

# セット帳票

1. 概要 .....	2
2. 定義方法 .....	4
2-1. Job ファイルの作成 .....	4
2-2. Form ファイルの確認 .....	7
2-3. 入力データ .....	8
2-4. データマッピング .....	8
2-4-1. 入力データが Text ファイル、CSV ファイルの場合 .....	9
2-4-2. 入力データが XML ファイルの場合 .....	10
2-4-3. 入力データがデータベースの場合 .....	11
2-5. プレビュー確認 .....	12

## 1. 概要

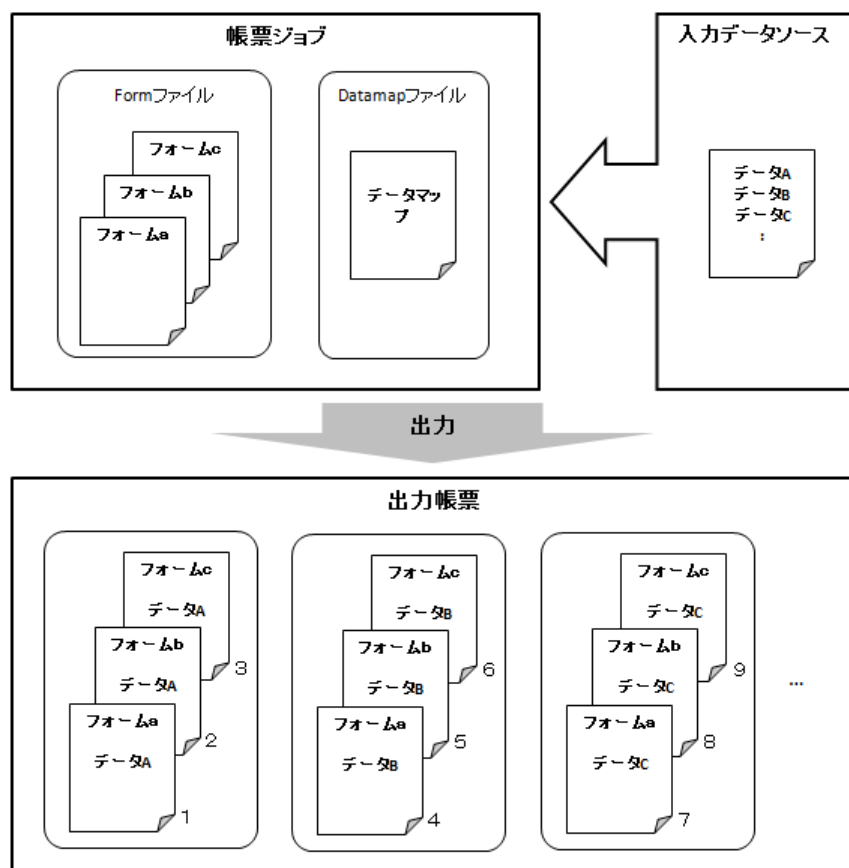
セット帳票機能は特殊なマルチフォーム機能です。

同一データを複数のフォームに割り当て、1つのセットとして出力します。変数名を全Formファイルで共通とすることで、マッピングした1箇所のデータを全てのFormファイル上に表示することができます。

a, b, c という3種類のフォームを持つセット帳票に対して、A, B, Cの3件分の入力データが与えられた場合、以下の順序で出力処理を行います。

1. フォーム a、入力データ A
2. フォーム b、入力データ A
3. フォーム c、入力データ A
4. フォーム a、入力データ B
5. フォーム b、入力データ B
6. フォーム c、入力データ B
7. フォーム a、入力データ C
8. フォーム b、入力データ C
9. フォーム c、入力データ C

図：イメージ



1 ページ分の入力データ 1 件に対し、フォーム a、b、c を一つのセットとして出力します。  
このように A、B、C の 3 件分の入力データが与えられた場合は、3 セット計 9 ページ分の帳票が出力されます。  
このセット帳票機能は、複写式のような伝票形式の帳票を出力する際に非常に効果的です。

## 2. 定義方法

セット帳票機能を利用するための作業の流れは通常の帳票作成と同様で、以下のようになります。

- \* 出力ジョブの定義
- \* フォームレイアウトの設計
- \* データマッピングの定義
- \* 帳票出力

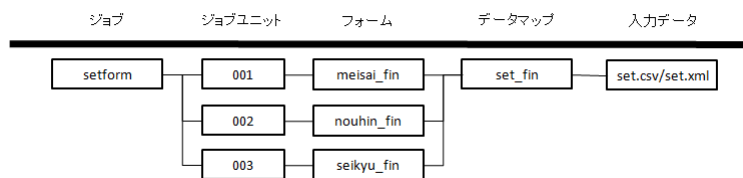
帳票サンプルを使用してセット帳票機能を利用した帳票の作成手順と定義方法について説明します。帳票サンプルに使用する帳票資源ファイルは、“帳票資源フォルダ¥tut¥set”に用意されています。

帳票資源フォルダはマネージャの[ヘルプ]-[バージョン情報]-[バージョン情報詳細]で確認できます。

※ Datamap ファイルは CSV データ形式と XML データ形式の 2 種類が用意されてますので、いずれかを使用してください。入力データファイルには Datamap ファイルで定義した形式のファイルを使用してください。

今回使用する帳票サンプルの全体の構成イメージは以下のようになります。

図：構成イメージ



セット帳票の特徴として、複数のフォームに対して 1 つのデータマップでデータマッピングを行います。データマップにはジョブユニットに登録されているすべてのフォームの変数がインポートされています。

### 2-1. Job ファイルの作成

セット帳票とする帳票ジョブを作成します。初めに Job ファイルの作成を行います。

#### ◆◆ 操作 ◆◆

マネージャを起動して、ディレクトリツリーから“帳票資源フォルダ¥tut¥set”を参照します。  
ツールボタン[ジョブ新規作成]をクリックします。

ジョブの新規作成ダイアログが表示されます。

## ◆◆ 操作 ◆◆

帳票名に“セット帳票”、Job ファイル名に“setform”(※)と入力します。

[ジョブ新規作成後に Form エディタを開く] チェックボックスのチェックを外します。

[OK] ボタンをクリックします。

※チュートリアルで同名の帳票を作成済の場合は別名を設定して下さい。

図：ジョブの新規作成ダイアログ

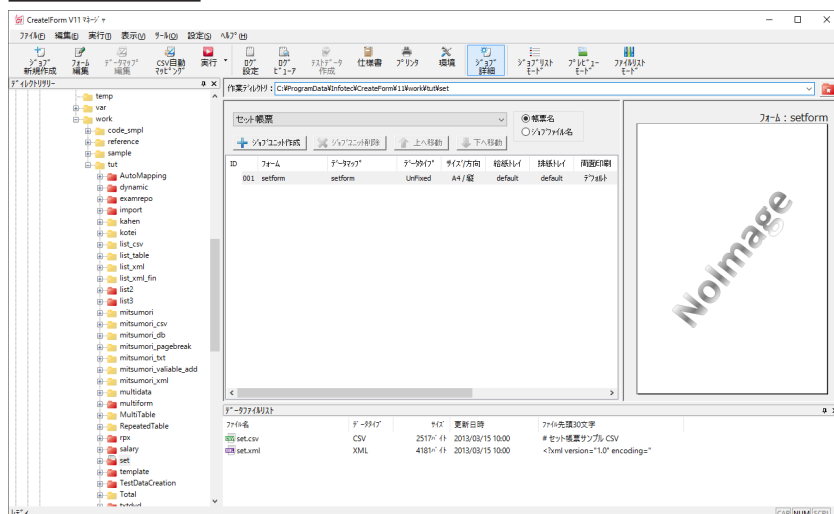
Job ファイルの新規作成が行われ、空の Form ファイルと Datamap ファイルが用意されました。現在定義されている空の Form ファイルと Datamap ファイルを、サンプルとして用意されている Form ファイルと Datamap ファイルに置き換えます。

## ◆◆ 操作 ◆◆

ツールボタン [ジョブ詳細] をクリックします。

ジョブ詳細画面において [ID] “001” 行をダブルクリックします。

図：ジョブ詳細画面



ジョブユニットのプロパティダイアログが表示されます。

## ◆◆ 操作 ◆◆

データマップ列の“setform”をダブルクリックして、“set\_fin.dmp”または“set\_fin.dmx”を選択し、[OK] ボタンをクリックします。

※“set\_fin. dmp”はCSVデータ形式、“set\_fin. dmx”はXMLデータ形式のDatamapファイルです。任意でいずれかのデータマップ形式を選択してください。なお、後述の「2-4. データマッピング」でそれぞれのデータマップ形式ごとの説明がありますので、データマップ形式に合わせてご覧ください。

[ フォーム ] に“meisai\_fin”、[ データマップ ] に“set\_fin”と表示されていることを確認し、[OK] ボタンをクリックします。

図：ジョブユニットのプロパティダイアログ

以上で [ ジョブユニット ID ] “001” の編集が完了しました。ジョブユニットを2つ追加し、残り2つのフォームの定義を行います。

#### ◆◆ 操作 ◆◆

[ ジョブユニット作成 ] ボタンをクリックします。

ジョブユニットの作成ダイアログが起動します。

#### ◆◆ 操作 ◆◆

ジョブユニットの作成ダイアログで [ セット帳票 ] ラジオボタンを選択します。

[ フォームファイル名 ] を初期値のまま、[OK] ボタンをクリックします。

図：ジョブユニットの作成ダイアログ

ジョブユニットが1つ追加されます。同様の操作をもう一度行い、もう1つジョブユニットを追加します。

ジョブユニットの追加後、ジョブユニットの編集ダイアログから“nouhin\_fin”と“seikyu\_fin”を定義していきます。

なお、セット帳票ではデータソースは一つですので、[ジョブユニットID]“002”以降の定義では、フォームの項目のみを定義していきます。

#### ◆◆ 操作 ◆◆

フォーム項目にある[参照]ボタンをクリックし、“nouhin\_fin.hst”と“seikyu\_fin.hst”のフォームをそれぞれジョブユニット“002”とジョブユニット“003”へ定義します。

3つのフォームの定義が完了すると、以下のようになります。

図：ジョブ詳細

The screenshot shows a window titled 'セット帳票' (Set Form) with a dropdown menu and several buttons: '+ ジョブユニット作成' (Add Job Unit), 'x ジョブユニット削除' (Delete Job Unit), '上へ移動' (Move Up), and '下へ移動' (Move Down). Below these is a table with the following data:

ID	フォーム	データマップ*	データタイプ*	サイズ/方向	給紙トレイ	排紙トレイ	両面印刷
001	meisai_fin	set_fin	CSV	A4 / 縦	default	default	デフォルト
002	nouhin_fin			A4 / 縦	default	default	デフォルト
003	seikyu_fin			A4 / 縦	default	default	デフォルト

以上で Job ファイルの作成は終了です。複数の Form ファイル“meisai\_fin”、“nouhin\_fin”、“seikyu\_fin”に対して1つの Datamap ファイル“set\_fin”が定義されています。これがセット帳票の特有の定義形式です。

## 2-2. Form ファイルの確認

ジョブユニットの作成時に“meisai\_fin”(明細書)、“nouhin\_fin”(納品書)、“seikyu\_fin”(請求書)と3つのフォームの定義を行いました。

ここでは、サンプル帳票で用意されているそれぞれの Form ファイルの内容を確認します。なお、これら3種類の Form ファイルはほぼ同じレイアウトでデザインされています。

Form エディタを使用して各 Form ファイルの内容を確認します。

#### ◆◆ 操作 ◆◆

ジョブ詳細画面からジョブユニットをクリックして選択し、ツールボタン[フォーム編集]をクリックします。

図：Form ファイル

The image shows three Form files side-by-side, labeled 明細書 (Details), 納品書 (Delivery), and 請求書 (Invoice). Each form has a header section with company information, a main table area, and a footer section with summary data and buttons.

それぞれの Form ファイルはほぼ同じレイアウトでデザインされているのが確認できます。また、各 Form ファイルでは同名の変数オブジェクトが配置されていることが確認できます。

#### ◆◆ 操作 ◆◆

各 Form ファイルの内容の確認ができたなら、Form エディタを終了します。

## 2-3. 入力データ

今回のセット帳票で使用する入力データには 3 件分のデータが用意されています。  
 “set.csv”(CSV データ形式) および “set.xml”(XML データ形式) です。  
 データマップの形式に合わせてデータファイルの内容を確認します。

#### ◆◆ 操作 ◆◆

データリストからデータファイル名 (“set.csv” または “set.xml”) をダブルクリックします。

Text エディタが起動します。

#### ◆◆ 操作 ◆◆

データファイルの内容が確認できたなら、Text エディタを終了します。

## 2-4. データマッピング

セット帳票として帳票の出力を行うには、Datamap ファイルを Form ファイルごとに用意するのではなく、すべての Form ファイル共通の Datamap ファイルを一つだけ用意します。  
 データマッピングの手順は、データマップ形式ごとに異なりますが、基本的には同じ作業になります。なお、サンプル帳票では既にデータマッピングが行われているので、先ほどの Job ファイル作成時に定義した Datamap ファイルの内容を確認します。

#### ◆◆ 操作 ◆◆

ジョブ詳細画面からジョブユニットをクリックして選択し、ツールボタン [ データマップ編集 ] をクリックします。

データ形式に応じた Datamap エディタが起動します。



## 2-4-1. 入力データがText ファイル、CSV ファイルの場合

それぞれのForm ファイルに含まれている変数オブジェクトがすべてインポートされ、データマッピングされていることを確認してください。変数オブジェクトには、他のForm ファイルに同名の変数オブジェクトが存在する変数オブジェクトと、そのForm ファイル固有の変数オブジェクトの2種類があります。

なお、この帳票サンプルではそれぞれのForm ファイルに含まれる変数オブジェクトがインポート済みの状態ですが、新規にデータマッピングを行う場合は、それぞれのフォームから変数オブジェクトをすべてインポートする必要があります。

### << Tips >>

変数オブジェクトのインポートはDatamap エディタのツールメニュー [ 編集 ] - [ 変数のインポート ] を選択し、対象のForm ファイルを選択します。選択したForm ファイルに含まれる変数オブジェクトの一覧が表示されますので、内容を確認して変数オブジェクトのインポートを行ってください。

次に、Datamap エディタのデータマップ設定から改ページの設定を行います。

今回のサンプル帳票では22行のデータが1ページ分のデータとして行数で改ページを行うように設定されています。

### ◆◆ 操作 ◆◆

ツールメニュー [ ファイル ] - [ データマップ設定 ] を選択します。

[ データマップ設定 ] ダイアログが表示されますので、[ ページ ] タブをクリックし、マッピング領域の [ 行数 ] と、改ページ処理の [ 行数で改ページ ] チェックボックスを確認します。

図：データマップ設定ダイアログ

行数での改ページが設定されている場合、1ページ分のデータの行数をマッピング領域の [ 行数 ] で設定します。

### ◆◆ 操作 ◆◆

内容の確認ができたなら [ キャンセル ] ボタンをクリックし、Datamap エディタを終了します。

## 2-4-2. 入力データが XML ファイルの場合

それぞれの Form ファイルに含まれている変数オブジェクトがすべてインポートされ、データマッピングされていることを確認してください。変数オブジェクトには、他の Form ファイルに同名の変数オブジェクトが存在する変数オブジェクトと、その Form ファイル固有の変数オブジェクトの 2 種類があります。

なお、この帳票サンプルではそれぞれの Form ファイルに含まれる変数オブジェクトがインポート済みの状態ですが、新規にデータマップを作成する場合は、それぞれのフォームから変数オブジェクトをすべてインポートする必要があります。

### << Tips >>

変数オブジェクトのインポートは Datamap エディタのツールメニュー [ 編集 ] - [ 変数のインポート ] を選択し、対象の Form ファイルを選択します。選択した Form ファイルに含まれる変数オブジェクトの一覧が表示されますので、内容を確認して変数オブジェクトのインポートを行ってください。

次に、Datamap エディタのデータマップ設定から改ページの設定を行います。

XML ファイルを使用する場合は、改ページ設定として「どの要素（タグ）以下を 1 ページ分のデータとしてみなすか」という設定を行います。

今回のサンプル帳票の XML ファイル構造は以下のようになっています。

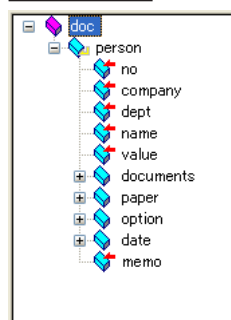
```
/doc
  /person
    /...
```

「person」要素が一人分のデータ（レコード）となっていることが分かるかと思います。

そこで、この「person」要素に対して改ページの指定を行います。

※ XML での改ページ設定の詳細についてはマネージャのメニュー [ ヘルプ ] - [ オンラインマニュアル ] から「3. 機能リファレンス」-「3.2 エディタの操作」-「XML 形式の Datamap エディタ」の「3-3. XML での改ページ設定について」をご覧ください。

図：XML ツリー



### ◆◆ 操作 ◆◆

ツールメニュー [ モード ] - [ 改ページ設定 ] を選択します。

改ページ設定モードに切り替わりますので、この状態で画面右側の XML ツリーから「person」要素をクリックします。「改ページ設定」ダイアログが表示されますので、改ページ設定を確認します。

図：改ページ設定ダイアログ

この改ページ設定では「person」要素の終了ごとに改ページが行われます。

#### ◆◆ 操作 ◆◆

内容の確認ができたなら [ キャンセル ] ボタンをクリックし、Datamap エディタを終了します。

### 2-4-3. 入力データがデータベースの場合

今回の帳票サンプルではデータベースは使用していませんが、データベースを使用する場合も他のデータマップ形式の場合と同様に、それぞれのフォームから変数オブジェクトをすべてインポートする必要があります。

なお、データベースを使用する場合も他のデータマップ形式と同様に、改ページ設定として「何レコード（何行分）を1ページ分のデータとしてみなすか」という設定を行います。

図：行数設定

#### ◆◆ 操作 ◆◆

ツールメニュー [ ファイル ] - [ ファイル設定 ] を選択します。

[ ファイル設定 ] ダイアログが表示されるので、「ページ行数」の値を適切な値に設定します。

## 2-5. プレビュー確認

ここまで説明した内容で、セット帳票の基本的な設定は完了です。

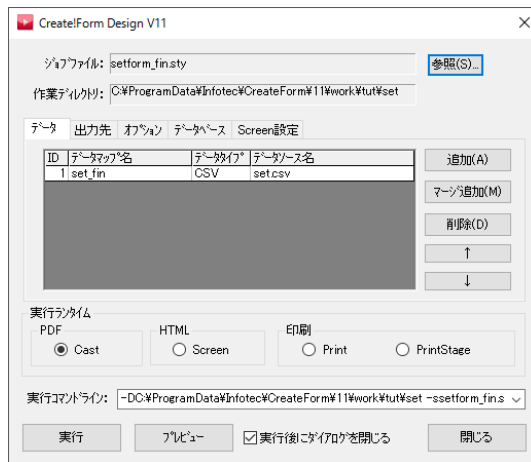
それでは、実際に作成した Job ファイルとデータファイルを使用してセット帳票の出力結果を確認します。

### ◆◆ 操作 ◆◆

Job ファイル “setform.sty” を選択し、ツールメニュー [ 実行 ] をクリックします。

テスト実行ダイアログが表示されます。

図：テスト実行ダイアログ



「データマップ名」に “set\_fin” が表示されていることを確認して下さい。

「データソース名」は「2-1. ジョブファイルの作成」で定義したデータマップ形式によって異なります。「データタイプ」が CSV の場合は “set\_fin.csv” が、XML の場合は “set\_fin.xml” が表示されていることを確認してください。

もし、データマップ名が正しく表示されていない場合は、ジョブファイル “setform.sty” のジョブユニットの定義が正しく行われているか確認してください。また、データソース名が正しく表示されていない場合は、「データマップ名」を選択し、[ 追加 ] ボタンからデータソースの割り当てを行ってください。

それでは、実際にテスト実行を行ってセット帳票の出力結果を確認します。

### ◆◆ 操作 ◆◆

[ 実行ランタイム ]-[PDF]-[Cast] のラジオボタンが選択されていることを確認して、[ プレビュー ] ボタンをクリックします。

Adobe Acrobat、もしくは Adobe Reader が起動して帳票出力イメージが表示されます。

データマップ形式が CSV の場合、Datamap ファイルには行数による改ページが設定されています。そのため、22 行ごとに改ページが行われ 3 件分のデータが出力されます。

データマップ形式が XML の場合、Datamap ファイルには要素の終了ごとの改ページが設定されています。そのため、設定された要素 “person” の終了ごとに改ページが行われ、3 件分のデータが出力されます。

1 件のデータにつき、「明細書」「納品書」「請求書」の 3 ページが出力されるため、いずれのデータマップ形式でも出力されるページ数は 9 ページになります。それぞれのページの上部中央に出力される書類名および上部右側に出力される No はそれぞれ以下の通りです。

- 1 ページ目：明細書、No. 000000
- 2 ページ目：納品書、No. 000000
- 3 ページ目：請求書、No. 000000
- 4 ページ目：明細書、No. 000001
- 5 ページ目：納品書、No. 000001
- 6 ページ目：請求書、No. 000001
- 7 ページ目：明細書、No. 000002
- 8 ページ目：納品書、No. 000002
- 9 ページ目：請求書、No. 000002

図：セット帳票の出力結果 (1 ページ目から 3 ページ目)

The figure displays three sample forms for '明細書' (Statement of Work), '納品書' (Delivery Note), and '請求書' (Invoice). Each form is structured as follows:

- Header:** Includes company name (〇〇株式会社), address (〇〇市〇〇区〇〇町〇〇番地), and contact information (TEL: 03-XXXX-XXXX).
- Table:** A table with columns for '品名' (Item Name), '数量' (Quantity), and '単価' (Unit Price). The table is populated with sample data.
- Footer:** Includes a section for '合計金額' (Total Amount) and '消費税' (Consumption Tax).

以上でセット帳票の作成手順と定義方法についての説明を終了します。