニューアイテム： IODBDOC JavaEE開発モデル プログラム実行



プログラムを実行・確認する

1. 一般ユーザで intra-mart Accel Platform にログインします。
2. 「サイトマップ」-「 IM-PDFDesigner チュートリアル 」-「基本」-「 IODBDOC JavaEE開発モデル プログラム実行 」をクリックします。

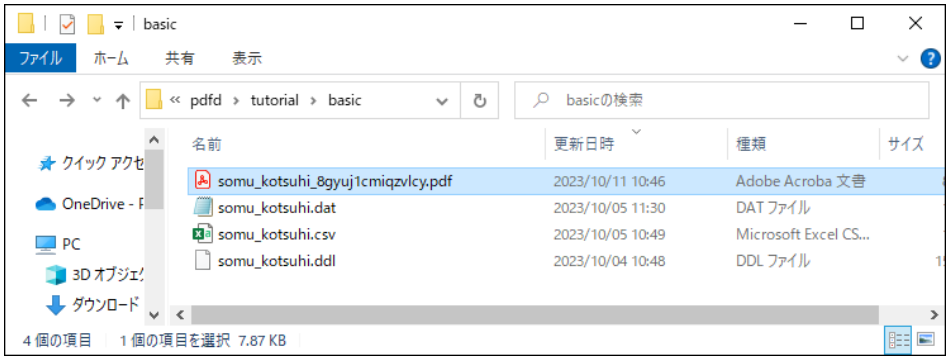


3. 表示された実行画面の「PDF作成」をクリックします。

プログラムが実行され、実行結果の画面に遷移します。

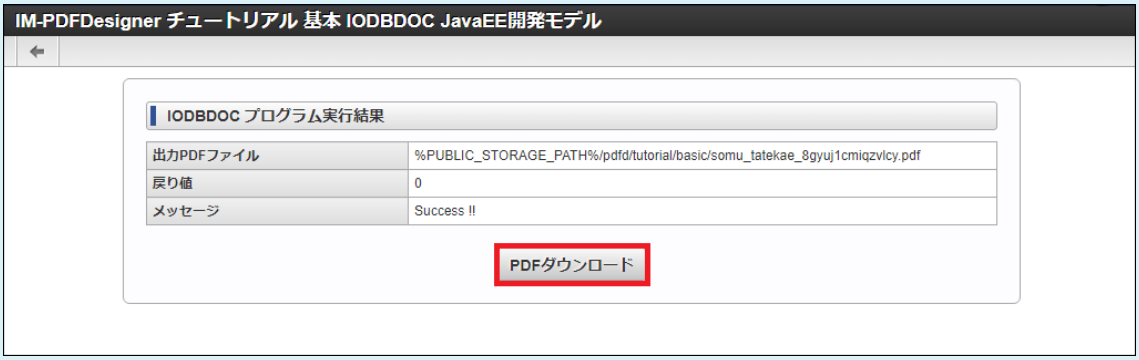
> **コラム**
> エラーが発生した場合は、エラーメッセージの内容に従いファイルの修正等を行ってください。

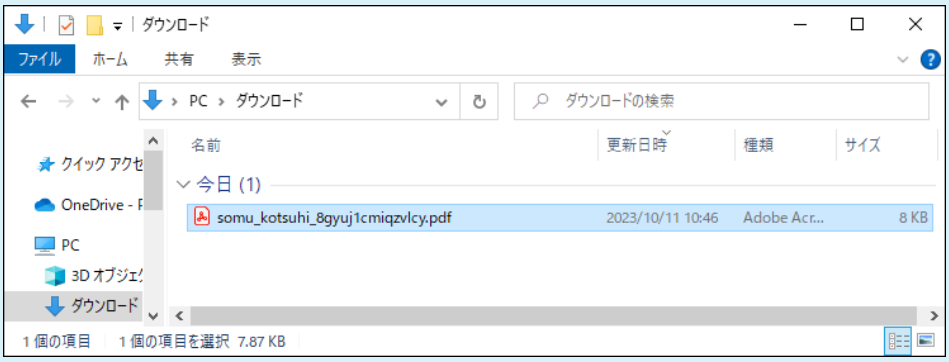
また、< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/basic >にPDFファイルが出力されます。



コラム

実行結果画面内の「PDFダウンロード」をクリックすると、PDFファイルがダウンロードされます。





4. 出力されたPDFファイルをPDFビューア（ Adobe Acrobat Reader など）で開き、正しく表示されることを確認します。

応用

- 概要
- 前提条件
- 実践

概要

本チュートリアルでは、プログラミング言語を使用したPDFファイルの応用的な作成方法を説明します。

前提条件

本チュートリアルを進めるにあたり、次の事前準備が行われていることが前提となります。

- IM-PDFDesigner for Accel Platform 、または、 IM-PDFDesigner FullPack for Accel Platform のセットアップが完了していること。
- intra-mart Accel Platform のテナント環境セットアップが完了していること。

実践

IM-LogicDesigner を利用したPDFファイル作成

本チュートリアルでは、「 *結合（単票形式／連票形式／複数の票形式）* 」の作成方法を基に、中間ファイル（IODファイル）の作成、結合、および、PDFファイルを作成します。

IM-LogicDesigner 上で利用できるように、処理は IM-PDFDesigner for Accel Platform のAPIを使用した JavaScript定義 として作成します。

> **コラム**
> FullPack版 IM-LogicDesigner 連携 を使用し、GUI上の操作だけ（ノーコード）で中間ファイルの作成、結合、および、PDFファイルを作成する方法については、「 IM-PDFDesigner FullPack for Accel Platform IM-LogicDesigner連携 操作ガイド 」-「 操作方法 」を参照してください。

チュートリアルを実施するにあたり、「 *前提条件* 」に加え、次の事前準備が行われていることが前提となります。

- IM-LogicDesigner のセットアップが完了していること。
- 2017 Summer 以降のバージョンの IM-PDFDesigner for Accel Platform 、または、 IM-PDFDesigner FullPack for Accel Platform のセットアップが完了していること。

また、次のzipファイルをダウンロードし、解凍してください。

構成	zipファイル
スタンドアローン構成	**tutorial_application_integration_ld.zip**
分散構成	**tutorial_application_integration_ld_rest.zip**

解凍したファイルの構成は、次の通りです。

フォルダ名／ファイル名	説明
layout/	帳票レイアウト関連フォルダ
iodoc/	IODOC 用帳票レイアウト関連フォルダ
iodbdoc/	IODBDOC 用帳票レイアウト関連フォルダ
import/	IM-LogicDesigner インポート関連フォルダ
im_logicdesigner-data.zip	IOD／PDFファイル出力用ユーザ定義、フロー定義をまとめたzipファイル

< import/im_logicdesigner-data.zip >を、 IM-LogicDesigner のインポート画面からインポートしてください。

次の手順に沿って、チュートリアルを進めます。

PDFファイルの作成

本項目では、ロジックフローのデバッグ機能を利用し、中間ファイル（IODファイル）を作成・結合後、PDFファイルを作成します。

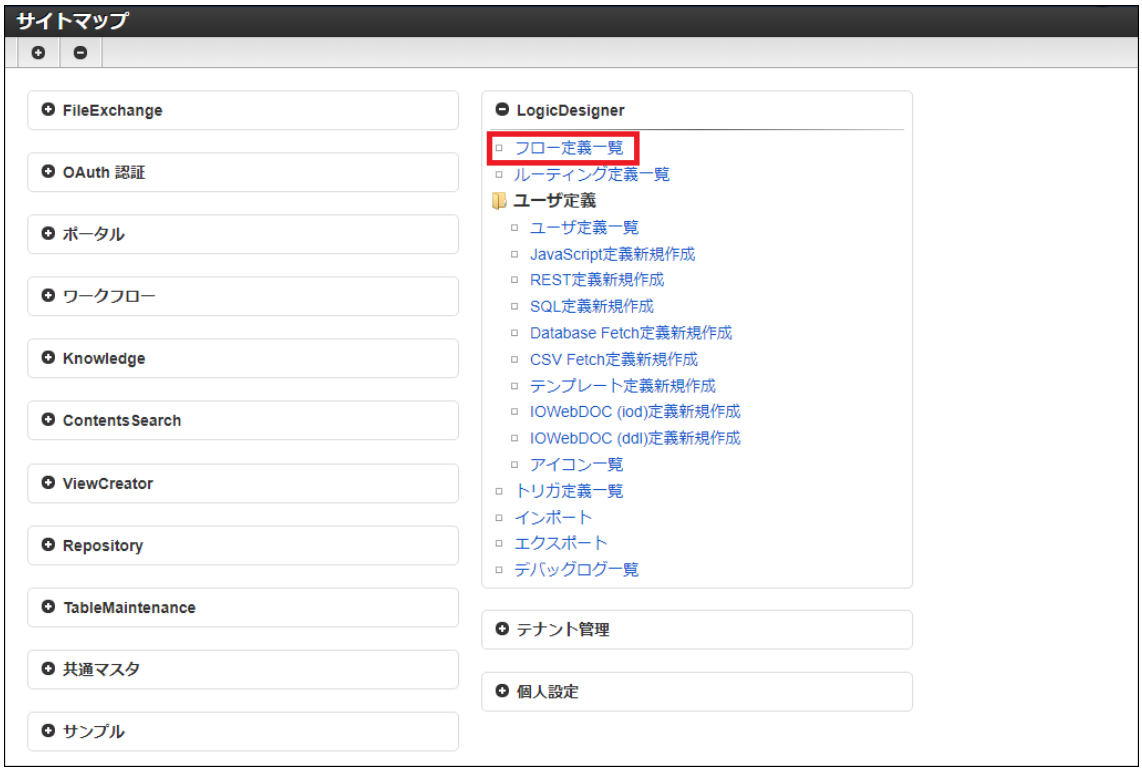
- 帳票レイアウトファイル／データファイルを設置する
- ロジックフロー定義のデバッグ画面を開く
- 入力値を設定し、デバッグを実行する
- 実行結果を確認する

帳票レイアウトファイル／データファイルを設置する

1. < %PUBLIC_STORAGE_PATH% /pdfd/tutorial/application/integration_ld >ディレクトリを作成します。
2. < layout/iodoc >フォルダ、および、< layout/iodbdoc >フォルダを< %PUBLIC_STORAGE_PATH% /pdfd/tutorial/application/integration_ld >配下に設置します。

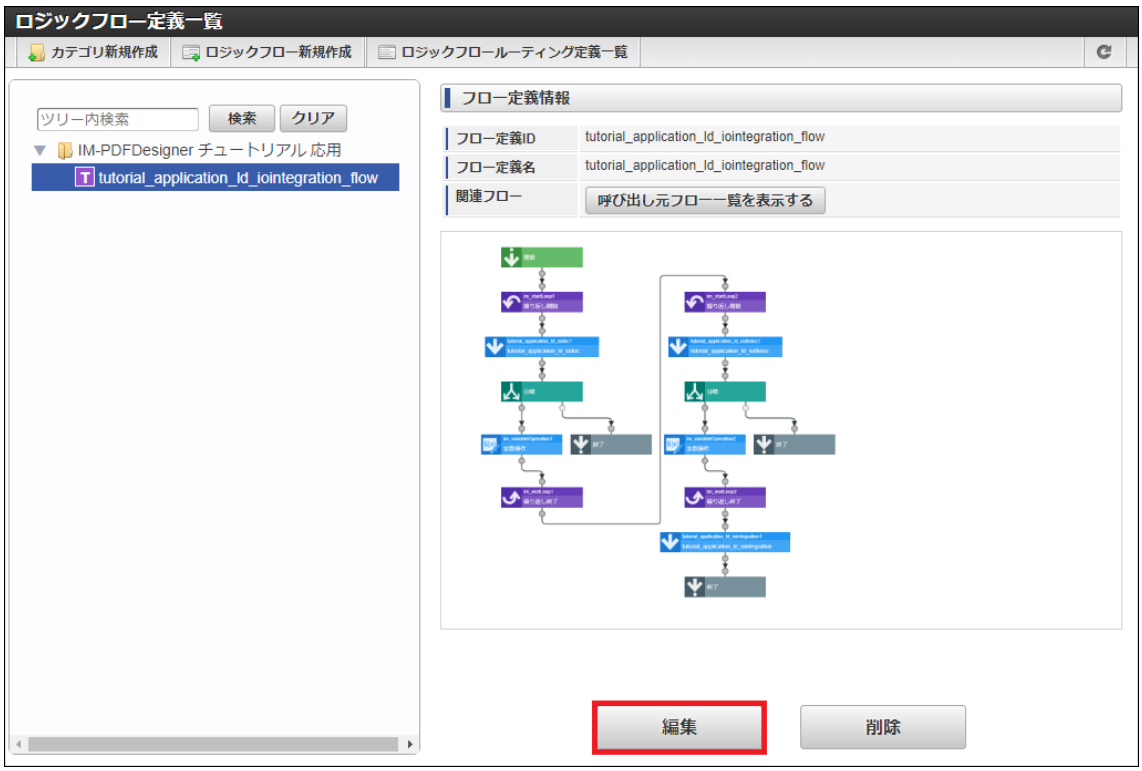
ロジックフロー定義のデバッグ画面を開く

1. 「サイトマップ」-「LogicDesigner」-「フロー定義一覧」をクリックします。

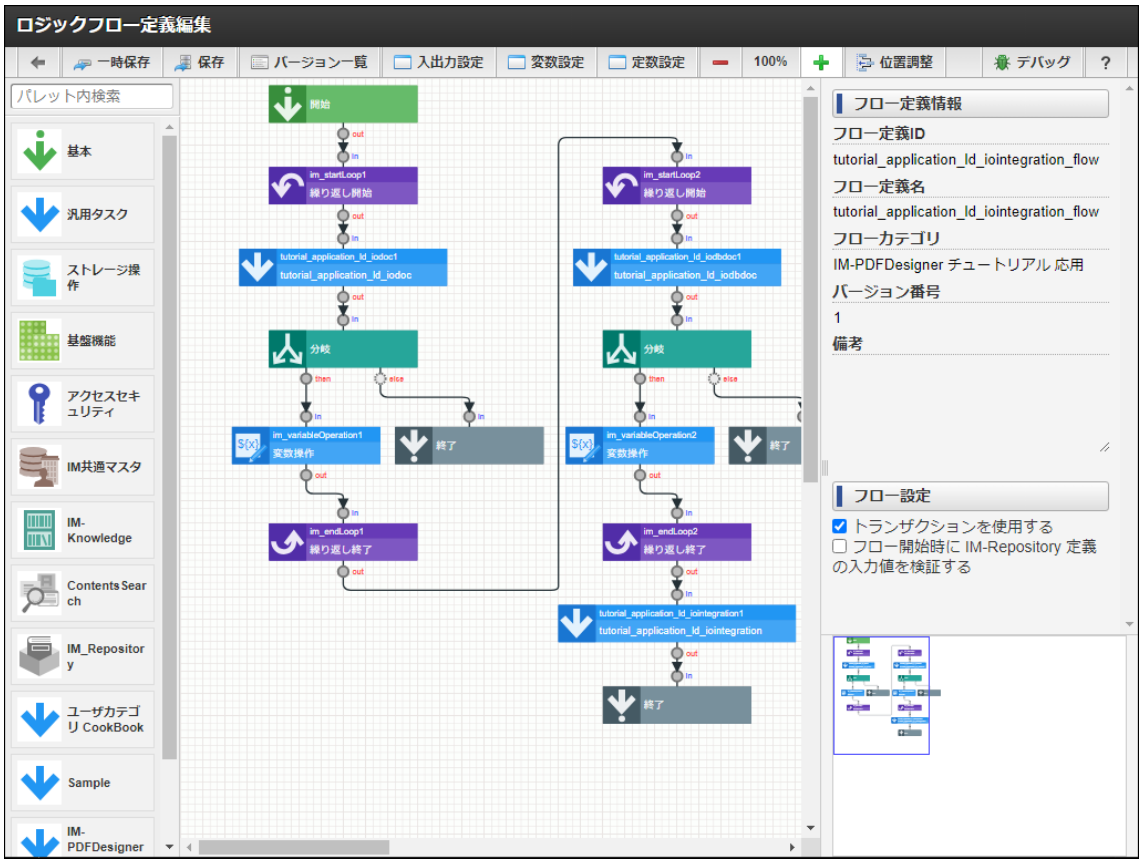


2. 「 IM-PDFDesigner チュートリアル 応用」から次のフロー定義を選択し、「編集」をクリックします。

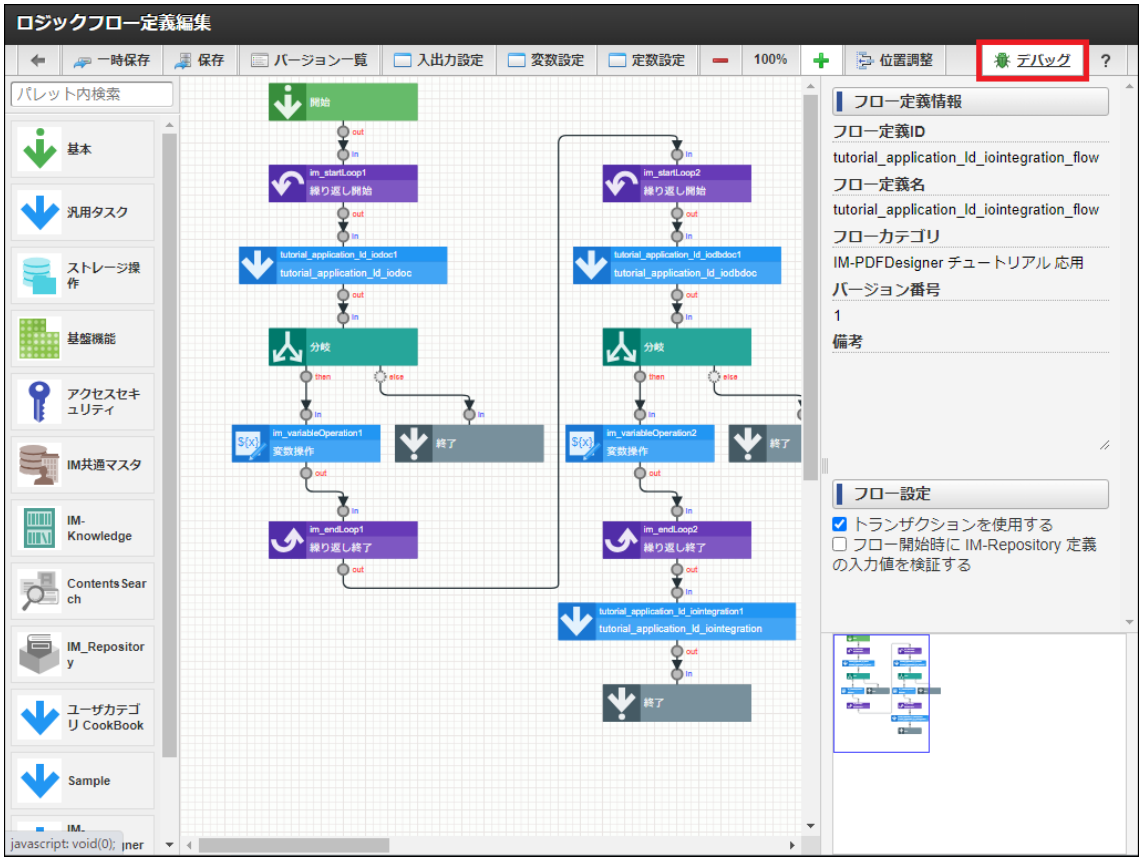
構成	フロー定義
スタンドアローン構成	tutorial_application_ld_iointegration_flow
分散構成	tutorial_application_ld_iointegration_flow_rest



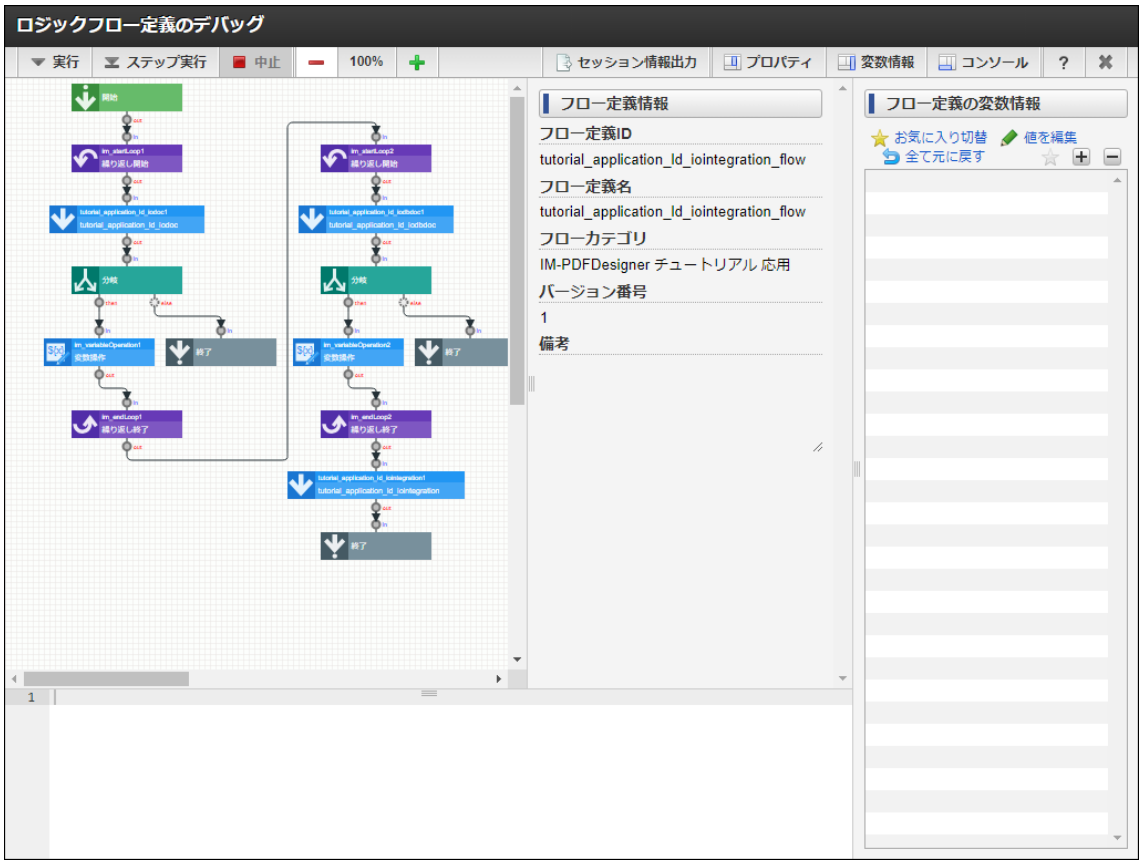
「ロジックフロー定義編集」画面が開きます。



3. 「デバッグ」をクリックします。

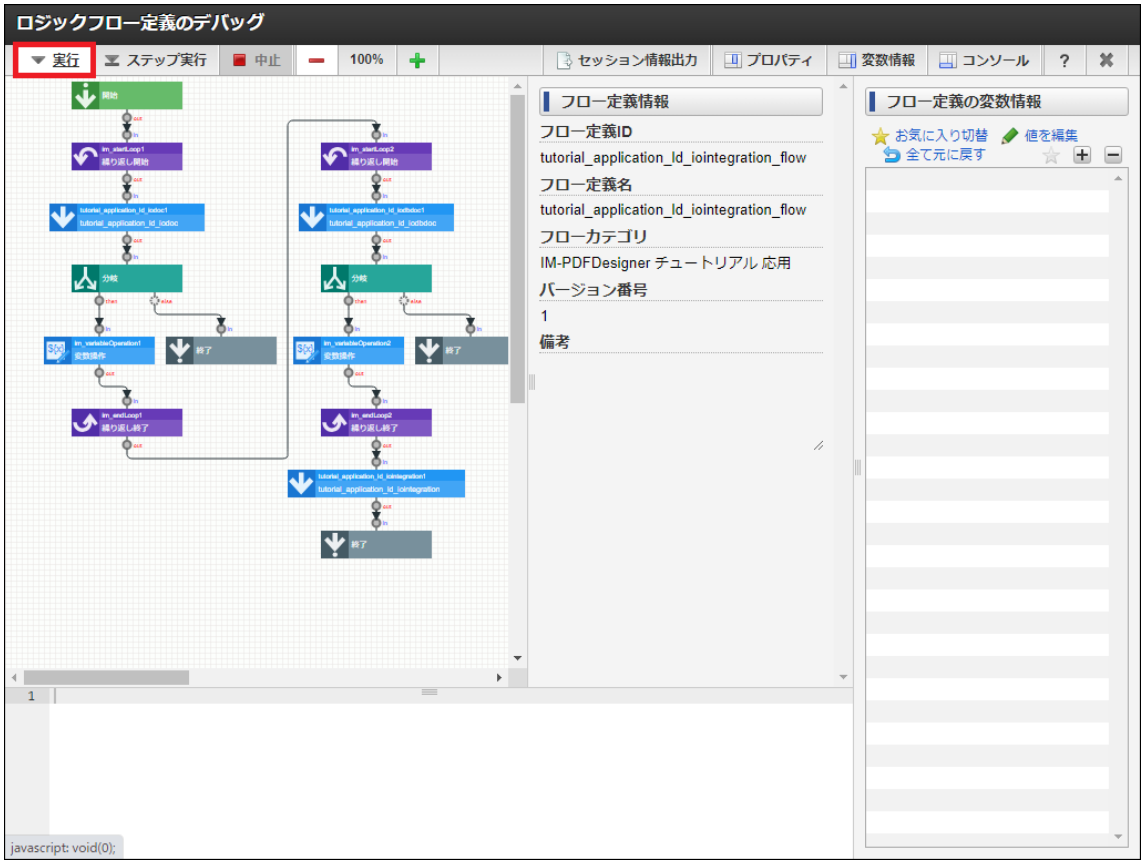


「ロジックフロー定義のデバッグ」画面が開きます。



入力値を設定し、デバッグを実行する

1. 「実行」をクリックします。



2. 各変数に次の値を設定し、「実行」をクリックします。
   フロー定義の入力値の項目は次の通りです。

```
tutorial_application_ld_iointegration_flow／tutorial_application_ld_iointegration_flow_rest <object>
├─ iodoc <object[]>
│    ├─ layoutFilePath <string>
│    ├─ datFilePath <string>
│    └─ outputIodFilePath <string>
├─ iodbdoc <object[]>
│    ├─ layoutFilePath <string>
│    ├─ globalDataFilePath <string>
│    ├─ colDataFilePath <string>
│    └─ outputIodFilePath <string>
└─ outputPdfFilePath <string>
```

項目名	必須/任意	型	配列/リスト	説明
iodoc	iodbdocを指定しない場合は必須	object	配列	ユーザ定義「tutorial_application_ld_iodoc／tutorial_application_ld_iodoc_rest」の各入力値の配列
iodbdoc	iodocを指定しない場合は必須	object	配列	ユーザ定義「tutorial_application_ld_iodbdoc／tutorial_application_ld_iodbdoc_rest」の各入力値の配列
outputPdfFilePath	必須	string	なし	ユーザ定義「tutorial_application_ld_iointegration／tutorial_application_ld_iointegration_rest」の入力値「outputPdfFilePath」

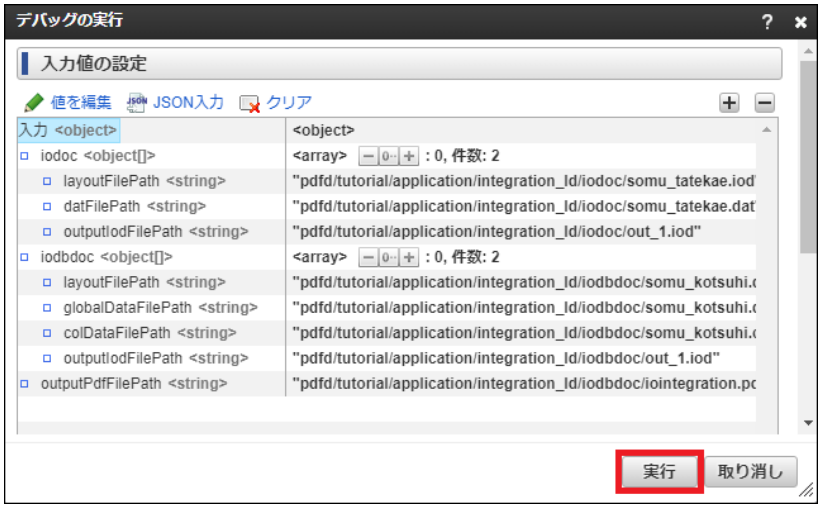
設定値は次の通りです。
< 値を編集 >

変数	値
iodoc	

変数	値
[0] : layoutFilePath	pdfd/tutorial/application/integration_ld/iodoc/somu_tatekae.iod
datFilePath	pdfd/tutorial/application/integration_ld/iodoc/somu_tatekae.dat
outputIodFilePath	pdfd/tutorial/application/integration_ld/iodoc/out_1.iod
[1] : layoutFilePath	pdfd/tutorial/application/integration_ld/iodoc/somu_setubi_ringi.iod
datFilePath	pdfd/tutorial/application/integration_ld/iodoc/somu_setubi_ringi.dat
outputIodFilePath	pdfd/tutorial/application/integration_ld/iodoc/out_2.iod
iodbdoc	
[0] : layoutFilePath	pdfd/tutorial/application/integration_ld/iodbdoc/somu_kotsuhi.ddl
globalDataFilePath	pdfd/tutorial/application/integration_ld/iodbdoc/somu_kotsuhi.dat
colDataFilePath	pdfd/tutorial/application/integration_ld/iodbdoc/somu_kotsuhi.csv
outputIodFilePath	pdfd/tutorial/application/integration_ld/iodbdoc/out_1.iod
[1] : layoutFilePath	pdfd/tutorial/application/integration_ld/iodbdoc/jinji_tukin_kostuhi.ddl
globalDataFilePath	pdfd/tutorial/application/integration_ld/iodbdoc/jinji_tukin_kostuhi.dat
colDataFilePath	pdfd/tutorial/application/integration_ld/iodbdoc/jinji_tukin_kostuhi.csv
outputIodFilePath	pdfd/tutorial/application/integration_ld/iodbdoc/out_2.iod
outputPdfFilePath	pdfd/tutorial/application/integration_ld/iointegration.pdf

< JSON入力 >


```
{
  "iodoc": [
    {
      "layoutFilePath": "pdfd/tutorial/application/integration_ld/iodoc/somu_tatekae.iod",
      "datFilePath": "pdfd/tutorial/application/integration_ld/iodoc/somu_tatekae.dat",
      "outputIodFilePath": "pdfd/tutorial/application/integration_ld/iodoc/out_1.iod"
    },
    {
    "layoutFilePath": "pdfd/tutorial/application/integration_ld/iodoc/somu_setubi_ringi.iod",
    "datFilePath": "pdfd/tutorial/application/integration_ld/iodoc/somu_setubi_ringi.dat",
    "outputIodFilePath": "pdfd/tutorial/application/integration_ld/iodoc/out_2.iod"
    }
  ],
  "iodbdoc": [
    {
      "layoutFilePath": "pdfd/tutorial/application/integration_ld/iodbdoc/somu_kotsuhi.ddl",
      "globalDataFilePath": "pdfd/tutorial/application/integration_ld/iodbdoc/somu_kotsuhi.dat",
      "colDataFilePath": "pdfd/tutorial/application/integration_ld/iodbdoc/somu_kotsuhi.csv",
      "outputIodFilePath": "pdfd/tutorial/application/integration_ld/iodbdoc/out_1.iod"
    },
    {
      "layoutFilePath": "pdfd/tutorial/application/integration_ld/iodbdoc/jinji_tukin_kostuhi.ddl",
      "globalDataFilePath": "pdfd/tutorial/application/integration_ld/iodbdoc/jinji_tukin_kostuhi.dat",
      "colDataFilePath": "pdfd/tutorial/application/integration_ld/iodbdoc/jinji_tukin_kostuhi.csv",
      "outputIodFilePath": "pdfd/tutorial/application/integration_ld/iodbdoc/out_2.iod"
    }
  ],
  "outputPdfFilePath": "pdfd/tutorial/application/integration_ld/iointegration.pdf"
}
```

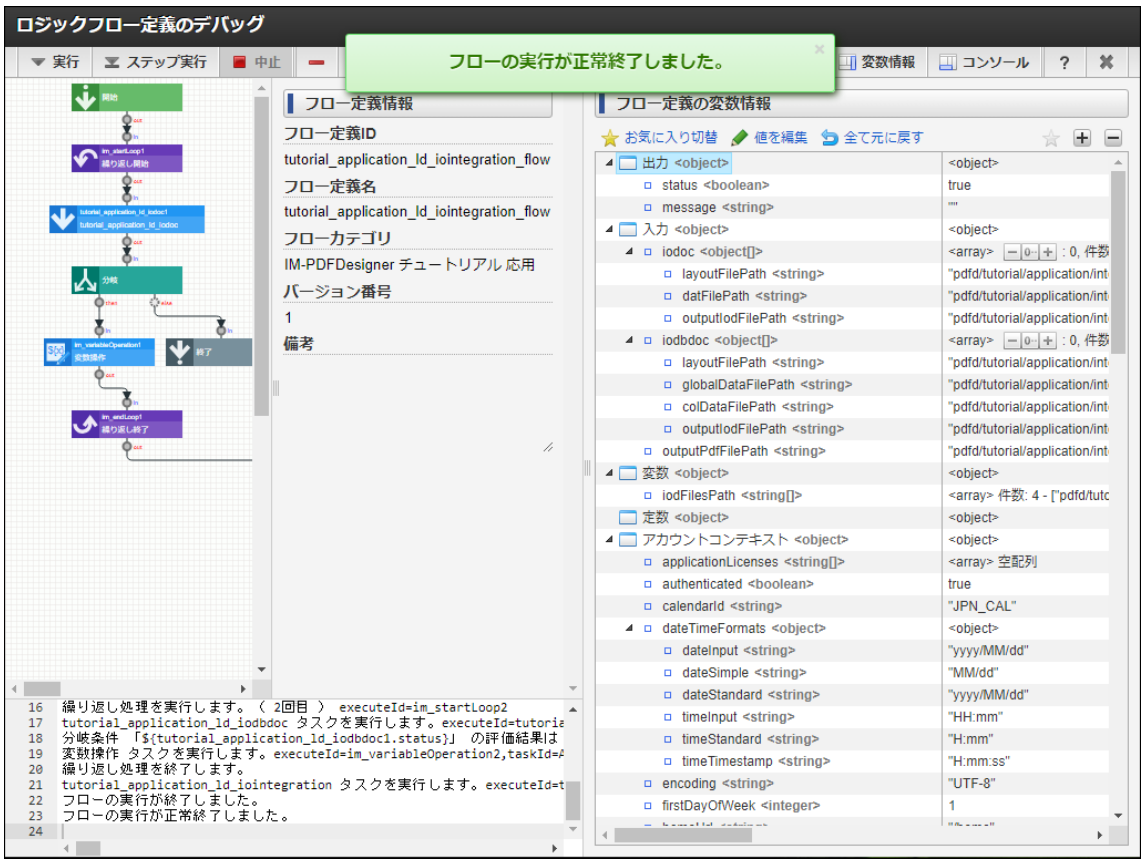


> **コラム**
> ユーザ定義の入出力値については「 *ユーザ定義タスク* 」を参照してください。

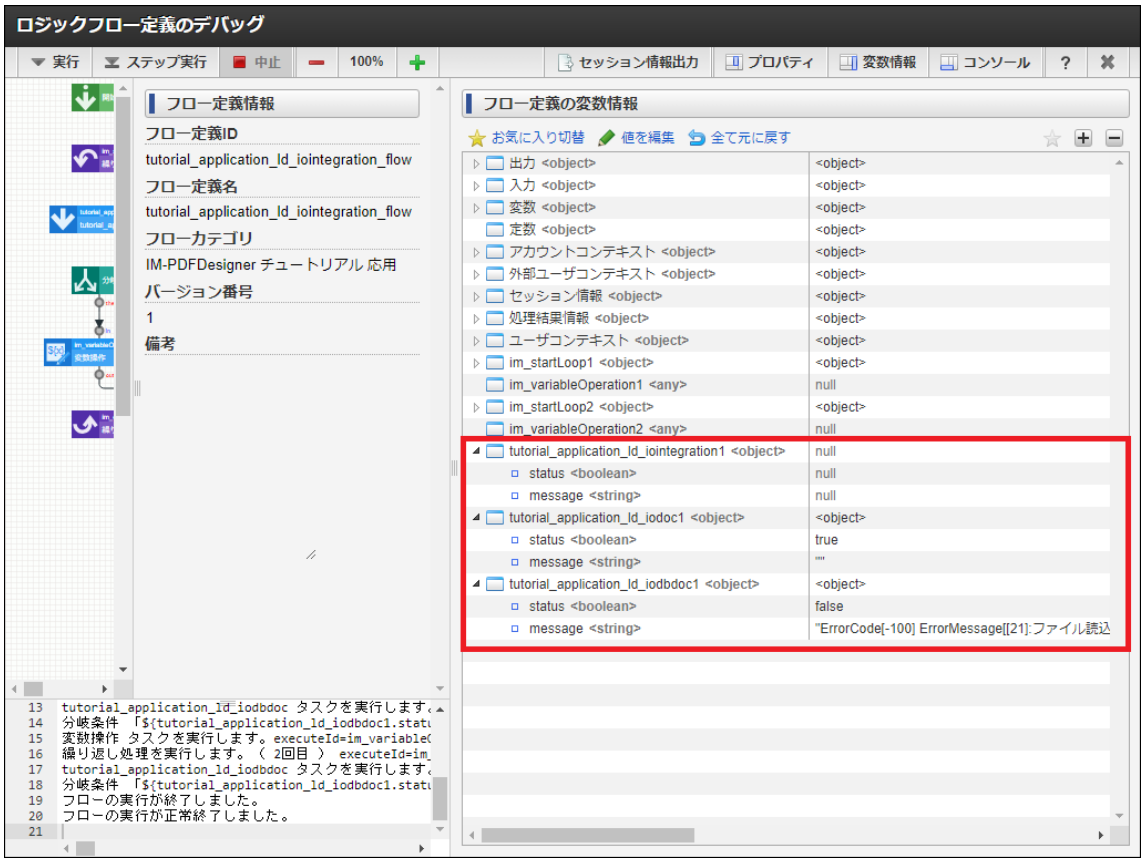
> **コラム**
> 本フロー定義では、次の順番でファイルが結合されます。
>
> 1. 入力値「iodoc」の配列[0]～末尾
> 2. 入力値「iodbdoc」の配列[0]～末尾

3. 「決定」をクリックします。

4. デバッグが開始されます。
   正常にデバッグが終了した場合、その旨のメッセージが表示され、変数情報ペイン、および、コンソールペインが更新されます。

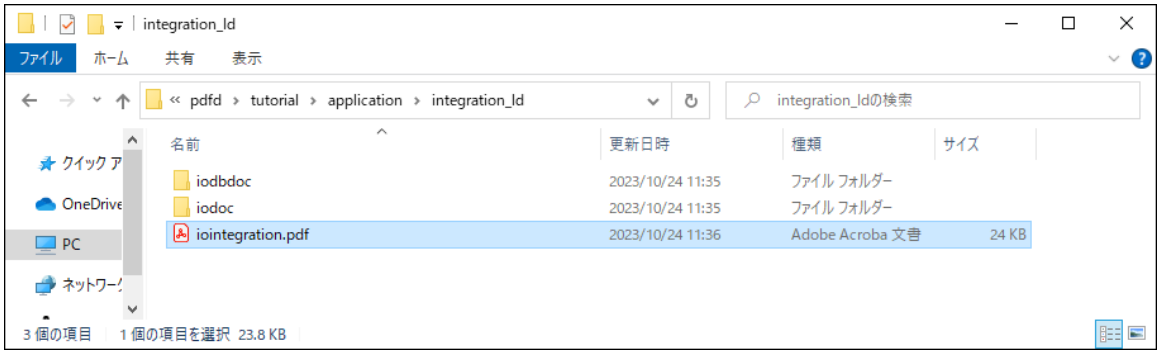


IOD/PDFファイル出力処理に失敗した場合は、デバッグ実行時のユーザ定義の返却値「status」、および、「message」を確認してください。



実行結果を確認する

1. 「outputPdfFilePath」に指定した出力先に、PDFファイルが出力されていることを確認します。



2. 出力されたPDFファイルをPDFビューア（ Adobe Acrobat Reader など）で開き、正しく表示されることを確認します。

本チュートリアルのユーザ定義（ JavaScript定義 ）では、 IM-PDFDesigner for Accel Platform の スクリプト開発モデル 用APIを使用しています。

処理内容を変更したい場合は、ユーザ定義を調整してください。

ユーザ定義の詳細については、次を参照してください。

ユーザ定義タスク

本チュートリアルで使用しているユーザ定義タスクは、次の通りです。

IODファイル出力（単票形式 IODOC ）

単票形式 IODOC を使用し、中間ファイル（IODファイル）を出力するタスクです。

構成	ユーザ定義
スタンドアローン構成	tutorial_application_ld_iodoc
分散構成	tutorial_application_ld_iodoc_rest

編集	ユーザ定義ID	ユーザ定義名	種別	ユーザカテゴリ	呼出元
	tutorial_application_ld_iodoc	tutorial_application_ld_iodoc	javascript	IM-PDFDesigner チュートリアル 応用	
	tutorial_application_ld_iodoc_rest	tutorial_application_ld_iodoc_rest	javascript	IM-PDFDesigner チュートリアル 応用	

入力値

```
tutorial_application_ld_iodoc／tutorial_application_ld_iodoc_rest <object>
├─ layoutFilePath <string>
├─ datFilePath <string>
└─ outputIodFilePath <string>
```

項目名	必須/任意	型	配列/リスト	説明
layoutFilePath	必須	string	なし	IODOC 帳票レイアウトファイルのパブリックストレージパス
datFilePath	必須	string	なし	DAT形式データファイルのパブリックストレージパス
outputIodFilePath	必須	string	なし	出力IODファイルのパブリックストレージパス

出力値

```
tutorial_application_ld_iodoc／tutorial_application_ld_iodoc_rest <object>
├─ status <boolean>
└─ message <string>
```

項目名	型	配列/リスト	説明
status	boolean	なし	true：IODファイル出力成功時 false：IODファイル出力失敗時
message	string	なし	IODファイル出力成功時：空文字 IODファイル出力失敗時：エラーメッセージ

スクリプト

次は、ユーザ定義「tutorial_application_ld_iodoc」のスクリプトです。

コラム

ユーザ定義「tutorial_application_ld_iodoc_rest」のスクリプトの場合、20行目が次に変更となります。

```
const iodoc = new IODocRemote(input.layoutFilePath, "");
```

```
/**
 * run.
 *
 * @param input {Object} - task input data.
 * @return {Object} task result.
 */
function run(input) {

 try {
  if (!input.layoutFilePath) {
   throw new Error("帳票レイアウトファイルパスにnull、または、空文字が指定されています。");
  }
  if (!input.datFilePath) {
   throw new Error("DAT形式データファイルファイルパスにnull、または、空文字が指定されています。");
  }
  if (!input.outputIodFilePath) {
   throw new Error("出力IODファイルパスにnull、または、空文字が指定されています。");
  }

  const iodoc = new IODoc(input.layoutFilePath, "");

  /* DAT形式データファイルを設定 */
  iodoc.setDAT(input.datFilePath);

  /* IOD ファイルを生成 */
  const result = iodoc.toIOD(input.outputIodFilePath);

  if (result >= 0) {
   return {
    status: true,
    message: ""
   };
  } else {
   return {
    status: false,
    message: "ErrorCode[" + result + "] ErrorMessage[" + iodoc.getMessage() + "]"
   };
  }
 } catch (error) {
  return {
   status: false,
   message: error.message
  };
 }
}
```

IODファイル出力（連票形式 IODBDOC ）

連票形式 IODBDOC を使用し、中間ファイル（IODファイル）を出力するタスクです。

構成	ユーザ定義
スタンドアローン構成	tutorial_application_ld_iodbdoc
分散構成	tutorial_application_ld_iodbdoc_rest

編集	ユーザ定義ID	ユーザ定義名	種別	ユーザカテゴリ	呼出元
	tutorial_application_ld_iodbdoc	tutorial_application_ld_iodbdoc	javascript	IM-PDFDesigner チュートリアル 応用	
	tutorial_application_ld_iodbdoc_rest	tutorial_application_ld_iodbdoc_rest	javascript	IM-PDFDesigner チュートリアル 応用	

入力値

```
tutorial_application_ld_iodbdoc／tutorial_application_ld_iodbdoc_rest <object>
├─ layoutFilePath <string>
├─ globalDataFilePath <string>
├─ colDataFilePath <string>
└─ outputIodFilePath <string>
```

項目名	必須/任意	型	配列/リスト	説明
layoutFilePath	必須	string	なし	IODBDOC 帳票レイアウトファイルのパブリックストレージパス
globalDataFilePath	任意	string	なし	外部変数用データファイルのパブリックストレージパス
colDataFilePath	必須	string	なし	行データ用データファイルのパブリックストレージパス
outputIodFilePath	必須	string	なし	出力IODファイルのパブリックストレージパス

出力値

```
tutorial_application_ld_iodbdoc／tutorial_application_ld_iodbdoc_rest <object>
├─ status <boolean>
└─ message <string>
```

項目名	型	配列/リスト	説明
status	boolean	なし	true：IODファイル出力成功時 false：IODファイル出力失敗時
message	string	なし	IODファイル出力成功時：空文字 IODファイル出力失敗時：エラーメッセージ

スクリプト

次は、ユーザ定義「tutorial_application_ld_iodbdoc」のスクリプトです。

コラム

ユーザ定義「tutorial_application_ld_iodbdoc_rest」のスクリプトの場合、14行目が次に変更となります。

```
const iodbdoc = new IODBDocRemote(input.layoutFilePath);
```

```
/**
 * run.
 *
 * @param input {Object} - task input data.
 * @return {Object} task result.
 */
function run(input) {

 try {
  if (!input.outputIodFilePath) {
   throw new Error("出力IODファイルパスにnull、または、空文字が指定されています。");
  }

  const iodbdoc = new IODBDoc(input.layoutFilePath);
  const colData = readDataFile(input.colDataFilePath, ",");

  if (!!input.globalDataFilePath) {
   const globalDataLimit = 1;
   const globalData = readDataFile(input.globalDataFilePath, "=", globalDataLimit);

   // 外部変数値を設定
   globalData.forEach(function (row) {
    if (row.length == 2) {
     iodbdoc.setGlobal(row[0], row[1]);
    }
   });
  }

  // データ行にカラムを追加
  for (let i = 0; i < colData.length; i++) {
   iodbdoc.setColStart();
   for (let j = 0; j < colData[i].length; j++) {
    iodbdoc.setCol(colData[i][j]);
   }
   iodbdoc.setColEnd();
  }

  /* IOD ファイルを生成 */
  const result = iodbdoc.toIOD(input.outputIodFilePath);

  if (result >= 0) {
   return {
    status: true,
    message: ""
   };
  } else {
   return {
    status: false,
    message: "ErrorCode[" + result + "] ErrorMessage[" + iodbdoc.getMessage() + "]"
   };
  }
 } catch (error) {
  return {
   status: false,
   message: error.message
  };
 }
}

/**
 * readDataFile
 * 指定したデータファイルを読み込み、データを返します。
 * @param dataFilePath {string} - 対象データファイルのパブリックストレージパス
 * @param delimiter {string} - 区切り文字
 * @param limit {number} - 分割される文字列の数
 * @return {Object} 対象データファイルのデータ群
 */
function readDataFile(dataFilePath, delimiter, limit) {
 if (!dataFilePath) {
```

```
    throw new Error("読込対象データファイルパスにnull、または、空文字が指定されています。");
  }

  const storage = new PublicStorage(dataFilePath);
  let data = [];

  // データファイルを読み込む
  storage.openAsText(function(reader, error) {
    if (error !== null) {
      throw error;
    }

    let i = 0;

    // 1行ずつ読み込み、データを作成
    reader.eachLine(function (line) {
      if (i === 0) {
        line = line.replace(/^\ufeff/, "");
      }
      if (limit === 1) {
        let temp = [];
        temp = line.split(delimiter, limit);
        temp[1] = line.replace((temp + delimiter), "");
        data[i] = temp;
      } else {
        data[i] = line.split(delimiter);
      }
      i++;
    });
  });

  return data;
}
```

PDFファイル出力（IODファイル結合）

IOIntegration クラスを使用し、中間ファイル（IODファイル）を結合後、PDFファイルを出力するタスクです。

構成	ユーザ定義
スタンドアローン構成	tutorial_application_ld_iointegration
分散構成	tutorial_application_ld_iointegration_rest

編集	ユーザ定義ID	ユーザ定義名	種別	ユーザカテゴリ	呼出元
	tutorial_application_ld_iointegration	tutorial_application_ld_iointegration	javascript	IM-PDFDesigner チュートリアル 応用	
	tutorial_application_ld_iointegration_rest	tutorial_application_ld_iointegration_rest	javascript	IM-PDFDesigner チュートリアル 応用	

入力値

```
tutorial_application_ld_iointegration／tutorial_application_ld_iointegration_rest <object>
├─ iodFilesPath <string[]>
└─ outputPdfFilePath <string>
```

項目名	必須/任意	型	配列/リスト	説明
iodFilesPath	必須	string	配列	結合用IODファイルのパブリックストレージパス群
outputPdfFilePath	必須	string	なし	出力PDFファイルのパブリックストレージパス

出力値

```
tutorial_application_ld_iointegration／tutorial_application_ld_iointegration_rest <object>
├─ status <boolean>
└─ message <string>
```

項目名	型	配列/リスト	説明
status	boolean	なし	true：PDFファイル出力成功時 false：PDFファイル出力失敗時
message	string	なし	PDFファイル出力成功時：空文字 PDFファイル出力失敗時：エラーメッセージ

スクリプト

次は、ユーザ定義「tutorial_application_ld_iointegration」のスクリプトです。

コラム

ユーザ定義「tutorial_application_ld_iointegration_rest」のスクリプトの場合、17行目が次に変更となります。

```
const iointegration = new IOIntegrationRemote();
```

コラム

文書情報を設定する場合は、スクリプトの25、28行目のコメントを外してください。

コラム

セキュリティ情報を設定する場合は、スクリプトの31、35行目のコメントを外してください。

```
/**
 * run.
 *
 * @param input {Object} - task input data.
 * @return {Object} task result.
 */
function run(input) {

 try {
  if (!input.iodFilesPath) {
   throw new Error("IODファイルパス群にnullが指定されています。");
  }
  if (!input.outputPdfFilePath) {
   throw new Error("出力PDFファイルパスにnull、または、空文字が指定されています。");
  }

  const iointegration = new IOIntegration();

  /* 結合するIODファイルを追加 */
  input.iodFilesPath.forEach(function(iodFilePath) {
   iointegration.add(iodFilePath);
  });

  /* 文書情報を設定 */
  /* iointegration.defineTitle("タイトル");
  iointegration.defineSubTitle("サブタイトル");
  iointegration.defineAuthor("作成者");
  iointegration.defineApplication("アプリケーション"); */

  /* セキュリティ情報を設定 */
  /* iointegration.setOpenPassword("open");
  iointegration.setSecurityPassword("sec");
  iointegration.printSecurity("PRINT_DISABLE");
  iointegration.modifySecurity("MODIFY_DISABLE");
  iointegration.copySecurity("COPY_AND_ACCESSBILITY_DISABLE"); */

  /* PDFファイルを生成 */
  const result = iointegration.toPDF(input.outputPdfFilePath);

  if (result >= 0) {
   return {
    status: true,
    message: ""
   };
  } else {
   return {
    status: false,
    message: "ErrorCode[" + result + "] ErrorMessage[" + iointegration.getMessage() + "]"
   };
  }
 } catch (error) {
  return {
   status: false,
   message: error.message
  };
 }
}
```

ステータスコード表

次のAPIで返却される戻り値の一覧です。

- PDFファイルを出力するAPI
- IODファイルを出力するAPI

正常終了時の具体的なステータスコードについては「IM-PDFDesigner for Accel Platform API ドキュメント」を参照してください。

ステータスコード	内容
0以上	正常終了
-1	MS-DOSのInt21ファンクションコール4B00が無効
-2	実行ファイルが見つからない
-3	パスが見つからない
-4	ファイルオープン数エラー
-5	ダイナミックリンクライブラリ実行エラー
-6	データセグメントエラー
-7	OSのメモリエラー
-8	システムエラー
-9	OSのメモリエラー
-10	現在実行中のOSには未対応
-11	■ 実行に必要なファイルが壊れている ■ 関数パラメータ不正
-12	ランタイムのプラットフォームエラー
-13	ランタイムのプラットフォームエラー
-14	ファイルタイプエラー
-15	実行ファイルのバージョンエラー
-16	実行ファイルのロードエラー
-17	DLLのロードエラー
-18	アプリケーションのロードエラー
-19	実行ファイルのロードエラー
-20	実行に必要なファイルが壊れている
-21	■ 現在実行中のOSには未対応 ■ 帳票エンジン/帳票レイアウトファイルの配置パスが128bytesを超えている可能性があります
-22	テンポラリファイル作成失敗
-23	■ システムエラー ■ データが不正
-24	システムエラー
-25	システムエラー
-26	未定義のエラー
-27	未定義のエラー
-28	未定義のエラー

ステータスコード	内容
-29	未定義のエラー
-30	未定義のエラー
-31	未定義のエラー
-32	未定義のエラー
-33	システムエラー
-100	ファイルアクセスエラー
-101	パラメータエラー
-102	メモリエラー
-103	ランタイムモジュールの起動エラー ▪ 帳票レイアウトの保存バージョンと、サーバ上の帳票出力エンジンのバージョンが一致していない可能性があります。 ▪ 帳票レイアウトの保存バージョンが 3.x で、サーバ上の帳票出力エンジンのバージョンが 1.9.x である場合などが該当します。帳票レイアウト保存時のバージョンを、サーバ上の帳票出力エンジンのバージョンと一致させてください。
-104	IOWebDOCのセットアップエラー
-105	IOWebDOCのライセンスエラー
-106	印刷中のエラー
-107	直接印刷中のキャンセル
-200	セキュリティエラー(パスワードが不正等)
-515	データカラム数が多い
-516	データカラム数が少ない
-999	その他のエラー
-1001	レイアウトファイルのパスが未定義
-1002	レイアウトファイルが存在しない
-1003	変換定義ファイル(cdd)のパスが未定義
-1004	変換定義ファイル(cdd)が存在しない
-1005	データファイルのパスが未定義
-1006	データファイルが存在しない
-1007	データが設定されていない
-1008	出力先PDFファイルのパスが未定義
-1009	出力先IODファイルのパスが未定義
-1010	データファイルのロードに失敗
-1011	IODOCラインタイム実行時エラー
-1012	IOWebDOC Java-Interfaceライセンスエラー
-1020	オープンパスワードとセキュリティパスワードに同じパスワードを設定している
-1021	PDFファイルのセキュリティパスワードが未設定
-1022	PDFファイルのセキュリティ情報が未設定(印刷可否、編集可否等)
-2000	▪ サービスのURLが未設定 ▪ 変換サービスの実行に失敗

ステータスコード	内容
-2001	変換サービスの実行に失敗
-2002	▪ 変換サービスの実行に失敗 ▪ 接続タイムアウトが発生
-2003	変換サービスの実行に失敗
-2004	タイムアウトに負の値を設定している
-2005	受信ファイル展開中のエラー
-2006	受信ファイル内に出力IODファイルが存在しない
-2007	受信ファイル内に出力PDFファイルが存在しない
-2008	受信ファイル内にログファイルが存在しない
-3000	結果テキストの作成に失敗
-3001	WEBCELAの内部処理のエラー
-3002	IODBDOCの内部処理のエラー
-3003	圧縮ファイルの作成に失敗
その他	その他のエラー

処理を正常終了できなかった場合は、メッセージ取得メソッドから、返却されたエラーコードに対応するエラーメッセージを取得できます。

注意

-2000番台、および、-3000番台のエラーについては、Restサービス側で問題が発生しています。

PDF帳票サーバ の、 Apache Tomcat のログを併せて確認してください。

サポート

Webにて当製品に対するサポート、および、技術情報を公開しています。

当製品に関して不明な点などがある場合、情報検索、または、「 intra-mart サポートサイト 」に問い合わせしてください。

intra-mart e Builder for Accel Platform との連携方法

intra-mart e Builder for Accel Platform と連携して、 IM-PDFDesigner for Accel Platform を開発することができます。

本セットアップ以外の intra-mart e Builder for Accel Platform に関するセットアップ手順、および、アプリケーション開発手順については、次を参照してください。

- 「intra-mart e Builder for Accel Platform セットアップガイド」
- 「intra-mart e Builder for Accel Platform アプリケーション開発ガイド」

システム要件

利用する intra-mart e Builder for Accel Platform のシステム要件 に準じます。

セットアップ

セットアップ手順は、利用する環境に合わせて適宜読み替えて作業してください。

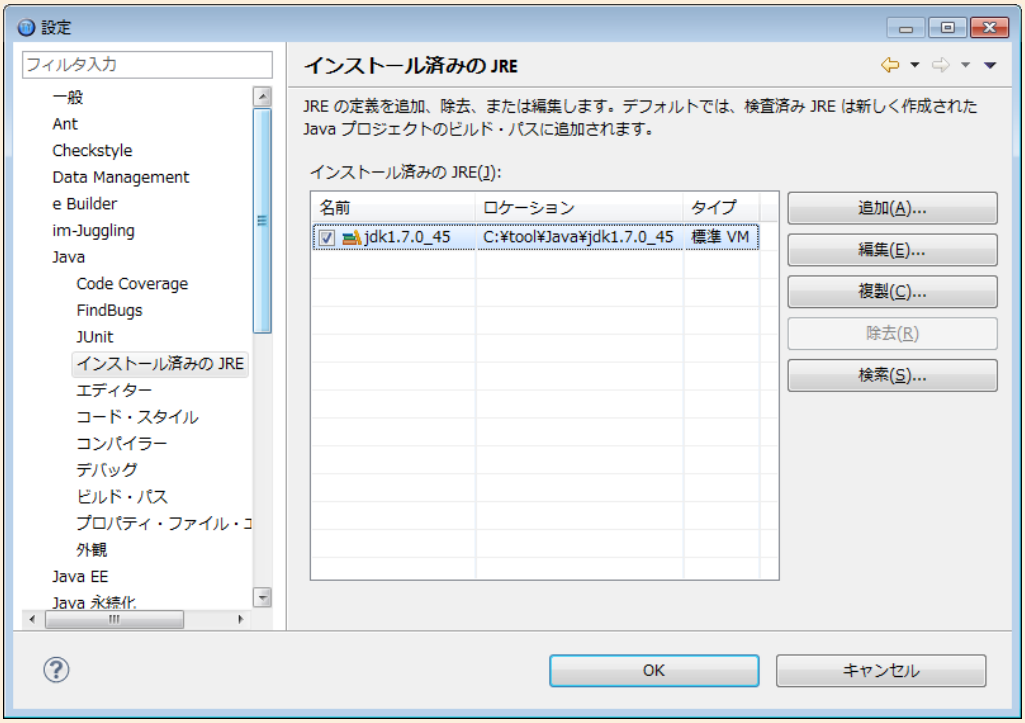
1. 「intra-mart e Builder for Accel Platform セットアップガイド」-「e Builder のインストール 」に従って、 intra-mart e Builder for Accel Platform をインストールします。
2. 「intra-mart e Builder for Accel Platform セットアップガイド」-「e Builder 起動後の設定 」に従って、 intra-mart e Builder for Accel Platform の起動後の設定を行います。
3. 「intra-mart e Builder for Accel Platform セットアップガイド」-「デバッグサーバの構築 」、および、「IM-PDFDesigner for Accel Platform セットアップガイド 」に従って、上記手順で構築した intra-mart e Builder for Accel Platform 環境に IM-PDFDesigner for Accel Platform をセットアップします。

注意

intra-mart e Builder for Accel Platform は、OSに設定されている環境変数を認識しない場合があり、UnsatisfiedLinkErrorのエラーが発生することがあります。

エラーが発生した場合は、次の対応を行ってください。

1. intra-mart e Builder for Accel Platform を起動します。
2. メニューバーから「ウィンドウ」-「設定」を選択します。
3. 設定画面から「Java」-「インストール済みのJRE」を選択します。

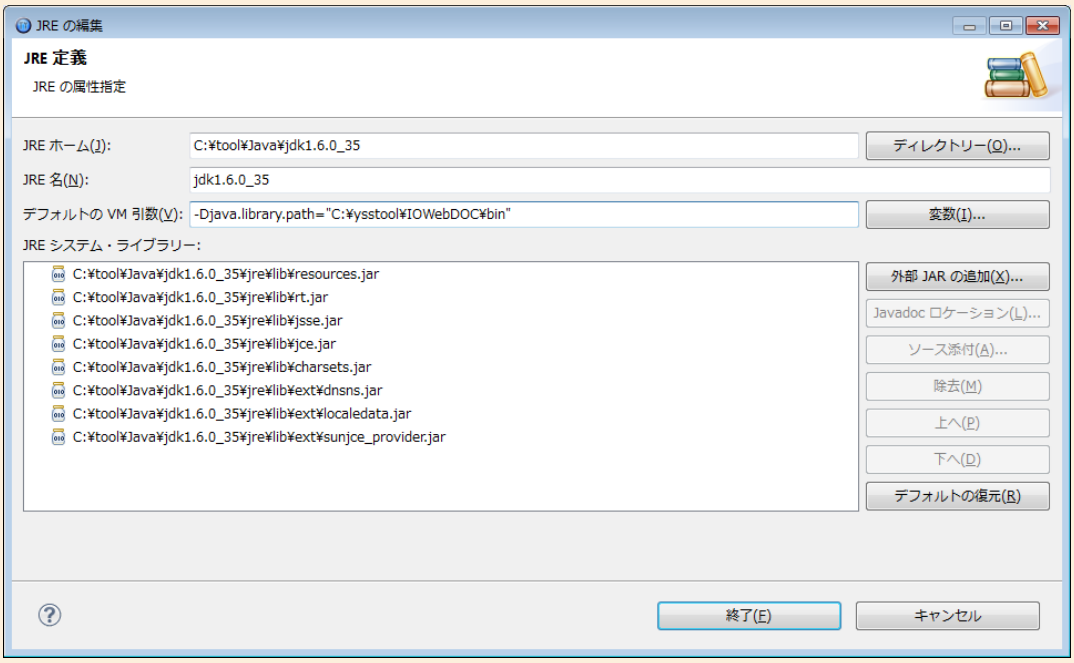


4. 利用しているJDKを選択して、「編集」をクリックします。
5. 「デフォルトのVM引数」に、次のように< %IOWEBDOC_HOME% /bin >を指定します。

```
-Djava.library.path=”C:\tool\IOWebDOC\bin”
```

> **注意**
>
> VMの引数が複数の場所で設定されている場合、優先順位の高い設定のみが有効になります。
>
> その場合には、優先順位の高い設定に上記の引数を追加してください。

6. 「終了」をクリックします。



上記の対応を行っても、同様のエラーが発生する場合は、次の対応を行ってください。

1. < %RESIN_HOME%/win64 >に、< %IOWEBDOC_HOME% /bin/iowebjav.dll >をコピーします。
2. Resin 、および、 intra-mart e Builder for Accel Platform を再起動します。

項目

- グループ化機能
- 文字サイズの自動縮小機能
- ページ自動振り分け機能

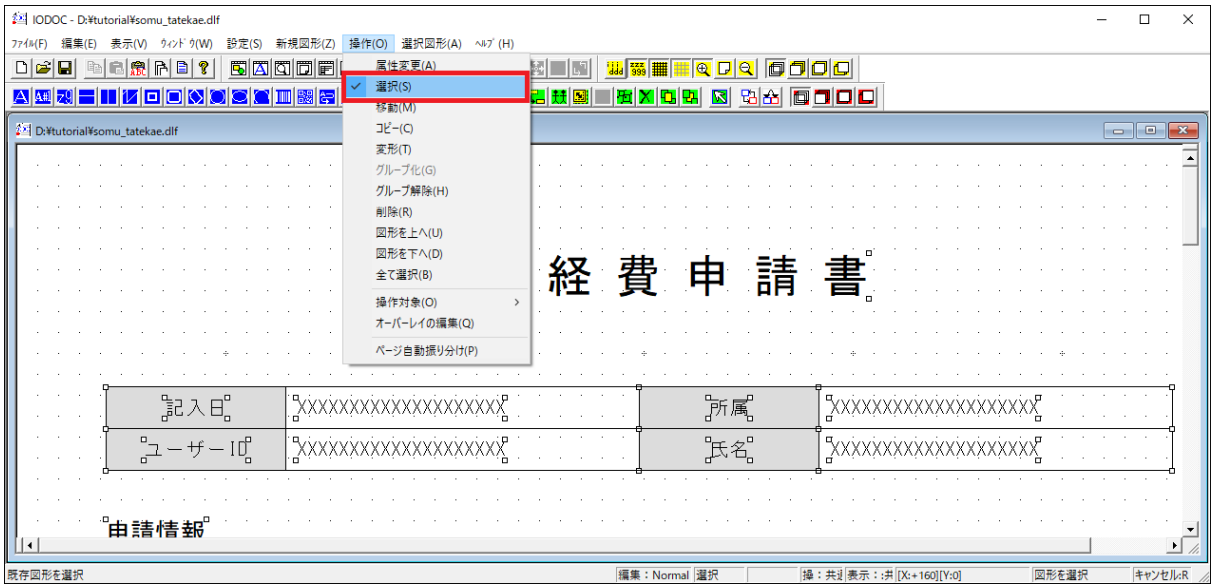
グループ化機能

グループ化機能とは、複数のオブジェクトを束ねて、1つのオブジェクトとして扱うことができる機能です。

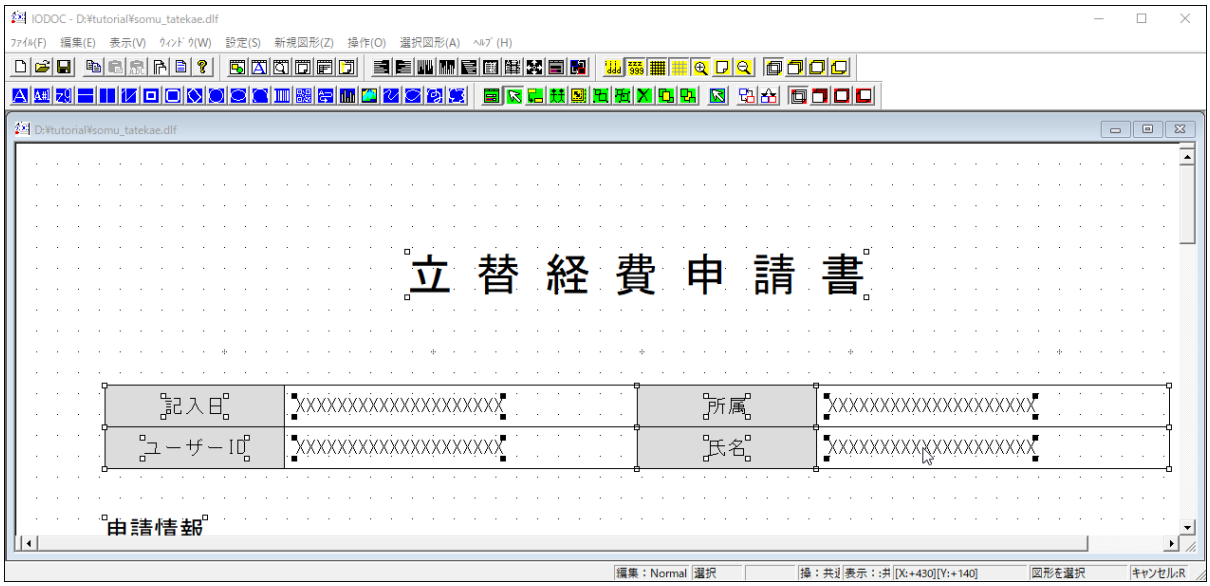
上記機能を使用することで、複数のオブジェクトをまとまった単位でコピー、削除、および、移動などの操作を行うことが可能です。

機能の使用手順は、次の通りです。

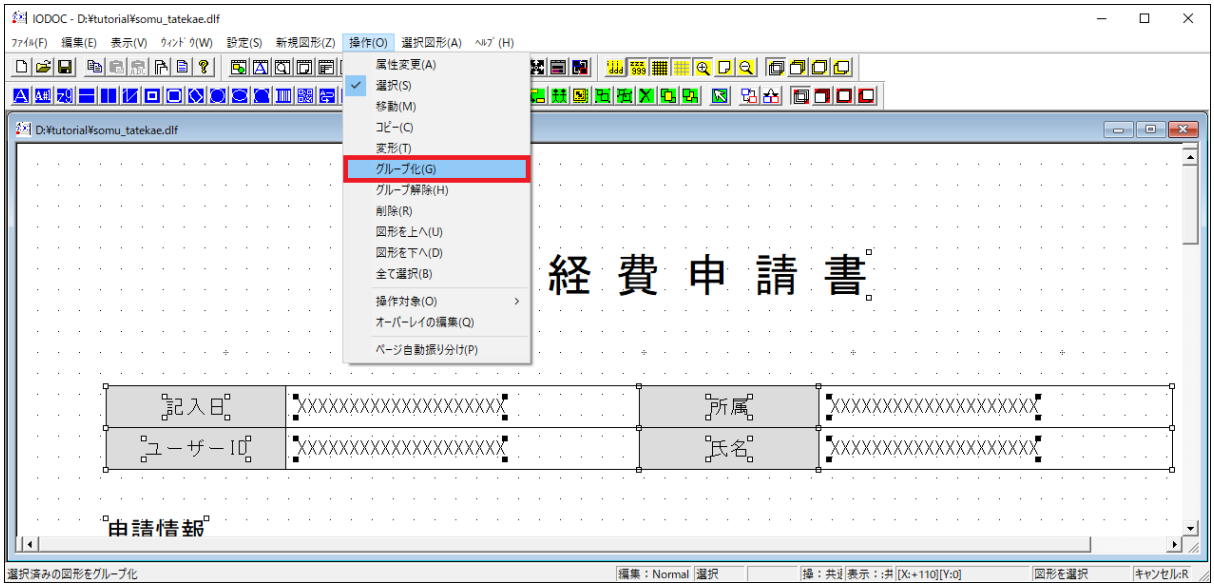
1. メニューバーから「操作」-「選択」を選択します。



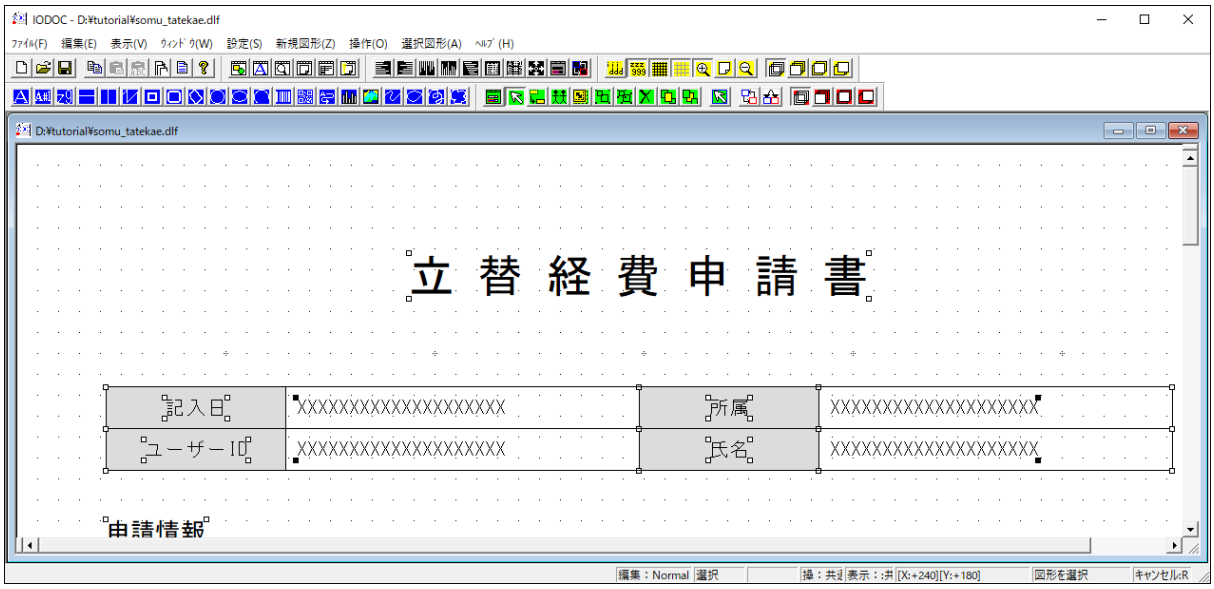
2. グループ化するオブジェクトをクリックします。
   オブジェクトの四隅の点が、白色から黒色に変化します。



3. メニューバーから「操作」-「グループ化」を選択します。



選択したオブジェクトを1つのオブジェクトとして囲うように、四隅の点の表示が変化します。



文字サイズの自動縮小機能

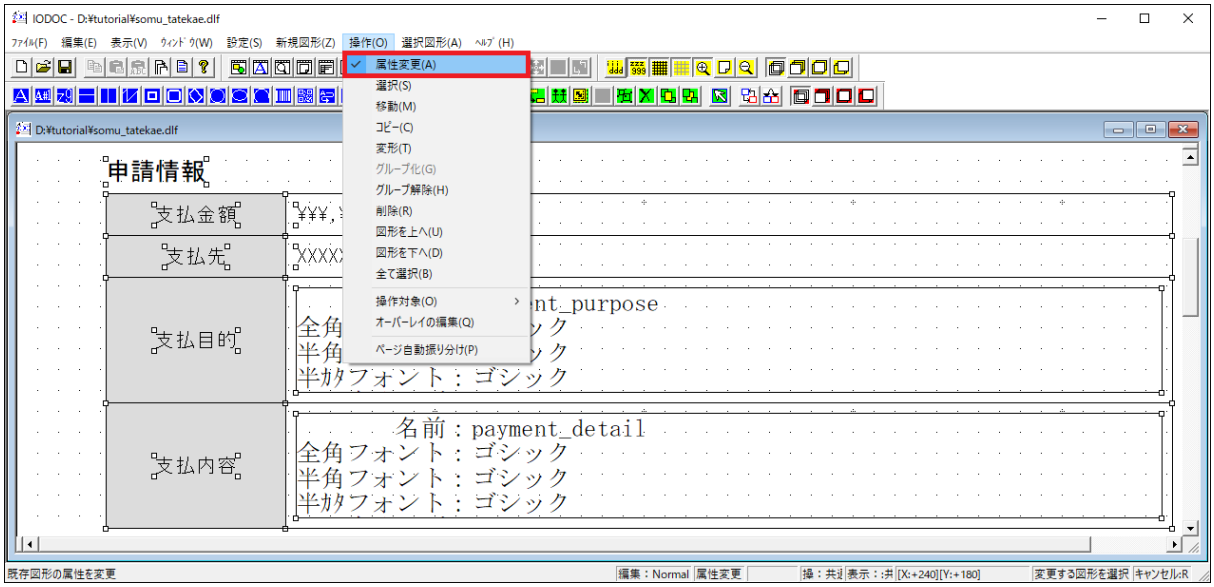
文字サイズの自動縮小機能とは、文字枠のオブジェクトにおいて、文字枠のサイズと文字数に合わせて、文字サイズを自動縮小する機能です。

上記機能を使用することで、文字枠内に収まらない文字数が渡された場合、文字枠内に収まるように文字サイズが自動で縮小されます。

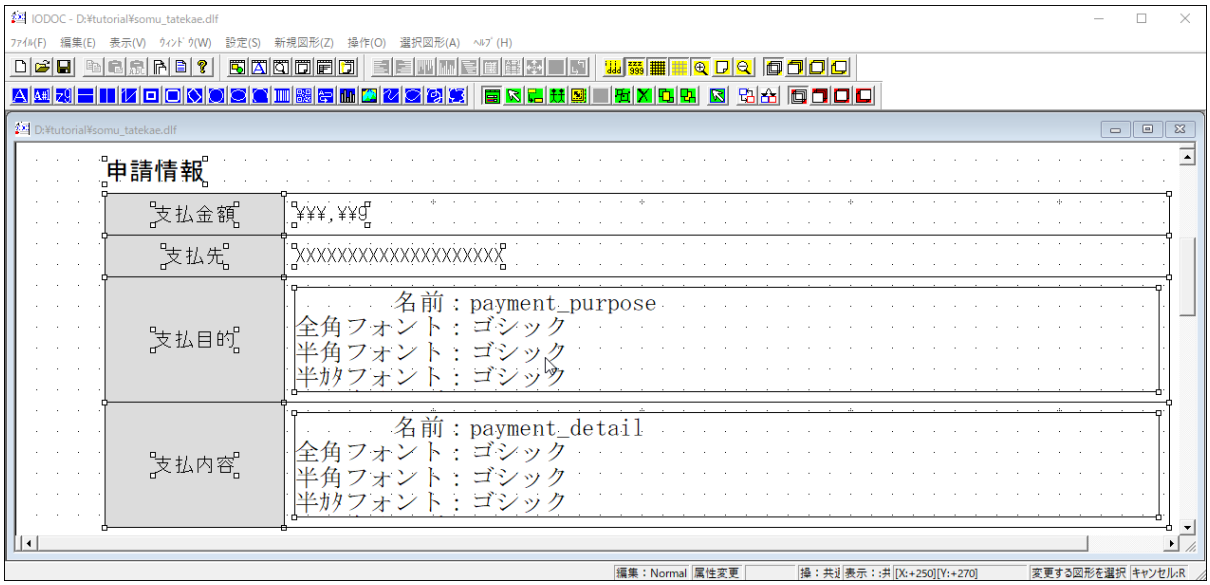
> **注意**
>
> 上記機能を使用する際、文字枠に渡す文字数に制限があります。
>
> 詳細については、「IM-PDFDesigner for Accel Platform リリースノート」-「文字枠の文字サイズを自動縮小する機能使用時は、入力できる文字数に制限があります」を参照してください。

機能の使用手順は、次の通りです。

1. メニューバーから「操作」-「属性変更」を選択します。



2. 文字サイズの自動縮小機能を有効にする文字枠をクリックします。



3. 「属性設定」画面-「文字枠」タブの、「文字サイズを自動縮小」にチェックを入れ、「OK」をクリックします。

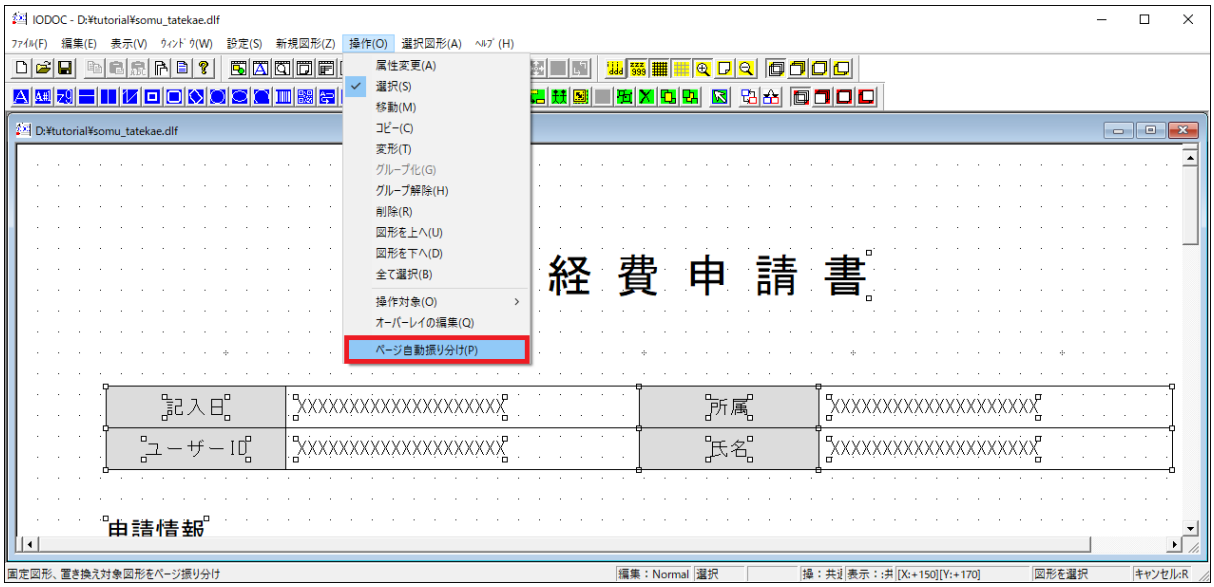
ページ自動振り分け機能

ページ自動振り分け機能とは、固定図形、置き換え図形などのオブジェクトを、通常ページとオーバーレイページに自動で振り分ける機能です。

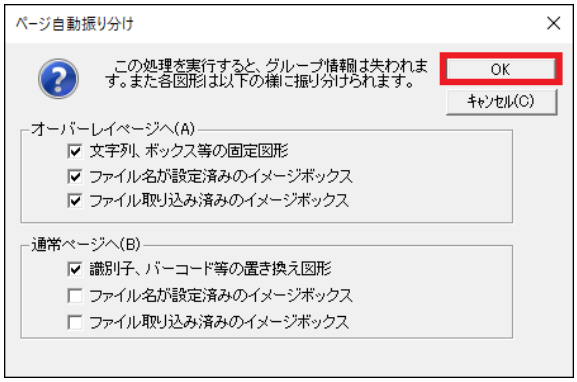
オーバーレイとは、同一のレイアウトを複数ページ作成する場合において、 固定のレイアウト部分の情報を1ページ分だけ保持することで、PDFファイルのファイルサイズの縮小や処理速度を向上させるための機能です。

機能の使用手順は、次の通りです。

1. メニューバーから「操作」-「ページ自動振り分け」を選択します。

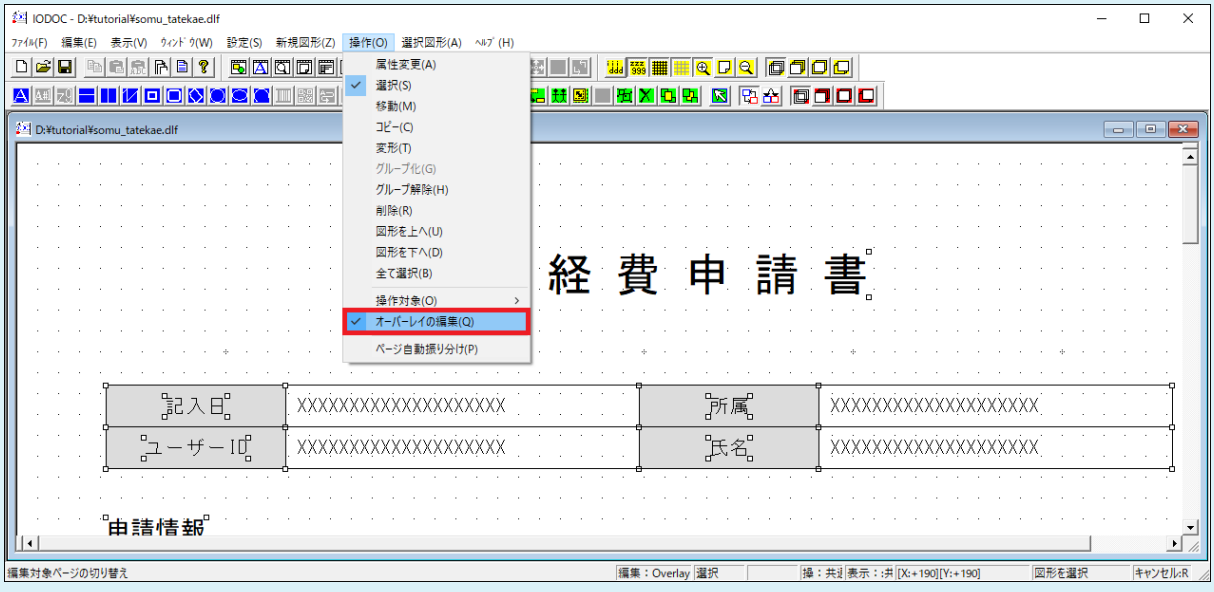


2. 「OK」をクリックします。



コラム

オーバーレイページに移動したオブジェクトを編集する場合、メニューバーから「操作」-「オーバーレイの編集」を選択します。



PDFファイルへの印影付与について

IM-PDFDesigner for Accel Platform では、印影を付与したようなPDFファイルを作成することが可能です。

作成方法は、次の通りです。

- 画像の付与

  印影の画像データを埋め込む方法です。

  社外向けの認印として社外文書に押印する会社印や社印など、情報が固定している印影を付与する場合に使用します。

- 文字や日付などの付与

  印影を模して文字や日付などの動的なデータを埋め込みます。

  社内向けの承認印として社内文書に押印する「部署/日付/氏名」が入った三段印など、情報が動的な印影を付与する場合に使用します。

  > **注意**
  > 動的なデータのみを埋め込むため、枠線がない状態の印影を付与したような見た目になります。

各データが埋め込まれるタイミングは、PDFファイル作成時のみです。

> **注意**
> 外部から受け取ったPDFファイルなど、作成後のPDFファイルに対して後からデータを埋め込むことはできません。
>
> 作成後のPDFファイルにデータを埋め込む場合は、 IM-PDFCoordinator for Accel Platform を利用してください。

画像の付与

帳票デザインツール IODOC 、および、 IODBDOC で画像の項目を設定します。

設定した画像の項目に、印影の画像データを渡すことで、印影が表示されます。

> **注意**
> 表示される画像は透過しないため、画像の項目の背面に他の項目を設定した場合、画像が重なり、背面の項目が隠れます。
>
> そのため、画像の項目が背面になるよう重なり順を調整してください。

> **注意**
> 帳票レイアウトファイルに画像の項目を設定していても、画像データを渡さない場合、画像は埋め込まれず、PDFファイル上に印影は表示されません。
>
> 印影の表示・非表示を切り替える場合は、画像データの引き渡しを上位プログラムで制御してください。

文字や日付などの付与

帳票デザインツール IODOC 、および、 IODBDOC で、付与する印影の情報となる氏名、日付などの項目を設定します。

設定した項目にデータを渡すことで、枠線がない状態の印影を模したものが表示されます。

> 注意
> 枠線がある状態の印影を表示する場合は、 IM-PDFCoordinator for Accel Platform を利用してください。

「^」の扱いについて

単票形式 IODOC の文字枠に埋め込むデータとして、改行など通常の文字では表せない特殊な文字をCSVファイルで指定する際、制御文字として「^」を使用します。

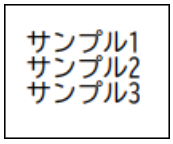
「^」 のあとに、ASCIIコードの16進数を指定することで、特殊な文字を表現します。

使用例は、次の通りです。

- 改行する場合
  - CSVファイル

    ```
    サンプル1^0Aサンプル2^0Aサンプル3
    ```

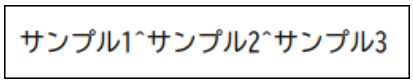
  - PDFファイル

- 「^」 を表示する場合
  - CSVファイル

    ```
    サンプル1^5Eサンプル2^5Eサンプル3
    ```

  - PDFファイル

> 注意
> 下位バージョンとの互換性維持のため、「^0D」による改行が可能となっていますが、通常は「^0A」を使用してください。

IM-PDFDesigner for Accel Platform で廃止されたAPI

IM-PDFDesigner for Accel Platform では、 IM-PDF デザイナー Ver.7.x の次のクラス、メソッドが廃止されています。

廃止されたクラス、メソッドを利用している場合は、 IM-PDF デザイナー Ver.7.x から移行する際に、ソースコードの修正が必要です。

廃止クラス

IM-PDFDesigner for Accel Platform で廃止されたクラスは、次の通りです。

クラス名	移行先
AbstractBuilder	なし
AbstractPageBuilder	なし
IOCelaPageBuilder	CSVCela

クラス名	移行先
IOCelaPageWriter	CSVCela
IODocPageBuilder	CSVDoc
IODocPageWriter	CSVDoc
PageWriter	なし
PDFBuilder	IOIntegration
PDFWriter	IOIntegration

IM-PDFDesigner for Accel Platform で利用可能なクラスについては、「IM-PDFDesigner for Accel Platform API ドキュメント」を参照してください。

廃止メソッド

IM-PDFDesigner for Accel Platform で廃止されたメソッドは、次の通りです。

メソッド名	移行先
setCompression	なし

IM-PDFDesigner for Accel Platform で利用可能なメソッドについては、「IM-PDFDesigner for Accel Platform API ドキュメント」を参照してください。

連票形式 IOCELA の出力カスタマイズ

連票形式 IOCELA は、設定ファイルで出力を制御することが可能です。

設定した内容は、 連携エンジン IOWebDOC をインストールしたサーバ全体に影響します。

> **コラム**
> 作成済みのPDFファイルは影響を受けません。

> **注意**
> 設定はサーバ単位で有効になります。帳票毎に設定を切り替えることはできません。

項目

- 設定手順
- カスタマイズ項目
- 設定ファイルサンプル

設定手順

1. テキストエディタを起動します。
2. 「 *カスタマイズ項目* 」のパラメータを、次の形式で指定します。

```
パラメータ名=設定値
```

> **コラム**
> パラメータの設定例については、「 *設定ファイルサンプル* 」を参照してください。

3. < %IOWEBDOC_HOME% /etc/ >に、< cela.txt >というファイル名で保存します。

以上で設定は完了です。

パラメータ名	概要
font	半角、全角、半角カタカナ用の全てのフォントを、 指定したフォントに変更します。
font1	半角用のフォントを、 指定したフォントに変更します。
font2	全角用のフォントを、 指定したフォントに変更します。
font3	半角カタカナ用のフォントを、 指定したフォントに変更します。
mode	空白行の出力を制限します。
noheadspace	ヘッダ部の後ろの間隔を制御します。
nofootspace	フッタ部の前の間隔を制御します。
pattern	網掛けパターンを塗りつぶし色パターンに変換して出力します。
V4821compat	パターンの出力方法を設定します。

font／font1／font2／font3

フォント名を指定し、半角、全角、または、半角カタカナ用のフォントを指定したフォントに変更します。

mode

空白行の出力を制限します。

次のいずれかを指定します。デフォルトは「old」です。

設定値	説明
old	空白行を出力する
fix	空白行を出力しない、かつ、フッタ位置固定
var	空白行を出力しない、かつ、フッタ位置可変

noheadspace

ヘッダ部の後ろの間隔を制御します。

次のいずれかを指定します。デフォルトは「n」です。

設定値	説明
y	ヘッダのスペースを空けない
n	ヘッダのスペースを空ける

nofootspace

フッタ部の前の間隔を制御します。

次のいずれかを指定します。デフォルトは「n」です。

設定値	説明
y	フッタのスペースを空けない
n	フッタのスペースを空ける

pattern

網掛けパターンを塗りつぶし色パターンに変換して出力します。

帳票レイアウトファイルで指定できるパターン（1～9）に対してRGB値（0～255）を指定します。

```
pattern={
1=c 255 0 0
2=c 0 255 0
3=c 0 0 255
4=c 0 255 255
5=c 255 0 255
6=c 255 255 0
7=c 128 255 255
8=c 255 128 255
9=c 255 255 128
}
```

> **注意**
> RGB値の前に「c」を必ず指定してください。

V4821compat

パターンの出力方法について、V4.8.2.1以前の方法、または、V4.8.2.2以降の方法を指定します。

- V4.8.2.1以前
  項目のパターン（網掛け）指定の出力時に罫線を出力する。
- V4.8.2.2以降
  項目のパターン（網掛け）指定の出力時に罫線を出力しない。

  > **注意**
  > 古いレイアウトを使用した場合に、罫線が消えてしまう問題が発生する場合があります。

次のいずれかを指定します。デフォルトは「n」です。

設定値	説明
y	V4.8.2.1以前の方法でパターンを出力する
n	V4.8.2.2以降の方法でパターンを出力する

設定ファイルサンプル

設定ファイル< cela.txt >のサンプルは、次の通りです。

```
#
# IOCELA runtime mode
#
#default:
# mode=old
# noheadspace=0
# nofootspace=0
# V4821compat=n
# font=
# font1=
# font2=
# font3=
#

################################################################################

# old:OldVersion, fix:FooterFixation, var:FooterVariable
#mode=old
#mode=fix
mode=var

################################################################################

# Between header block and data block
# y=no space
noheadspace=y

################################################################################

# Between footer block and data block
# y=no space
nofootspace=y

################################################################################

#Since IODOC V4.8.2.2/IOWebDOC V1.8.2.2
#Pattern frame control(Pattern and no frame support)
# y=Mode is older than V4.8.2.2
V4821compat=n

################################################################################

#Since IODOC V4.8.4/IOWebDOC V1.8.4
#Font control
#font=ＭＳ 明朝
font1=Courier New
#font2=ＭＳ 明朝
#font3=ＭＳ 明朝

################################################################################

#Since IODOC V4.9.1.2/IOWebDOC V1.9.1.2
################################################################################

pattern={
1=c 255   0   0
2=c   0 255   0
3=c   0   0 255
4=c   0 255 255
5=c 255   0 255
6=c 255 255   0
7=c 128 255 255
8=c 255 128 255
9=c 255 255 128
}
```