

【セッションNo. 4】

# Valence App Builder RPG連携テクニック

株式会社ミガロ。  
RAD事業部 技術支援課  
尾崎 浩司



# 【アジェンダ】

- Valenceについて
- Valence App Builder 開発手法
- Valence App Builder RPG連携方法の基本
- Valence RPG ToolKit 活用テクニック
- 新機能 Fusion5250



# Valenceについて



# ■ Valenceの概要

- IBM i を Web環境で活用できるモダナイゼーション運用・開発ツール
- 3つのポイント

## ローコード開発ツールによる超高速アプリ構築

ほぼノンプログラミングでアプリが作成できる開発ツールを搭載。ウィザードを使用した3つのステップで簡単構築。複雑な処理が必要な場合は、RPGでロジック追加が可能。

## ビジュアルなIBM i 運用管理ツール群

ジョブの管理、DB2/400データベースの閲覧・編集、スプール・ファイルの確認等のIBM iの開発・運用管理に必要なツールが、IBM i コマンド無しでビジュアルに使用可能。

## IBM i のみで環境構築可能

IBM i 標準のHTTP Server機能を使用する為、環境構築にWindows ServerやWebsphere Application Server(WAS)は不要。Valence開発者および使用者は、全てブラウザからWebアクセスで利用可能。



# ■ Valenceの構成

- ローコード開発ツール
  - Valence App Builder  
IBM i 業務アプリを超高速で実現するローコード開発プラットフォーム。  
作成したValenceアプリは、Valence Portalから実行可能
- Valenceユーティリティ
  - Valence iAdmin  
ブラウザやスマホ/タブレットといったモバイルからIBM iの運用管理が実現できるユーティリティ
  - Valence File Editor  
DB2/400データベースをブラウザ上で直接編集できるユーティリティ
  - Valence Spool Viewer  
アウトキューおよびスプールファイルを閲覧できるユーティリティ
- Valenceポータル
  - Valence Portal  
IBM i で稼働するValenceアプリの使用・管理の為に用意された統合画面

## Valenceの構成

ローコード  
開発ツール

**Valence  
App Builder**

Valence  
ユーティリティ

**Valence  
iAdmin**

**Valence  
File Editor**

**Valence  
Spool Viewer**

Valenceポータル

**Valence  
Portal**



# Valence画面イメージ

## Valenceポータル

valence

テクニカルセミナー

- 商品メンテナンス (新規登録)
- ユーザーマスターメンテナンス
- 商品マスター権
- 担当者権

例 - Nitro App Builder

- Customer Dashboard
- IBM i Tables
- Customer Maintenance
- Customer Maintenance (Map)

**Valence Portal**

## Valenceユーティリティ

Nitro Admin - CPU

Valence

Nitro File Editor

VALENCEBLVMAUTP

並び替え | 著者コード asc

著者コード ↑	著者名	最終字種
1	山田 太郎	東京情報大
2	田中 花子	大阪音楽大
3	大森 和子	愛知大
4	加藤 伸一	高山経済大

トップ活動ジョブ (2)

ジョブ名	開始	ユーザー
VALENCE01	21:40:00	QDZDZTTP

**Valence iAdmin**

## Valence File Editor

## Valence Spool Viewer

## Valence アプリケーション (作成例)

IBM i Dashboard

18:44:42  
2018/09/05

ディスク使用率(IBM ユーザー単位)

CPU使用率

活動ジョブ数  
308

ディスク使用率  
77.27%

活動ジョブ

ジョブ名	CPU	ステータス	サブシステム
024196QZTMHHTPVALENCE42	0.4	RUN	QHTPPSVR
024197QZTMHHTPVALENCE42	0.1	TMS	QHTPPSVR
024198QZTMHHTPVALENCE42	0.1	TMS	QHTPPSVR
024199QZTMHHTPVALENCE42	0.1	TMS	QHTPPSVR
024200QZTMHHTPVALENCE42	0.1	TMS	QHTPPSVR

得最先マスターメンテナンス

東京都の取引先

200103 : 海南興業株式会社

住所: 東京都世田谷区

〒158-0004 東京都世田谷区

TEL: FAX:

## ■ 今回のポイント

1. 本セッションでは、ローコード開発ツール[Valence App Builder]のアプリケーション開発テクニックを紹介
  - ウィザードだけで出来るValence App Builder開発の基本
  - RPGを組み合わせたロジック追加方法
  - Valenceの機能を拡張する RPG ToolKit
2. 次バージョンに搭載される新機能について紹介
  - 新機能 Fusion5250

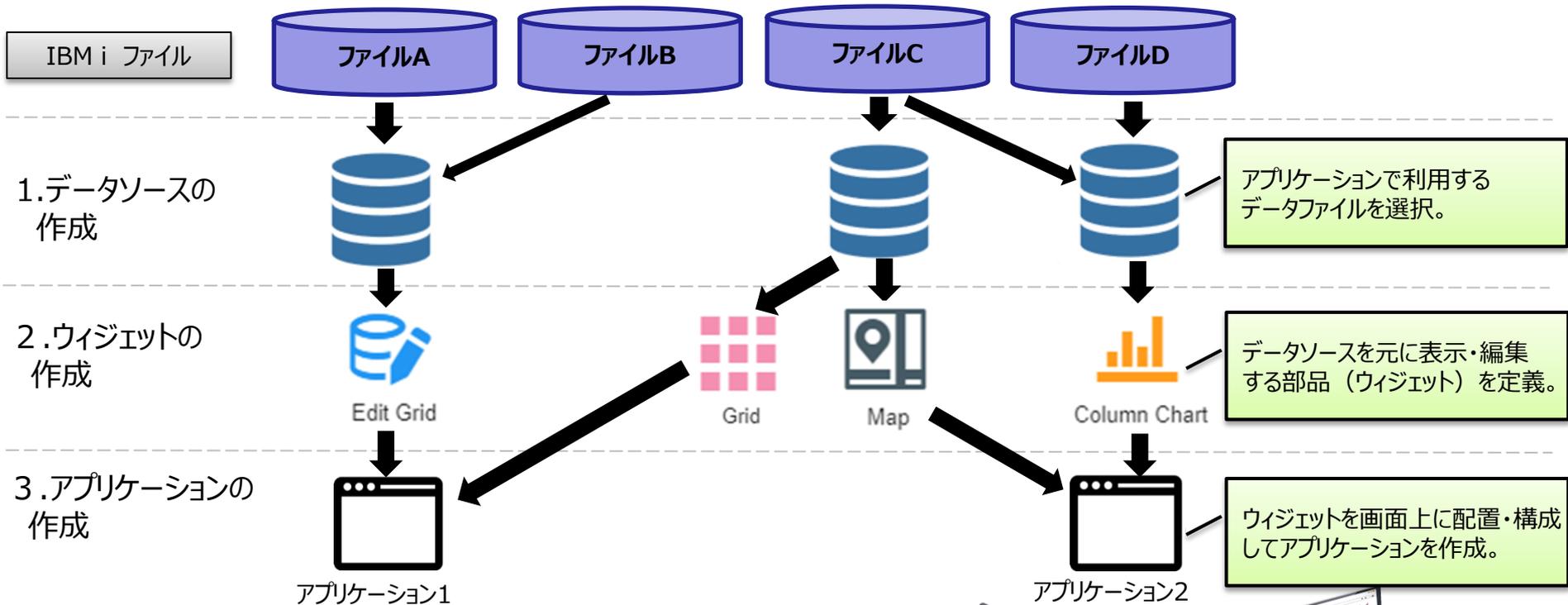


# Valence App Builder 開発手法



# Valence App Builder 開発手法

- ウィザードを使用したシンプルな3つの開発ステップ



# Valence App Builder 開発手法

## ステップ1：データソースの作成

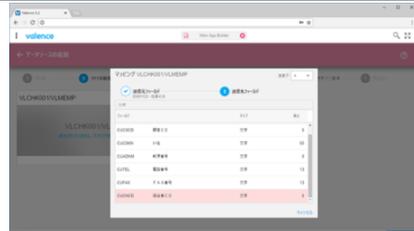
- IBM i上のファイル、フィールドを選択し、絞り込み条件や並び順等を指定する。

### 1. ファイル



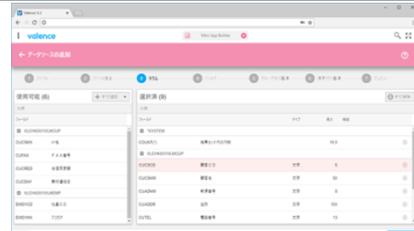
データの元となるファイルを選択

### 2. ファイル結合



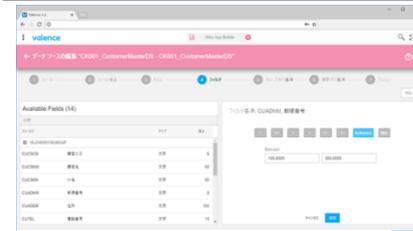
複数ファイル使用時の結合条件を指定

### 3. カラム



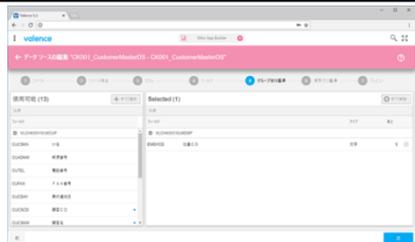
データベースから必要な項目を選択

### 4. フィルタ



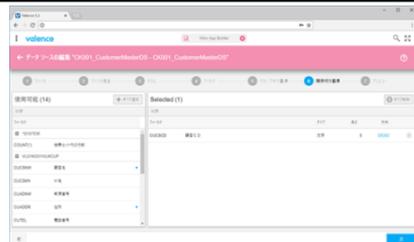
レコードの抽出条件を指定

### 5. グループ分け



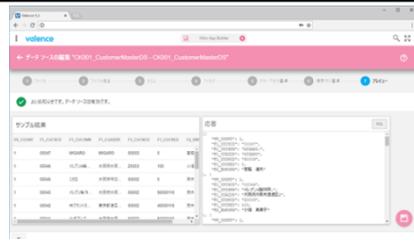
項目単位で数値またはカウンター集計

### 6. 順序付け



昇順/降順、出力順を指定

### 7. プレビュー



設定条件を実行、データを表示

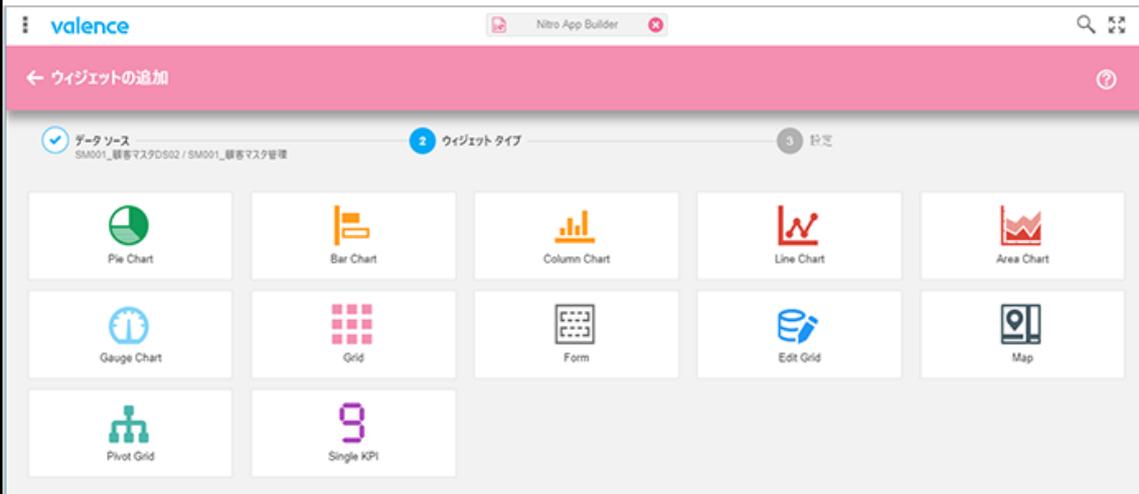


名前を付けてデータソースを保存

# ■ Valence App Builder 開発手法

## • ステップ2：ウィジェットの作成

- ステップ1で作成したデータソースを元にウィジェットを作成。
- 12種類よりウィジェットを選択し、カラムの紐づけやフィルタ設定等を行う。



### 【ウィジェットのタイプ】

#### • グリッド

グリッド（表） / ピボットグリッド  
編集用グリッド

#### • グラフ

円グラフ / 横棒グラフ /  
縦棒グラフ / 折れ線グラフ /  
面グラフ / ゲージ

#### • その他

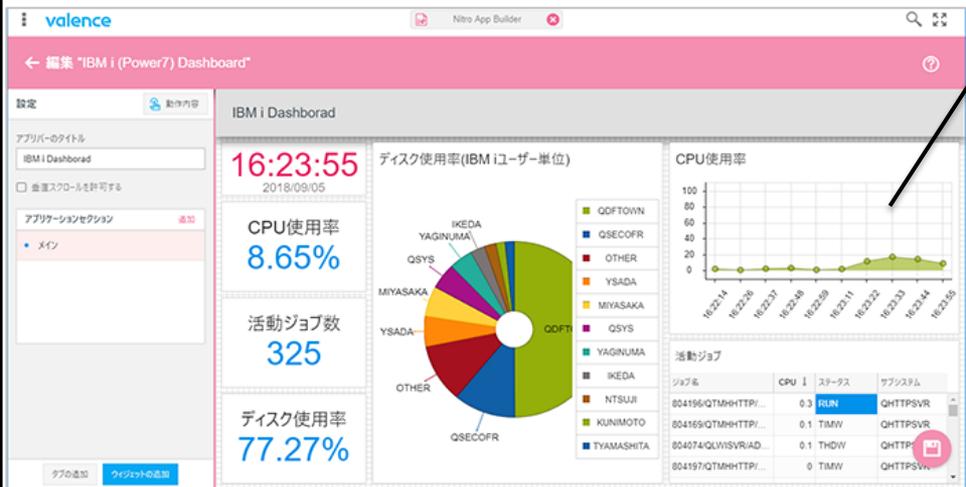
フォーム / 指標表示(KPI) / 地図



# Valence App Builder 開発手法

## ステップ3：アプリケーションの作成

- ステップ2で作成した1つあるいは複数のウィジェットを配置して、アプリケーションを作成する。



画面セクション（ワークスペース）上に1つあるいは複数のウィジェットを配置

ウィジェットのイベント（行クリック等）に対して、ウィジェットの表示/非表示や、別のウィジェットへのデータ絞り込み等のアクションが設定できる。



# Valence App Builder RPG連携方法の基本



# ■ Valence App Builder RPG連携

## • Valenceのウィジェット

- 編集グリッド（Edit Grid）ウィジェットを使用したデータソースの更新が可能。
- Edit Gridの編集機能や、Formウィジェット上の入力欄に対して、必須項目の指定もできる。

## • ウィジェット単体ではできない事

- 項目相関関係に基づくエラーチェック
- 受注データソースを更新する時に、関連する在庫ファイルをあわせて更新



Valence App Builder単体で実現できない処理については、  
RPGを追加実装することで、対応する。



# ■ Valence App Builder RPG連携

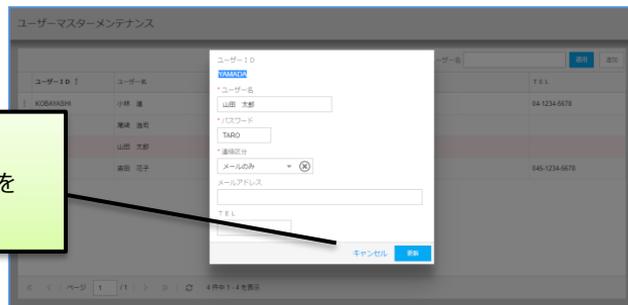
## • Valence App BuilderでRPGを呼び出すタイミング

1. グリッド上の行やグラフをクリックした時 あるいは、  
ウィジェットやアプリケーションセクション上に追加した  
ボタンをクリックした時 (**ボタンクリック**)



2. 編集グリッド(Edit Grid)で、レコードの追加/編集/  
削除を行う時 (**編集**)

Edit Gridの行編集時  
「更新」クリック時にRPGを  
呼び出し



3. グリッド等でユーザーがフィルタ (絞り込み) を行った時  
(**フィルタ**)

Gridのフィルタ条件を  
指定して「適用」クリック時に  
RPGを呼び出し



# ■ Valence App Builder RPG連携

## • RPG呼び出し設定方法

### 1. ボタンクリック

- アプリケーション作成時の「動作内容」を使用して、「RPGプログラムの呼び出し」を設定する。

The screenshot shows the Valence App Builder interface. On the left, the '動作内容' (Action Content) panel is visible, with a red circle around the 'RPGプログラムの呼び出し' (Call RPG Program) action. A green box labeled 'ボタン追加' (Add Button) points to the 'ボタン' (Button) widget in the application section. A purple arrow points from the 'RPGプログラムの呼び出し' action to the configuration dialog. A green box labeled 'アクション追加' (Add Action) points to the 'RPGプログラムの呼び出し' action in the list. The configuration dialog, titled 'RPGプログラムの呼び出し', shows the application section as 'サブ > 受注登録ヘッダウィジェット - 受注登録ヘッダ' and the action as 'オプション'. The 'プログラム' (Program) field is set to 'TEC010'. A red box highlights the 'TEC010' field, with a green box containing the text: 「RPGプログラムの呼び出し」アクションを追加してプログラムIDを指定する。 (Add the 'Call RPG Program' action and specify the program ID.)

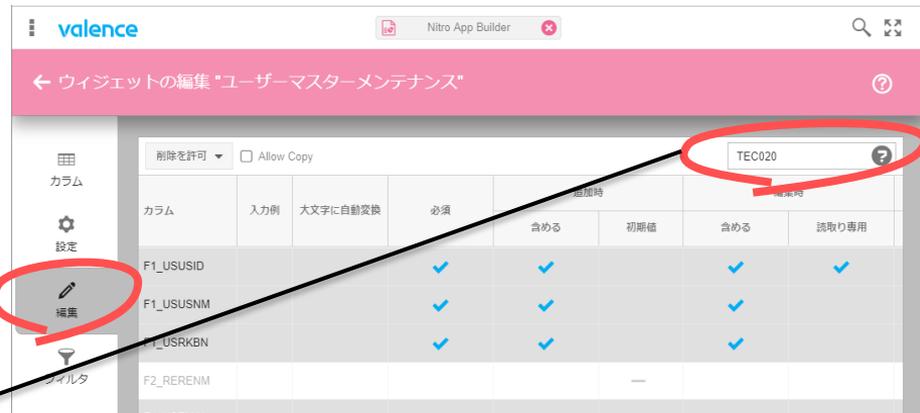
# Valence App Builder RPG連携

## RPG呼び出し設定方法

### 2. 編集

- EditGridウィジェットの設定：「編集」タブで、RPGプログラムを指定する。

レコード編集の条件を設定する  
「編集」タブ上でプログラムIDを指定する。



### 3. フィルタ

- ウィジェットの設定：「フィルタ」タブで、RPGプログラムを指定する。

データソースの絞り込み条件を設定する  
「フィルタ」タブ上でプログラムIDを指定する。



# ■ Valence App Builder RPG連携

## • Valence App Builderで使用するRPGプログラム

- App BuilderとRPGを連携する為、APIプログラムが含まれるテンプレートを使用する。  
テンプレートファイルをコピーして、プログラムを作成する。

Valenceライブラリ (V5.2の場合 VALENCE52)  
QRPGLESRC

1. ボタンクリック : **EXNABBTN**
2. 編集 : **EXNABVAL**
3. フィルタ : **EXNABFLT**

PDM を使用したメンバーの処理

ファイル . . . . . QRPGLESRC  
ライブラリー : VALENCE52 位置指定 . . . . .

オプションを入力して、実行キーを押してください。

2= 編集    3=コピー    4= 削除    5= 表示    6= 印刷    7= 名前の変更  
8= 記述の表示    9= 保管    13=テキスト の変更    14=コンパイル    15=モジュール 作成 . . .

OPT	メンバー	タイプ	テキスト
—	<b>EXNABBTN</b>	SQLRPGLE	NAB Examples - Button Call
—	<b>EXNABFLT</b>	SQLRPGLE	NAB Examples - Widget Filter Program
—	<b>EXNABVAL</b>	SQLRPGLE	NAB Examples - Datasource Edit Validation
—	EXNS01	RPGLE	Valence Examples - No Security - Simple Form
—	EXPDF01	RPGLE	Valence Examples - PDF - Create a PDF from RPG 1
—	EXPDF02	RPGLE	Valence Examples - PDF - Create a PDF from RPG 2

テンプレートは、フリーフォームRPGで作成されており、連携に必要な処理が予め記述されている



# ■ Valence App Builder RPG連携

## ● クリック時 EXNABBTN テンプレートプログラム

```
0001.00 /copy qcylesrc, vvHspec
0002.00 ** -----
0003.00 **      Copyright (C) 2008-2018 CNX Corporation
0004.00 ** -----
0005.00 **      Object ID: EXNABBTN
0006.00 **      Version: V5.2
```

<<< 中略 >>>

```
0104.00 ** NOTE: this must be compiled with RPGPPOPT = *LVL2
0105.00 ** -----
0106.00 d exnabbtn      pr 1-①
0107.00 d exnabbtn      pi
0108.00 /define nabButton
0109.00 /include qcylesrc, vvNabTmpl
0110.00 ** -----
0111.00 ** program start
0112.00 ** -----
0113.00 /free
0114.00 Initialize();
0115.00 // place your code within the Process procedure...
0116.00 //
0117.00 Process();
0118.00 CleanUp();
0119.00 *inlr=*on;
0120.00 /end-free
0121.00 **
```

### 1-①

実行するプログラムの定義  
(プロトタイプ (PR) およびプロシージャ-  
インターフェース (PI) )

アプリ作成時には、[プログラムID] を指定する。

### 1-②

プログラムのメイン処理

Processサブプロシージャを呼び出ししている。  
Processサブプロシージャ内にロジックを作成する。

# ■ Valence App Builder RPG連携

## • クリック時 EXNABBTN テンプレートプログラム

```
0122.00 p Process      b
0123.00 d              pi
0124.00 d IMyChar      s          10a
0125.00 d IMyNum     s          10i 0
0126.00 d IMyStmt   s          65535a
0127.00 /free
0128.00 // sample of retrieving filter values...
0129.00 //
<<< 中略 >>>
0157.00 //
0158.00 vvOut_toJsonPair(' success:true, info:Processing completed, refresh:true');
0159.00 /end-free
0160.00 p              e
0161.00 /include qcpylesrc, vvNabTmpI
```

1-③

1-③

Process サブプロシージャ処理

この中に必要なアプリケーション処理を記述する。

テンプレートプログラムには、フリーフォームRPGによるコーディング例がコメントとして記載されている。  
(実際の開発時は、コメントは削除可能。)

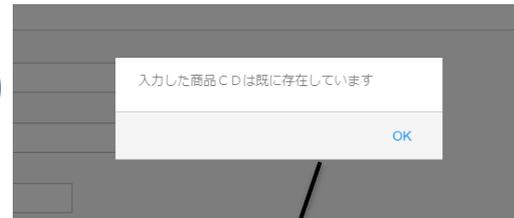
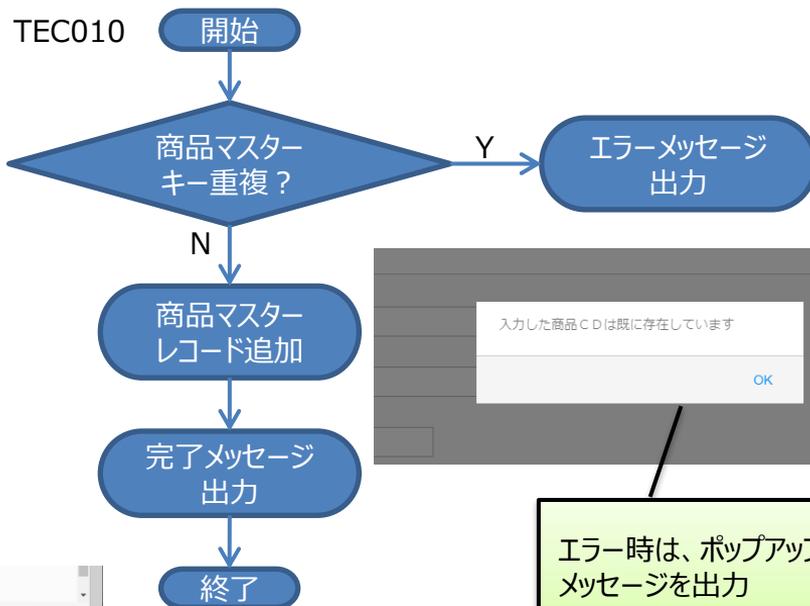
テンプレートプログラムには、メインプログラムが記述されており、その中で、サブプロシージャが呼び出されるようになっている。サブプロシージャ内に必要なロジックを追加すればよい。

# Valence App Builder RPG連携

## サンプルプログラム 1 (ボタンクリック)

- フォームウィジェットに追加した[登録]ボタン押下時に、RPG (ID:TEC010) を呼び出す。

正常終了時は、  
お知らせメッセージを出力



エラー時は、ポップアップ  
メッセージを出力

# Valence App Builder RPG連携

## TEC010: 商品マスタメンテナンス新規登録

```
0001.00 /copy qcpylesrc, vvHspec
0002.00 **
0003.00 ** TEC010: 商品マスタメンテナンス新規登録
0004.00 **
0005.00 F*
0006.00 F* ファイル定義
0007.00 F*
0008.00 F*<商品マスタ>
0009.00 FMSYOHIN UF A E K DISK
0010.00 F*
0011.00 d TEC010 pr
0012.00 d TEC010 pi
0013.00 /define nabButton
0014.00 /include qcpylesrc, vvNabTmpl
0015.00 **
0016.00 ** program start
0017.00 **
0018.00 /free
0019.00 Initialize();
0020.00 Process();
0021.00 CleanUp();
0022.00 *inlr=*on;
0023.00 /end-free
0024.00 **
0025.00 p Process b
0026.00 d pi
0027.00 D VSYHNC D S 10A
0028.00 D VSYHNM D S 32A
0029.00 D VTANKA D S 9 0
0030.00 D*
```

F仕様書

ファイル定義 (MSYOHIN : 商品マスタ)

2-①

プログラムID(TEC010)を指定

2-②

Valence Web画面上の入力値を保持する為の変数を定義

2-②



# Valence App Builder RPG連携

## TEC010

```
0031.00 /free
0032.00 //フォーム上の値を取得
0033.00 VSYHNGD = GetFormChar('F1_SYHNGD'); //商品CD
0034.00 VSYHNNM = GetFormChar('F1_SYHNNM'); //商品名
0035.00 VTANKA = GetFormNum('F1_TANKA'); //単価
0036.00 /end-free
0037.00 C*
0038.00 C*---キー重複チェック
0039.00 C VSYHNGD CHAIN MSYOHIR 81
0040.00 C *IN81 IFEQ *OFF
0041.00 C*
0042.00 /free
0043.00 //エラーメッセージを送信
0044.00 vvOut_toJsonPair('success:false,'
0045.00 + 'msg:入力した商品CDは既に存在しています');
0046.00 /end-free
0047.00 C*
0048.00 C ELSE
0049.00 C*---新規レコード登録
0050.00 C MOVEL VSYHNGD SYHNGD
0051.00 C MOVEL VSYHNNM SYHNNM
0052.00 C Z-ADD VTANKA TANKA
0053.00 C*
0054.00 C WRITE MSYOHIR
0055.00 C*
0056.00 /free
0057.00 //正常終了メッセージを送信
0058.00 vvOut_toJsonPair('success:true,refresh:true,'
0059.00 + 'info:登録が終了しました');
0060.00 /end-free
0061.00 C*
0062.00 C END
0063.00 P e
0064.00 /include qcpylesrc,vvNabTmpI
```

2-③

2-③

画面上の値を取得する処理  
GetFormChar (文字フィールド)  
GetFormNum (数値フィールド)

2-⑤

2-⑤

2-⑥

処理結果 (レスポンス) を送信  
する処理

vvOut\_toJsonPair  
success : true / false

2-④

2-④

アプリケーション処理  
キー重複チェック  
新規レコード登録

2-⑥

# ■ Valence App Builder RPG連携

- ボタンクリック： EXNABBTN
  - 画面上の情報を GetFormChar / GetFormNum にて取得
  - 処理結果（レスポンス）は、vvOut\_toJsonPairにて返却
  - 入出力部分のAPIは、フリーフォームを使用するが、その他部分は固定長RPGでも良い。

## 主なAPI

API	戻り値	パラメータ
GetFormChar	取得文字列	fld : ウィジェットのカラム (必須) fldType : 属性 ("O" or "J") を指定 ("A"の時は不要) length : "J"の時のみ、カラムのフィールド長を指定
GetFormNum	取得数値	fld : ウィジェットのカラム (必須)
vvOut_toJsonPair	---	success : 正常終了時 true / エラー時 false (必須) fld : エラー対象のフィールドを指定 msg : ポップアップメッセージ出力 fld指定時は、フィールド下部にメッセージを出力 info : SnackBar(画面下部) に一時メッセージ出力 refresh : true / false (true: ウィジェットを更新)



# ■ Valence App Builder RPG連携

## • RPGプログラムのコンパイル

- Valenceライブラリ（V5.2の場合、VALENCE52）をライブラリリストに追加する。
- RPGプリプロセッサ・オプション を [\*LVL2]とする。

```
SQL ILE RPG オブジェクト の作成 (CRTSQLRPG1)

選択項目を入力して、実行キーを押してください。

オブジェクト          > TEC010          名前
ライブラリ            > TCREP19        名前 , *CURLIB
ソース・ファイル      > QRPGLSRC       名前 , QRPGLSRC
ライブラリ            > TCREP19        名前 , *LIBL, *CURLIB
ソース・メンバー      > TEC010         名前 , *OBJ
ソース・ストリーム・ファイル

コミットメント制御    *NONE           *CHG, *ALL, *CS, *NONE...
リレーショナル・データベース
コンパイル・タイプ    > *PGM           *PGM, *SRVPGM, *MODULE
出力リスト            *NONE           *NONE, *PRINT
テキスト 記述         *SRCMBRTXT

追加のパラメーター

プリコンパイラ・オプション
値の続きは+
RPG プリプロセッサ・オプション  *LVL2          *NONE, *LVL1, *LVL2
ターゲット・リリース          *CURRENT       *CURRENT, *PRV, V5R4MO, ...
INCLUDE ファイル              *SRCFILE       名前 , *SRCFILE
ライブラリ                    *LIBL          名前 , *LIBL, *CURLIB
SQL INCLUDE ディレクトリー    *NONE
```



# Valence App Builder RPG連携

## サンプルプログラム2 (編集)

- Edit Gridで、新規レコード[作成] 時に、RPG (ID:TEC020) を呼び出す。



[追加]ボタンをクリックし、  
行登録画面に遷移  
(Edit Grid 標準機能)

ユーザーID ↑	ユーザー名	連絡区分名	メールアドレス	TEL
KOBAYASHI	小林 進	電話のみ		04-1234-5678
MIGARO	ミガロ 次郎	メール&電話	migaro@migaro.co.jp	06-6631-8601
OZAKI	尾崎 浩司	メールのみ	ozaki@abc.com	
YAMADA	山田 太郎	メールのみ	yamada@zzz.com	



連絡区分は、  
1: メールのみ  
2: 電話のみ  
3: メール&電話  
よりプルダウンで選択

ユーザーID ↑

ユーザー名  
ミガロユーザー

\*連絡区分  
メールのみ

メールアドレス

TEL

キャンセル 作成

データ作成時に、RPG (TEC020) を呼び出す。  
連絡区分の値により、下記エラーチェックを行う

- 1: メールのみ → メールアドレスは、必須
- 2: 電話のみ → TELは、必須
- 3: メール&電話 → メールアドレスおよびTEL ともに必須

未入力の場合、エラーメッセージを出力する。

エラーメッセージ出力

メールのみの場合、メールアドレスは必須です。

OK

# Valence App Builder RPG連携

## TEC020: ユーザーメンテナンス

- テンプレート: EXNABVAL をコピーして作成

```
0001.00 /copy qcpylesrc, vvHspec
0002.00 ** -----
0003.00 ** TEC020:ユーザーメンテナンス
0004.00 ** -----
0005.00 F* -----
0006.00 F* ファイル定義
0007.00 F* -----
0008.00 F* <メッセージマスタ>
0009.00 FMMSGP IF E K DISK
0010.00 F* -----
0011.00 d TEC020 3-① pr 10a
0012.00 d inMode *
0013.00 d inDataPtr *
0014.00 d outStopProcess... n
0015.00 d -----
0016.00 d TEC020 3-① pi 10a
0017.00 d inMode *
0018.00 d inDataPtr *
0019.00 d outStopProcess... n
0020.00 d -----
0021.00 /define nabValidation
0022.00 /include qcpylesrc, vvNabTmp1
0023.00 ** -----
0024.00 ** program start
0025.00 ** -----
```

F仕様書  
ファイル定義 (エラーメッセージマスタ)

3-①  
プログラムID(TEC020)を指定。

```
0023.00 ** -----
0024.00 ** program start
0025.00 ** -----
0026.00 /free
0027.00 Initialize();
0028.00
0029.00 // perform validations based on the mode...
0030.00 //
0031.00 if inMode = 'ADD';
0032.00 ProcessAdd();
0033.00 elseif inMode = 'EDIT';
0034.00 ProcessEdit();
0035.00 elseif inMode = 'DELETE';
0036.00 ProcessDelete();
0037.00 elseif inMode = 'POSTADD';
0038.00 ProcessPostAdd();
0039.00 elseif inMode = 'POSTDELETE';
0040.00 ProcessPostDelete();
0041.00 elseif inMode = 'POSTEDIT';
0042.00 ProcessPostEdit();
0043.00 endif;
0044.00
0045.00 CleanUp();
0046.00 *inlr=*on;
0047.00 /end-free
0048.00 ** -----
```

3-②

プログラムメイン処理  
inMode : 実行時のモード  
"ADD" - 行追加時  
"EDIT" - 行編集時  
"DELETE" - 行削除時

モード別のサブプロシージャ内に  
各プログラム処理を記述する。

3-②

# Valence App Builder RPG連携

## TEC020

```
0049.00 p ProcessAdd b
0050.00 d pi
0051.00 **
0052.00 D LUSRKBN S 1A
0053.00 D LUSEMAL S 40A
0054.00 D LUSTEL S 14A
0055.00 D LERR S 5A
0056.00 **
0057.00 /free
0058.00 //----- 画面上の値を取得
0059.00 LUSRKBN = GetValue(' MUSER' : ' USRKBN' ); //---連絡区分
0060.00 LUSEMAL = GetValue(' MUSER' : ' USEMAL' ); //---メールアドレス
0061.00 LUSTEL = GetValue(' MUSER' : ' USTEL' ); //---T E L
0062.00 /end-free
0063.00 C*-----連絡区分=1(メールのみの場合)
0064.00 C LUSRKBN IFEQ '1'
0065.00 C LUSEMAL IFEQ *BLANK
0066.00 C MOVEL 'E0010' LERR
0067.00 C ENDIF
0068.00 C ENDIF
0069.00 C*-----連絡区分=2(電話のみの場合)
0070.00 C LUSRKBN IFEQ '2'
0071.00 C LUSTEL IFEQ *BLANK
0072.00 C MOVEL 'E0020' LERR
0073.00 C ENDIF
0074.00 C ENDIF
```

3-③  
行追加時 アプリケーション処理  
ProcessAdd サブプロシージャ

3-④

```
/free
//----- 画面上の値を取得
LUSRKBN = GetValue(' MUSER' : ' USRKBN' ); //---連絡区分
LUSEMAL = GetValue(' MUSER' : ' USEMAL' ); //---メールアドレス
LUSTEL = GetValue(' MUSER' : ' USTEL' ); //---T E L
/end-free
```

3-③

3-④  
編集画面上の値を取得する処理。  
GetValue

```
0075.00 C*-----連絡区分=3(メール&電話の場合)
0076.00 C LUSRKBN IFEQ '3'
0077.00 C LUSEMAL IFEQ *BLANK
0078.00 C LUSTEL OREQ *BLANK
0079.00 C MOVEL 'E0030' LERR
0080.00 C ENDIF
0081.00 C ENDIF
0082.00 C*-----エラーメッセージ取得
0083.00 C LERR IFNE *BLANK
0084.00 C LERR CHAIN MMSGR
0085.00 C *IN91 IFEQ *OFF
0086.00 /free
0087.00 //エラーメッセージを送信
0088.00 SendError (MMSGTX);
0089.00 /end-free
0090.00 C ENDIF
0091.00 C ENDIF
0092.00 p e
0093.00 **
0094.00 p ProcessDelete b
91
0125.00 p e
0126.00 /include qcpylesrc, vvNabTmpI
```

3-⑤

```
/free
//エラーメッセージを送信
SendError (MMSGTX);
/end-free
```

<<< 中略 >>>

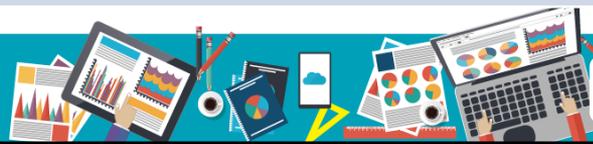
# ■ Valence App Builder RPG連携

## ● 編集： EXNABVAL

- 編集画面上の情報を GetValue にて取得。
- ロジックで編集画面に値をセットする場合、SetValueを使用。
- エラーダイアログを出力する場合は、SendErrorを使用。
- 入出力部分のAPIは、フリーフォームを使用するが、その他部分は固定長RPGでも良い。

### 主なAPI

API	戻り値	パラメータ
GetValue	取得文字列	file : ファイル名 (必須) field : フィールド名 (必須) fieldType : 属性 ("O" or "J") 指定 ("O","J"以外の時は不要)
SetValue	---	file : ファイル名 (必須) field : フィールド名 (必須) value : セットする値 (文字列) (必須)
SendError	---	msg : 出力するエラーメッセージ文字列 (必須)



# ■ Valence App Builder RPG連携

## • API一覧

### 1. ボタンクリック : EXNABBTN

API	I/O	概要
vvIn_char	I	ブラウザからのパラメータ取得
GetFormChar	I	フィールド値取得 (文字列)
GetFormNum	I	フィールド値取得 (数値)
GetSelectionChar	I	選択レコードのフィールド値取得
GetSelectionNum	I	選択レコードのフィールド値取得
GetFilterValue	I	フィルタ条件の取得
GetSqlStatement	I	関連するデータソースで実行されているSQL文の取得
vvOut_toJsonPair	O	ブラウザへ結果返却
vvOut_data	O	ブラウザへ結果返却 (フィルターウィジェット実行)

### 2. 編集 : EXNABVAL

API	I/O	概要
GetValue	I	入力値の取得
GetFilterValue	I	フィルタ条件の取得
IsChanged	I	値が変更されたかどうかの確認
SetValue	O	指定したフィールドに値をセット
SetDoNotProccess	O	指定したファイルの更新処理をスキップ
SendError	O	処理を中断しメッセージを表示

### 3. フィルタ : EXNABFLT

API	I/O	概要
GetValue	I	引数のフィルタ値の取得
SetValue	O	引数のフィールドにフィルタ値をセット
AddFilterString	O	指定されたフィルタフィールドにフィルタ追加
WriteAllFilters	O	全てのフィルタフィールドにフィルタ追加
SendError	O	処理を中断しメッセージを表示



# Valence RPG ToolKit 活用テクニック



# Valence RPG ToolKit

## RPG ToolKit

- RPGを使用して、Valenceを機能拡張するための仕組み

### RPG ToolKit の主な種類

分類	概要
vvIFS	IFS上のファイル操作
vvIn	ブラウザから情報取得
vvOut	ブラウザヘルスポンス返却 および CSV/Excel等の出力
vvMail	メール送信
vvPDF	PDFファイルの作成
vvSecure	Valence認証情報取得
vvUtility	Valenceのユーティリティ機能

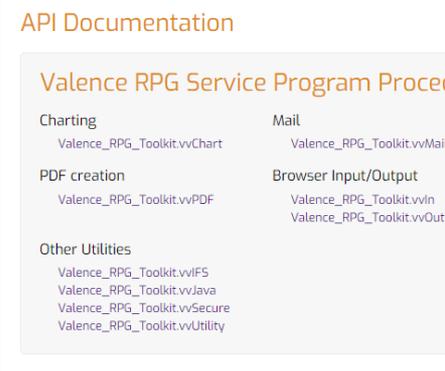
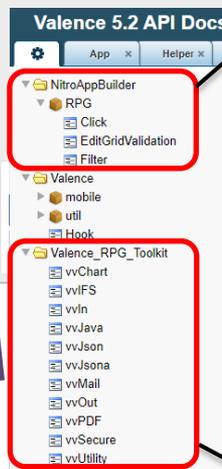
RPG ToolKit 活用テクニックとして、  
ファイルのダウンロード機能、メール送信機能 を紹介

### Valence Portal



Valence Portalにある「Valence API ドキュメント」に機能説明やサンプルが記述

App Builder RPG連携 のドキュメント (API詳細)



RPG ToolKitのドキュメント



# Valence RPG ToolKit - ファイルダウンロード

- サンプルプログラム3：PDFファイルダウンロード
  - 予めIFS上に保管しているPDFファイルをダウンロードする。

IBM i



Gridウィジェット：商品一覧より商品を選択すると、Formウィジェットに 選択した商品CDの詳細が表示。

商品一覧

商品CD ↑	商品名
A-00001	Valence App Builder
C-00001	プリントTシャツ
C-00002	アンダーシャツ
C-00003	インポートTシャツ
D-00001	柑橘のど飴

商品詳細

商品CD  
C-00001

商品名  
プリントTシャツ

単価  
2000

ダウンロード

IFS

pdfs

root > valence-5.2 > resources > pdfs

名前

- C-00001.pdf
- C-00002.pdf
- C-00003.pdf

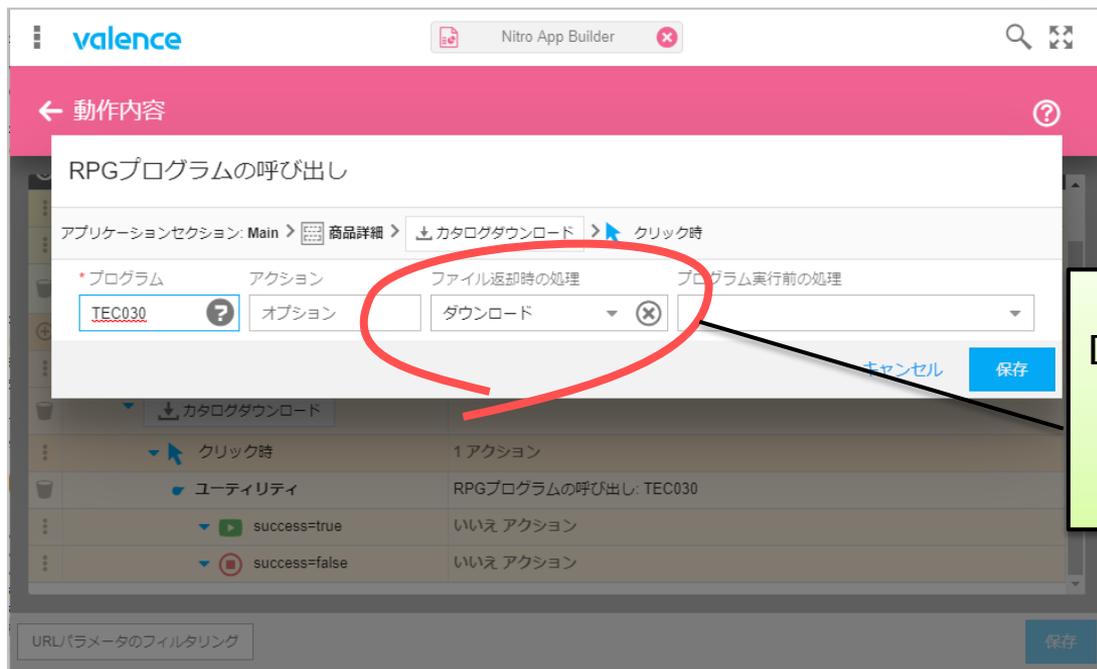
ファイル名：[商品CD].pdf というカタログファイルがIFS上に保管

ブラウザ上にファイルをダウンロード

# ■ Valence RPG ToolKit - ファイルダウンロード

## • ファイルダウンロード

- 「RPGプログラムの呼び出し」にて、[ファイル返却時の処理]に“ダウンロード”を指定する。



### [ファイル返却時の処理]

- ダウンロード : ファイルをダウンロード
- オープン : ファイルを直接開く
- プリント : ファイルを開き、印刷ダイアログ表示



# Valence RPG ToolKit - ファイルダウンロード

## TEC030: PDFダウンロード

```
0001.00 /copy qcpylesrc, vvHspec
0002.00 ** -----
0003.00 ** TEC030:PDFダウンロード
0004.00 ** -----
0005.00 d TEC030      pr
0006.00 d TEC030      pi
0007.00 /define nabButton
0008.00 /include qcpylesrc, vvNabTmp1
0009.00 ** -----
0010.00 ** program start
0011.00 ** -----
0012.00 /free
0013.00   Initialize();
0014.00   Process();
0015.00   CleanUp();
0016.00   *inlr=*on;
0017.00 /end-free
0018.00 ** -----
0019.00 p Process      b
0020.00 d              pi
0021.00 D SYHNC D      S      10A
0022.00 D ROOTPATH    S      20A
0023.00 D PDFPATH     S      64A
0024.00 D FILENAME    S      14A
0025.00 D*
0026.00 /free
0027.00 //フォーム上の値を取得
0028.00 SYHNC D = GetFormChar(' F1_SYHNC D '); //商品CD
```

4-②

IFS上にPDFファイルが存在するかを確認 (**vvIfs**)

4-①

PDFファイルが保管されているIFS上のパスを取得 (**vvUtility**)

```
0029.00
0030.00 //PDF保存先PATH取得
0031.00 ROOTPATH = vvUtility_getValenceSetting(' ROOT_PATH');
0032.00 PDFPATH = %trim(ROOTPATH) + 'resources/pdfs/';
0033.00
0034.00 //PDFファイル名取得
0035.00 FILENAME = %trim(SYHNC D) + '.pdf';
0036.00
0037.00 //PDFファイルの存在チェック
0038.00 if not vvIfs_pathExists(%trim(PDFPATH) + FILENAME);
0039.00 //ファイルが存在しない場合エラー
0040.00   vvOut_toJsonPair(' success:false, msg:PDFが存在しません。');
0041.00 else;
0042.00 //PDFファイルダウンロード
0043.00 vvOut.download = '1';
0044.00 vvOut.file = FILENAME;
0045.00 vvOut.file(%trim(PDFPATH) + FILENAME:vvOut);
0046.00 endif;
0047.00 /end-free
0048.00 p              e
0049.00 /include qcpylesrc, vvNabTmp1
```

4-①

4-②

4-③

4-③

PDFファイルのダウンロード (**vvOut**)

# ■ Valence RPG ToolKit - ファイルダウンロード

- ファイルダウンロード： vvOut\_file
  - ブラウザでファイルをダウンロード あるいは IFS上にファイルを保管する処理。
  - ファイル出力条件を vvOut パラメータに指定

## vvOut

API	戻り値	パラメータ
vvOut_file	---	filePath : 出力対象ファイルをフルパスで指定 (必須) vvOut : ファイル出力条件
vvout	---	download : '1'-ダウンロード, 'F'-IFS上へ出力 file : 出力するファイル名

## TEC030で使用した他のAPI

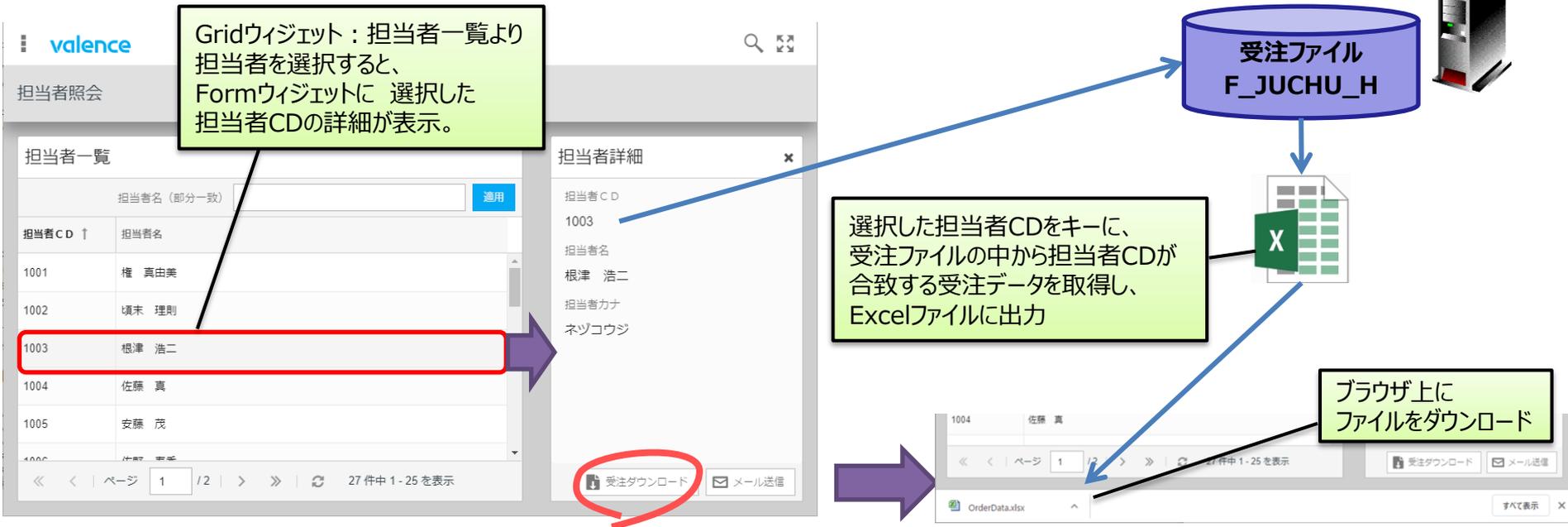
分類	API	戻り値	パラメータ
vvUtility	vvUtility_getValenceSetting	設定値	valenceSetting : Valence設定にて保管された設定値のキーを指定 (必須) キー一覧: VVSETTINGS
vvIfs	vvIfs_pathExists	*ON:有 *OFF:無	IFSPath : 存在をチェックしたいフルパスファイル名



# Valence RPG ToolKit - ファイルダウンロード

- サンプルプログラム4：Excelファイルダウンロード
  - 条件に合致するデータを抽出し、動的にExcelファイルを作成する。

IBM i



# ■ Valence RPG ToolKit - ファイルダウンロード

## • TEC040: 受注データExcelダウンロード

```
0001.00 /copy qcpylesrc, vvHspec
0002.00 ** -----
0003.00 ** TEC040: 受注データダウンロード
0004.00 ** -----
0005.00 d TEC040 pr
0006.00 d TEC040 pi
0007.00 /define nabButton
0008.00 /include qcpylesrc, vvNabTmp1
0009.00 ** -----
0010.00 ** program start
0011.00 ** -----
0012.00 /free
0013.00 Initialize();
0014.00 Process();
0015.00 CleanUp();
0016.00 *inlr=*on;
0017.00 /end-free
0018.00 ** -----
0019.00 p Process b
0020.00 d pi
0021.00 D TANTOCD S 4A
0022.00 D SQLSTR S 32766A
0023.00 D*
```

5-①

受注ファイル(F\_JUCHU\_H)から  
担当者コードが合致するデータを  
抽出するSQL文字列

```
0024.00 /free
0025.00 //フォーム上の値を取得
0026.00 TANTOCD = GetFormChar('F1_T_TANTO_CD'); //担当者CD
0027.00
0028.00 //SQL抽出条件
0029.00 SQLSTR = 'SELECT * FROM F_JUCHU_H'
0030.00 + ' WHERE H_TANTO_CD = ''' + TANTOCD + '''
0031.00 + ' ORDER BY H_JUCHU_NO';
0032.00
0033.00 //SQL抽出結果をエクセルダウンロード
0034.00 vvOut.download = '1';
0035.00 vvOut.file = 'OrderData.xlsx';
0036.00 vvOut.execSqlToSS(vvOut:SQLSTR);
0037.00 /end-free
0038.00 p e
0039.00 /include qcpylesrc, vvNabTmp1
```

5-①

5-②

5-②

Excelファイルのダウンロード (vvOut)



# ■ Valence RPG ToolKit - ファイルダウンロード

- Excelデータダウンロード： vvOut\_execSqltoSS

- SQLを実行した結果のデータをExcelファイルに出力。
- Excelファイル出力条件を vvOut パラメータに指定。

## vvOut

API	戻り値	パラメータ
vvOut_execSqltoSS	---	vvOut : Excelファイル出力条件 (必須) SQLstatement : データ抽出の為のSQL文 (必須) columnsPointer : 列情報のポインタ(オプション) numberOfColumns : 列情報指定時の列数 (オプション)

- CSVファイルを出力するAPIも用意されている。
- ファイルをREADしたデータをデータ構造体(DS)の配列へセットし、ファイル出力することも可能。

ファイル出力API	API	概要
	vvOut_execSQLtoCSV	SQLに合致するデータをCSVファイルに出力
	vvOut_toSS	DSの配列にセットされたデータをExcelファイルに出力
	vvOut_toCSV	DSの配列にセットされたデータをCSVファイルに出力



# ■ Valence RPG ToolKit - ファイルダウンロード

## • 出力結果のExcelファイル

H_JUCHU_NO	H_JUCHUBI	H_TANTO_CD	H_KKYAK_CD	H_JUCHUKIN	H_YOTEIBI	H_SYUKABI	H_SYUKABIN	H_SYUKA_NO
10,000,077	2018-03-26	1003	100020	4,408,230	2018-03-27	1900-01-01		
10,000,120	2018-04-21	1003	100006	780	2018-04-23	1900-01-01		
10,000,158	2018-07-22	1003	100017	106,800	2018-07-23	2018-07-25	郵便小包	1569143347
10,000,175	2018-06-06	1003	100012	1,445,500	2018-06-12	2018-06-06	FEDDEX	1656110686
10,000,183	2018-05-06	1003	100026	183,400	2018-05-08	1900-01-01		
10,000,233	2018-07-29	1003	100014	3,044,000	2018-08-05	2018-08-02	郵便小包	1925618472
10,000,270	2018-06-01	1003	100005	503,840	2018-06-07	2018-06-05	長谷川運輸	1645224477
10,000,300	2018-08-09	1003	100025	301,880	2018-08-10	1900-01-01		
10,000,302	2018-04-23	1003	100016	2,689,470	2018-04-27	2018-04-29	長谷川運輸	1506412851
10,000,346	2018-06-15	1003	100011	702,930	2018-06-20	2018-06-22	郵便小包	1606839667
10,000,381	2018-07-09	1003	100020	366,412	2018-07-12	2018-07-12	FEDDEX	1217493302
10,000,408	2018-05-12	1003	100009	1,692,200	2018-05-14	1900-01-01		
10,000,409	2018-08-06	1003	100025	446,390	2018-08-10	2018-08-11	郵便小包	1854163697
10,000,432	2018-05-15	1003	100008	852,500	2018-05-16	1900-01-01		
10,000,465	2018-07-15	1003	100020	54,710	2018-07-18	2018-07-20	長谷川運輸	1904991596
10,000,525	2018-04-01	1003	100026	3,601,456	2018-04-07	1900-01-01		
10,000,540	2018-04-19	1003	100011	4,000,940	2018-04-20	2018-04-25	郵便小包	1141427355
10,000,572	2018-06-20	1003	100005	474,735	2018-06-27	1900-01-01		
10,000,595	2018-03-21	1003	100006	623,600	2018-03-22	2018-03-24	田川急便	1217850569

SQLで抽出したフィールドがすべて出力されている。  
また、行タイトルは、フィールドIDがそのまま出力されている。

vvOut\_execSqltoSSでは、出力データの列情報を加工することも可能。

# Valence RPG ToolKit - ファイルダウンロード

## • TEC040 (処理追加)

```
0001.00 /copy qcpylesrc, vvHspec
0002.00 ** -----
0003.00 ** TEC040:受注データダウンロード
0004.00 ** -----
0005.00 d TEC040      pr
0006.00 d TEC040      pi
0007.00 /define nabButton
0008.00 /include qcpylesrc, vvNabTpl
0009.00 ** -----
0010.00 ** program start
0011.00 ** -----
0012.00 /free
0013.00   Initialize();
0014.00   Process();
0015.00   Cleanup();
0016.00   *inlr=*on;
0017.00 /end-free
0018.00 ** -----
0019.00 p Process      b
0020.00 d              pi
0021.00 D TANTOCD      S          4A
0022.00 D SQLSTR       S          32766A
0023.00 D COL          DS          LIKEDS (vvSSCoI)
0024.00 D              DIM(5) INZ
0025.00 U* -----
0026.00 /free
0027.00 //フォーム上の値を取得
0028.00 TANTOCD = GetFormChar ('F1_T_TANTO_CD'); //担当者CD
```

6-①  
列定義(vvSSCoI)の配列を定義  
(出力する列数分配列を定義)

6-①

6-②  
出力する列の定義を行う  
(フィールド名およびタイトルを指定)

```
0029.00
0030.00 //SQL抽出条件
0031.00 SQLSTR = 'SELECT * FROM F_JUCHU_H '
0032.00         + ' WHERE H_TANTO_CD = ''' + TANTOCD + ''' '
0033.00         + ' ORDER BY H_JUCHU_NO';
0034.00
0035.00 //表示列および列タイトル指定
0036.00 COL (1). SQLName = 'H_JUCHU_NO';
0037.00 COL (1). heading = '受注NO';
0038.00 COL (2). SQLName = 'H_JUCHUBI';
0039.00 COL (2). heading = '受注日';
0040.00 COL (3). SQLName = 'H_KKYAK_CD';
0041.00 COL (3). heading = '顧客CD';
0042.00 COL (4). SQLName = 'H_JUCHUKIN';
0043.00 COL (4). heading = '受注金額';
0044.00 COL (5). SQLName = 'H_TANTO_CD';
0045.00 COL (5). heading = '担当者CD';
0046.00
0047.00 //SQL抽出結果をエクセルダウンロード
0048.00 vvOut.download = '1';
0049.00 vvOut.file      = 'OrderData.xlsx';
0050.00 vvOut_execSqlToSS (vvOut:SQLSTR:%addr(COL):%Iem(COL));
0051.00 /end-free
0052.00 p              e
0053.00 /include qcpylesrc, vvNabTpl
```

6-②

6-③  
列定義配列のポインタと要素数を  
パラメータ指定してExcel出力

6-③



# ■ Valence RPG ToolKit - ファイルダウンロード

- Excel列定義： vvSSCol
  - 出力するExcelファイルの列定義を行う。

vvSSCol	API	戻り値	パラメータ
	vvSSCol	---	SQLName : 出力するフィールド名 heading : 列タイトル colWidth : 列の幅 omit : '1'-非表示列 指定 styleID : 出力書式

- 出力結果のExcelファイル

OrderData.xlsx - Microsoft Excel

	A	B	C	D	E	F	G	H
	受注NO	受注日	顧客CD	受注金額	担当者CD			
1								
2	10,000,077	2018-03-26	100020	4,408,230	1003			
3	10,000,120	2018-04-21	100006	780	1003			
4	10,000,158	2018-07-22	100017	106,800	1003			
5	10,000,175	2018-06-06	100012	1,445,500	1003			
6	10,000,183	2018-05-06	100026	183,400	1003			
7	10,000,233	2018-07-29	100014	3,044,000	1003			
8	10,000,270	2018-06-01	100005	503,840	1003			

プログラムで指定したフィールドのみ出力  
列タイトルも日本語で出力されている

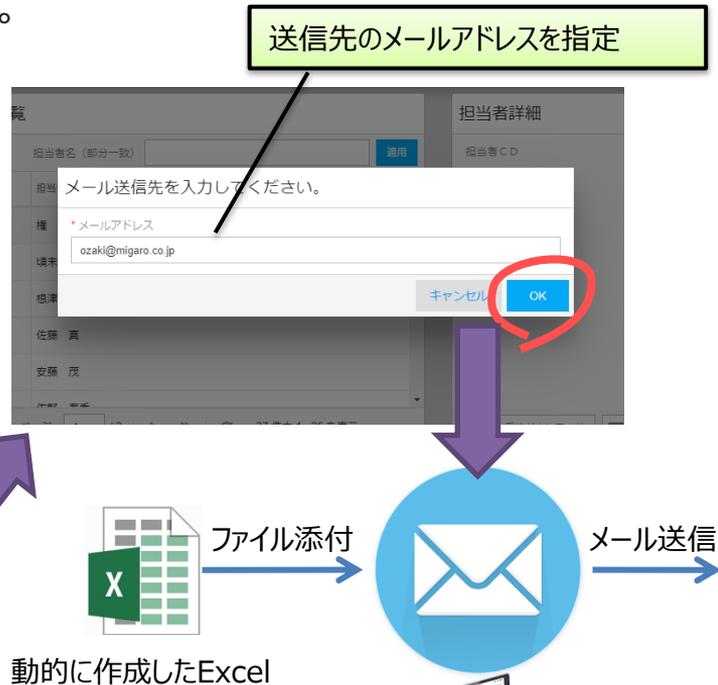
# ■ Valence RPG ToolKit – メール送信

- サンプルプログラム5：メール送信
  - 動的に作成したExcelファイルを添付したメールを送信する。

The screenshot shows the Valence RPG ToolKit interface. On the left, there is a table titled "担当者一覧" (Assignee List) with columns for "担当者C/D" and "担当者名". The table contains the following data:

担当者C/D	担当者名
1001	権 真由美
1002	頃末 理則
1003	根津 浩二
1004	佐藤 真
1005	安藤 茂

On the right, there is a "担当者詳細" (Assignee Details) modal showing details for "担当者C/D 1003", "担当者名 根津 浩二", and "担当者カナ ネットコウジ". At the bottom of the interface, there is a "メール送信" (Send Email) button, which is circled in red. A purple arrow points from this button to a modal window on the right.



# ■ Valence RPG ToolKit – メール送信

- 実行時にパラメータを追加

- 「RPGプログラムの呼び出し」にて、[プログラム実行前の処理] を追加する。

← 動作内容

RPGプログラムの呼び出し

アプリケーションセクション: Main > 担当者詳細 > メール送信 > クリック時

プログラム	アクション	ファイル返却時の処理	プログラム実行前の処理
TEC050	オプション		Prompt User For More Information

パラメータ入力画面のタイトル

メール送信先を入力してください。

ラベル	パラメーター名	タイプ
メールアドレス	MAILTO	Email

必須 \* 長さ

50

キャンセル OK

## Propmt User For More Infomation

ボタンクリック時に、追加情報を入力するプロンプト画面を開く設定

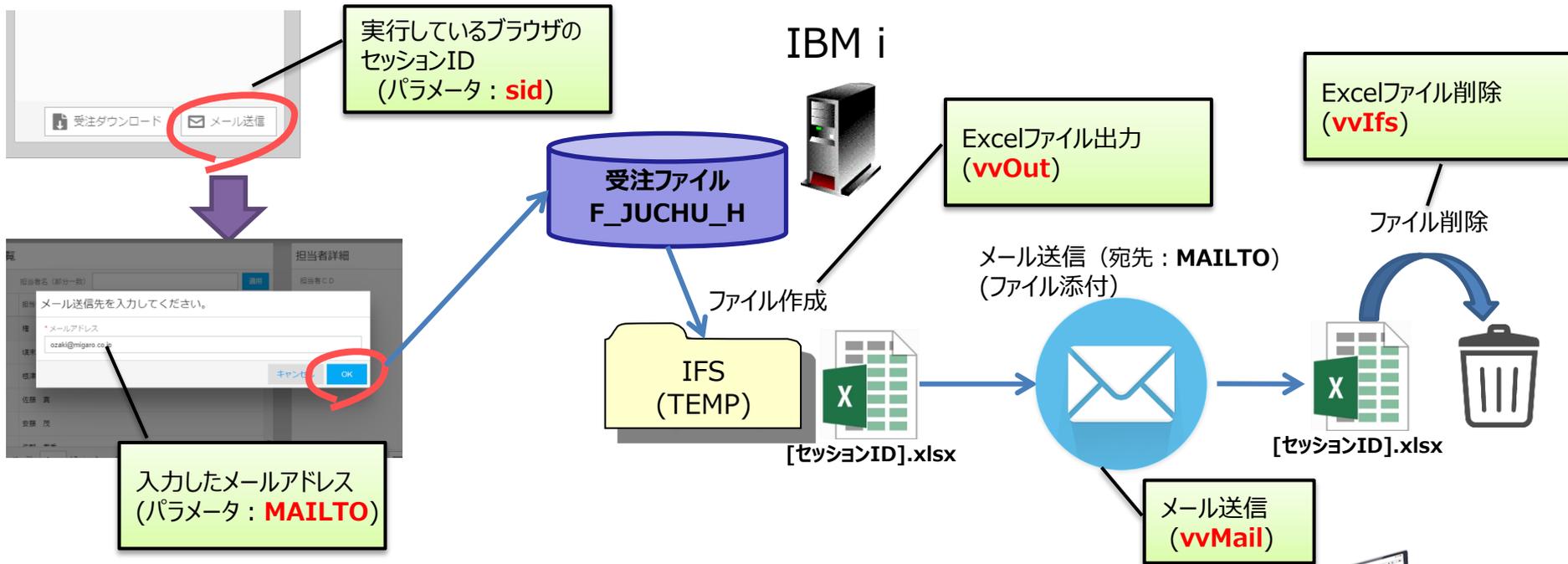
## メールアドレス 入力欄を表示

パラメータ名 : MAILTO  
タイプ : Email 形式  
桁数 : 50桁

# Valence RPG ToolKit – メール送信

## 処理概要

- IFS上に一意な名称を持つExcelを作成し、それを添付したメールを送信。処理後Excelは削除。



# Valence RPG ToolKit – メール送信

## • TEC050 : メール送信

```
0001.00 /copy qcpylesrc, vvHspec
0002.00 **
0003.00 ** TEC050:メール送信
0004.00 **
0005.00 d TEC050      pr
0006.00 d TEC050      pi
0007.00 /define nabButton
0008.00 /define includeEMAIL
0009.00 /include qcpylesrc, vvNabImpl
0010.00 **
0011.00 ** program start
0012.00 **
0013.00 /free
0014.00   Initialize();
0015.00   Process();
0016.00   CleanUp();
0017.00   *inlr=*on;
0018.00 /end-free
0019.00 **
0020.00 p Process      b
0021.00 d              pi
0022.00 D TANTOCD      S          4A
0023.00 D MAILTO       S          50A
0024.00 D TMPPATH      S          20A
0025.00 D SID          S          64A
0026.00 D SQLSTR       S          32766A
0027.00 D*
```

7-①

メール送信用API (**vvMail**)  
使用時は、定義を追加。

7-①

```
/define includeEMAIL
```

7-②

パラメータ(MAILTO)の値 および  
セッションID(sid)の取得 (**vvIn**)

```
0028.00 /free
0029.00 //フォーム上の値を取得
0030.00 TANTOCD = GetFormChar('F1_T_TANTO_CD'); //担当者CD
0031.00 MAILTO = vvIn_char('MAILTO'); //送信アドレス
0032.00 SID = vvIn_char('sid'); //セッションID
0033.00
0034.00 //ファイル保存先取得
0035.00 TMPPATH = vvUtility_getValenceSetting('TEMP_PATH');
0036.00
0037.00 //SQL抽出条件
0038.00 SQLSTR = 'SELECT * FROM F_JUCHU_H'
0039.00         + ' WHERE H_TANTO_CD = ''' + TANTOCD + '''
0040.00         + ' ORDER BY H_JUCHU_NO';
0041.00
0042.00 //SQL抽出結果をIFS上に出力
0043.00 vvOut.download = 'F';
0044.00 vvOut.file = %trim(TMPPATH) + SID + '.xlsx';
0045.00 vvOut_execSqlToSS(vvOut:SQLSTR);
```

7-②

7-③

7-③

IFS上のValenceのTempフォルダ内に、  
ファイル名：“[セッションID].xlsx”で、Excelファイルを出力 (**vvOut**)



# Valence RPG ToolKit – メール送信

## • TEC050 : メール送信

```
0047.00 //メール内容作成
0048.00 vvMail.from      = 'info@migaro.co.jp';
0049.00 vvMail.to        = MAILTO;
0050.00 vvMail.subject    = '担当者別受注一覧報告';
0051.00 vvMail.body       = '担当者CD=' + TANTOCD + '<br>'
0052.00                 + 'の受注一覧を報告いたします。';
0053.00 vvMail.attachment = vvOut.file;
0054.00 vvMail.attachAlias = 'OrderData.xlsx';
0055.00
0056.00 //メール送信
0057.00 if not vvMail_send(vvMail);
0058.00     vvOut_toJsonPair('success:false,msg:送信エラー');
0059.00 else;
0060.00     vvOut_toJsonPair('success:true,info:送信しました');
0061.00 endif;
0062.00
0063.00 //出力ファイルを削除
0064.00 vvIfs_deleteFile(vvOut.file);
0065.00
0066.00 /end-free
0067.00 p           e
0068.00 /include qcpylesrc, vvNabTmpI
```

7-④

7-④

メール送信パラメータ (**vvMail**) に  
送信元、送信先、タイトル、本文、添付ファイルを指定

7-⑤

7-⑤

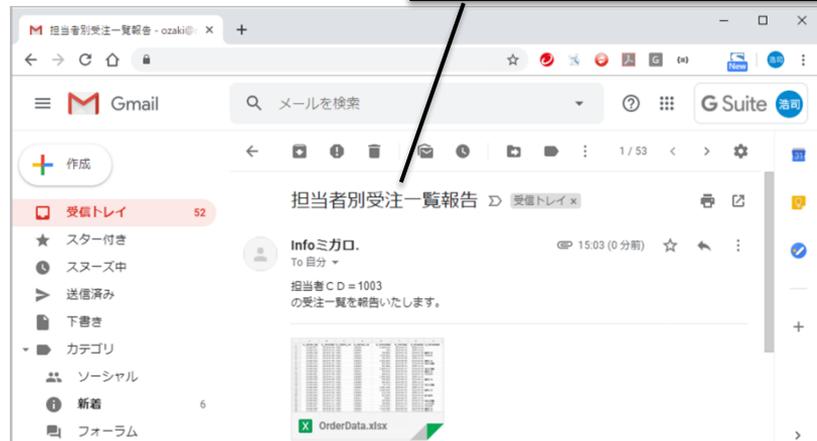
メール送信処理 (**vvMail\_send**)

7-⑥

7-⑥

メール送信終了後、出力したExcelファイルを削除 (**vvIfs**)

例 : Gmailで受信したメール



# ■ Valence RPG ToolKit – メール送信

- メール送信： `vvMail_send`
  - `vvMail`パラメータに指定された内容を元にメールを送信。
  - `vvMail`を使用する場合、はじめに"/define includeEMAIL"を追加。
  - メールサーバー（SMTP）の設定は、Valenceの[ポータル管理]→[設定]で行う。

## vvMail

API	戻り値	パラメータ
<code>vvMail_send</code>	*ON:成功 *OFF:エラー	<code>vvMail</code> : メール送信内容（必須）
<code>vvMail</code>	---	<code>from</code> : 送信元アドレス <code>to</code> : 送信先アドレス <code>cc</code> : ccアドレス <code>bcc</code> : bccアドレス <code>subject</code> : 件名 <code>body</code> : 本文 <code>attachment</code> : 添付ファイル（IFS上のフルパスファイル名） <code>attachAlias</code> : メールに表示される添付ファイル名



# ■ Valence RPG ToolKit – メール送信

## • TEC050 : その他API

- vvIn\_char : ブラウザからの入力情報(変数)を取得  
(変数 : **sid** は、実行しているブラウザのセッションIDを示す)
- vvIfs\_deleteFile : IFS上のファイルを削除

### TEC050で使用した他のAPI

分類	API	戻り値	パラメータ
vvIn	vvIn_char	取得値	variableName : 変数名 (必須) formatOption : TOUPPER(英大文字に変換) / TOLOWER(英小文字に変換)
vvIfs	vvIfs_deleteFile	0:成功 -1:エラー	path : 削除したいファイルのフルパスファイル名 (必須)



# 新機能 Fusion5250



# ■ 新機能 Fusion5250

## • Fusion5250

- Valence上で稼働する5250エミュレータ ユーティリティ
- クライアントアクセス、ACS等と同様の5250エミュレータがValence上で実行可能。ブラウザのみで、エミュレータが動作する為、場所を選ばず利用できる。

The image displays the Valence web interface. On the left, a 'ツール' (Tools) section contains icons for Nitro File Editor, Nitro Source Editor, Nitro AutoCode, and Fusion5250. The Fusion5250 icon is highlighted with a red box, and a purple arrow points from it to the right. On the right, two overlapping browser windows show the Fusion5250 application. The top window is in 'グリーンスクリーンモード' (Green Screen Mode), displaying a main menu with 11 numbered options in green text on a black background. The bottom window is in 'Valenceモード' (Valence Mode), displaying the same menu on a white background. A yellow callout box labeled 'Valenceモード' points to the bottom window. A green callout box labeled 'Fusion 5250 {JSON}' is positioned above the bottom window. A green callout box at the bottom left explains that the Fusion5250 tool runs in the browser, emulating the application's interface and allowing mode switching.

Valenceが実行されるブラウザ上にエミュレータ同様の画面が起動 (グリーンスクリーンモードとValenceモードの切替が可能)

グリーンスクリーンモード

Fusion 5250 {JSON}

Valenceモード

# ■ 新機能 Fusion5250

## • Fustion5250 マクロ機能

- エミュレータ上の操作（コマンド入力）をマクロとして登録可能。
- 多用するPC5250コマンドや、既存のPC5250アプリを直接呼び出すことができる。

マクロを記述したJSONファイルを  
(IFS) Valence-5.2/resources/fusionmacros/  
に保管する

```
MIGAROAPP1.json
1 {
2   "showProgress":false,
3   "steps":[
4     {
5       "col":"7",
6       "row":"20",
7       "type":"CALL CNXLIB/MIGAROAPP1"
8     },
9     {
10      "action":"ENTER",
11      "control":false,
12      "shift":false
13    }
14  ]
15 }
```

記述例：【ステップ1】  
7列目20行目（コマンド入力欄）  
にて、“CALL CNXLIB/MIGAROAPP1”と入力

記述例：【ステップ2】  
[実行キー]を押下

JSONファイル



Valence メニュー設定にて、  
Fusion5250を登録する際に  
保管したマクロを指定すればよい。

Valence Portalから、既存の  
PC5250アプリをシームレスに  
連携することが可能。

# Valence5.2+ 新機能

- Valence App BuilderとFusion5250との連携
  - App Builderの「動作内容」よりFusion5250を呼び出すアクションが指定可能。
  - データソースに定義されたフィールドをパラメータで受け渡すことが可能。



Valence App Builder  
アプリケーション作成：動作内容  
設定画面

マクロファイル名 および  
パラメータ (**param1 : 社員コード**) を指定

アプリケーションの実行

アプリケーションセクション: Main > CDMSY\_Title > 社員一覧タイトル > クリック時

\*アプリケーション

Fusion5250

任意のアプリケーションパラメータ

macro=MIGARO\_OZAKI&param1={F1\_SYSYCD}

F1_SYSYCD	社員コード
F1_SYSYKN	氏名 (カナ)
F1_SYSYKJ	氏名 (漢字)
F1_SYSYKB	社員区分

キャンセル 保存



# Valence5.2+ 新機能

## Valence App BuilderとFusion5250との連携

### 実行イメージ

Widgetに一覧表示された社員一覧から、社員をクリックして選択

App Builder  
で作成したアプリ

選択した社員コードの PC5250 入力画面を表示

1回目		2回目				
適用数	居住開始日	区分	借入金等の額	居住開始日	区分	借入金等の額
C	00000000		0	00000000		C

他の支払者	
住所	名称
	国内外 C 他の支払者のもとを退職した日 00000000
その他	

未払金額	災害者に係る	青色専従者条約免除
C	徴収猶予税額	C 0

F1= ヘルプ F3= 終了 F12= 前画面

Fusion5250  
既存の5250アプリ

既存の5250アプリをそのまま活用し、App Builderアプリとのシームレスな連携を実現



ご清聴ありがとうございました。

