

附録 B JCL4D コンポーネント

JCL4D とは
コンポーネントを組み込む
メソッド概要

JCL4D とは

「JCL4D」は Jirokichi Common Library for 4D の略。プロジェクトファイルで提供される当社製オープンソースライブラリである。

治郎吉商店では、4D アプリ開発の生産性を高める努力をしている。その一つが共通メソッド化だ。プロジェクトの中でよく使うコードを共通メソッドとして抽出し、その中から別プロジェクトでも使えそうなメソッドをコンポーネント化した。4D コマンドを一つずつ実行するよりも処理単位が大きくなるため、簡単でわかりやすくなる。

これらのメソッド群は「JCL4D.4dbase」というひとつのプロジェクトファイルにまとめてある。JCL4D を開発アプリのコンポーネントフォルダに組み込んで使用する。

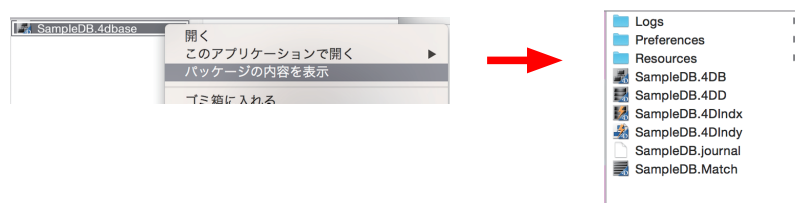
コンポーネントを組み込む

ここではコンポーネントを組み込む方法について説明する。

操作手順：

1. 4D を終了して、プロジェクトフォルダを開く

Mac OS の場合はプロジェクトファイルを選択して右クリックで【パッケージの内容を表示】を実行。



Windows の場合はプロジェクトフォルダを開く。

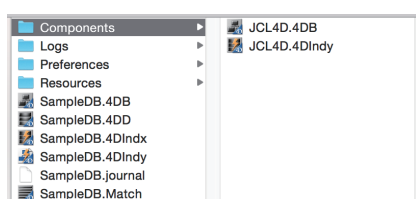
2. 「Components」フォルダを作成

OS の【新規フォルダ】を実行して、「Components」という名称のフォルダを作成。

名前	変更日	サイズ	種類
Components	2015/09/11 20:57	--	フォルダ
Logs	2015/08/30 18:49	--	フォルダ
Preferences	2015/08/30 18:49	--	フォルダ
Resources	2015/09/06 17:58	--	フォルダ
SampleDB.4DB	今日 9:58	459 KB	4D Int...ure File
SampleDB.4DD	今日 9:57	394 KB	4D Data File
SampleDB.4DIndx	今日 9:58	197 KB	4D data indexes
SampleDB.4DIndy	今日 9:58	197 KB	4D str...ndexes
SampleDB.journal	今日 9:58	4 KB	4D Journal File
SampleDB.Match	今日 9:57	212 バイト	4D str...re info

3. Components フォルダに「JCL4D.4DB」ファイルと「JCL4D.Indy」ファイルをコピー

ストラクチャファイルとストラクチャのインデックスファイルをコピーして、Components フォルダにペースト。



これでプロジェクトファイル「JCL4D.4DB」をプロジェクトファイル「SampleDB.4dbase」のコンポーネントとして組み込むことができた。元のプロジェクトファイル「JCL4D.4DB」をマトリックスデータベース、組み込む先のプロジェクトファイル「SampleDB.4dbase」をホストデータベース、組み込み後（Components フォルダに入っている状態）の「JCL4D.4DB」をコンポーネントデータベースと呼ぶ。

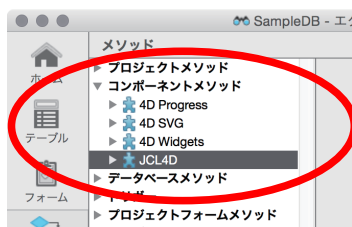
SampleDB.4dbase を開いて組み込み後の状態を確認してみよう

4. 「SampleDB.4dbase」を 4D で開く

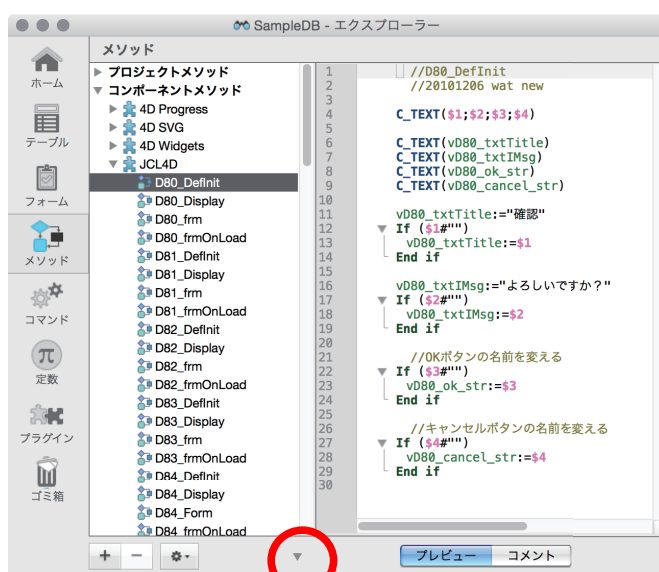
5. [デザインモードへ] をクリックして、【デザインモードへ】メニューを実行

6. エクスプローラを開く

「メソッド」を表示すると、「コンポーネントメソッド」という項目が追加されている。展開すると「JCL4D」の内容が読み込まれていることが確認できる。



さらに JCL4D を展開すると、利用可能なメソッドが一覧表示される。



図のように、▼マークをクリックすると、メソッドの中身が表示される。コンポーネントメソッドは開いて編集することはできないが、このようにして中身を見ることはできる。

メソッド概要

ここでは、SampleDB で使っている JCL4D メソッドについてその機能を説明する。

JCL4D メソッドの名前は、頭文字「JCL_」で始まり、次にコマンドグループを示す文字が続く。たとえば日付文字列を生成するメソッド「JCL_str_CreateDatemark」の場合、JCL_ の次の str_ がストリング、すなわち文字列を扱うメソッド群の仲間であることを示している。

次のようなメソッド群がある。

str	文字列メソッド	文字列を扱うためのメソッド群。
lst	リストボックスメソッド	リストボックス用のコマンド。
btn	ボタンメソッド	ボタンオブジェクトを制御するためのメソッド群。
dlg	ダイアログメソッド	カスタムダイアログボックスを表示するためのメソッド群。
file	ファイルメソッド	外部ファイルを読んだり作成したりするメソッド群。
prt	プリントメソッド	印刷の際に便利なメソッド群。

文字列メソッド

JCL_str_Datemark

デイトマーク文字列を生成する。

```
$datemarkStr:= JCL_str_Datemark($inDate;$inTime)
```

引数

inDate 日付を日付型で与える。

inTime 時刻を時刻型で与える。

説明

日付「inDate」と時刻「inTime」を文字列に変換して「20150815091059000」のような文字列を作成する。

リストボックスメソッド

JCL_lst_SelectedCount

リストボックス上で何行選択されているかを調べるメソッド。

```
$selectedCount:= JCL_lst_SelectedCount(->inListBoxPtr)
```

引数

->inListBoxPtr リストボックスの配列変数をポインタ型で与える。

戻り値

リストボックスで選択されている行数が返る。

説明

リストボックスの選択状態は、リストボックスに割り当てられている Boolean の配列変数に保持されている。選択されている行は True、選択されていなければ False が格納されている。

このメソッドは、この配列変数で True になっている要素を合計したものを選択行数として返す。

JCL_lst_Selected_Long

リストボックスで選択されている行の Longint の値を取得する。

```
$st_id:=JCL_lst_Selected_Long (->inListBoxPtr;->inLongAryPtr)
```

引数

->inListBoxPtr リストボックスの配列変数をポインタ型で与える。

->inLongAryPtr Longint 型の配列をポインタ型で与える。通常は、リストボックスに割り当てている配列変数を与える。

戻り値

Longint 型の数値を返す。

説明

リストボックス「inListBoxPtr」で始めに見つかった選択行の行番号を取得して、その行番号に対応する Longint 配列の要素の値を返す。

JCL_lst_SetSelect_byLong

プログラムを使って、リストボックスの行を選択状態にする。

JCL_lst_SetSelect_byLong (->inListBoxPtr;->inLongAryPtr;\$id)

引数

- >inListBoxPtr リストボックスの配列変数をポインタ型で与える。
- >inLongAryPtr Longint 型の配列をポインタ型で与える。
- \$id キー値です。選択対象の ID を与える。Longint 型の数値。

説明

リストボックスに連結されているロングの配列とキー値を比較して、一致した要素があれば、その行を選択状態にする。引数「\$st_id」がゼロの場合は選択なしになる。

ボタンメソッド

JCL_btn_SetVisible

ボタンを表示にしたり、非表示にする。

JCL_btn_SetVisible (->inButtonPtr;\$bool) ` ボタン非表示

引数

- >inButtonPtr 対象となるボタンオブジェクトを変数のポインタで与える。
- \$bool True または False を与える。

説明

「inButtonPtr」のボタンを「\$bool」で表示にしたり非表示にする。

JCL_btn_SetEnable_byListSelect

リストボックスの選択行がひとつの場合にだけボタンをイネイブルにする。

JCL_btn_SetEnable_byListSelect (->inListBoxPtr;->inButtonPtr)

引数

- >inListBoxPtr リストボックスの配列変数をポインタ型で与える。
- >inButtonPtr 対象となるボタンオブジェクトを変数のポインタで与える。

説明

第 1 引数のリストボックスに選択行がひとつだけの場合、第 2 引数のボタンをイネイブル（クリック可能なアクティブな状態）にする。選択行がないか複数あった場合、ボタンはディスエイブル（クリックできないグレイ表示の状態）にする。

ダイアログメソッド

JCL_dlg_NoYes

デフォルトがキャンセルの確認ダイアログを表示。Return キーはキャンセル、Escape キーは OK のショートカットキーが割り当てられている。

```
$dialog_ok:=JCL_dlg_NoYes ($title;$msg;$ok_btn_text;$cancel_btn_text)
```

引数

\$title	ダイアログのタイトル文字列。大きな文字で表示される。
\$msg	ダイアログに表示されるメッセージ本文。
\$ok_btn_text	OK ボタンのタイトル。
\$cancel_btn_text	キャンセルボタンのタイトル。

戻り値

OK ボタンが押された場合は「1」が、キャンセルボタンがクリックされた場合は「0」が返される。

説明

デフォルトボタンが「キャンセル」の確認ダイアログを表示。タイトル、メッセージ、OK ボタンテキスト、キャンセルボタンテキストの4つの文字列を与える。ボタンテキストについては、省略した場合はデフォルトの文字列が表示される。

JCL_dlg_Inform

OK ボタンだけのダイアログを表示。

```
JCL_dlg_Inform ($title;$msg;$ok_str)
```

引数

\$title	ダイアログのタイトル文字列。大きな文字で表示される。
\$msg	ダイアログに表示されるメッセージ本文。
\$ok_str	OK ボタンのタイトル。

説明

ユーザは [OK] をクリックするしかない。このダイアログは、プログラムで何かを実行している状態が終了したとき、終了したことをユーザに伝える場合などに使う。

ファイルメソッド

JCL_file_Create

ファイル保存ダイアログを表示して、ファイルを作成する。ファイル参照番号を返す。

```
$dlg_ok:=JCL_file_Create ($ext;->$doc;->$docName)
```

引数

\$ext	ファイル拡張子を与える。
-------	--------------

->\$doc ファイル参照番号をポインタで与える。この引数にファイル参照番号が代入される。

->\$docName ファイル名をポインタで与える。この引数にファイル名が代入される

戻り値

ファイル保存ダイアログが OK された場合は「1」を、キャンセルされた場合は「0」を返す。

説明

メソッドの戻り値が「1」の場合に、ファイル参照番号「\$doc」とファイル名「\$docName」を取得する。

JCL_file_Close

ファイルを閉じる。

JCL_file_CloseTextFile (\$doc)

引数

\$doc ファイル参照番号を与える。

説明

ファイル参照番号「\$doc」のファイルを閉じる。

JCL_file_WriteSJIS

文字列を Shift-JIS でテキストファイルに書き出す。

JCL_file_WriteSJIS (\$doc;\$text)

引数

\$doc ファイル参照番号を与える。

\$text 出力するテキストを与える。

戻り値

書き出しが成功すると「1」、失敗すると「0」を返す。

説明

「\$text」で与えられた文字列を「\$doc」で指定されたテキストファイルに「Shift_JIS」で書き出す。

JCL_file_WriteTab

タブコードをを書き出す。

JCL_file_WriteTab (\$doc)

引数

\$doc ファイル参照番号を与える。

説明

ファイル参照「\$doc」のファイルにタブコードをを書き出す。

JCL_file_WriteCRLF

改行コードを書き出す。

```
JCL_file_WriteCRLF ($doc)
```

引数

\$doc ファイル参照番号を与える。

説明

ファイル参照「\$doc」のファイルに「CR」と「LF」コードを書き出す。

JCL_file_Logout

デスクトップに「Logout.txt」というテキストファイルを作成して、メッセージを出力。文字コードは utf-8。追記（アペンド）モードで出力される。実行するたびに Logout.txt にメッセージが追記されていく。ログ出力をクリアする場合は、ファイル名を変更するか削除する。

```
C_TEXT($msg)
$msg:="funcA: START"
$msg:=$msg+": $sizeofAry=["+String($sizeofAry)+"]"
JCL_file_Logout ($msg)
```

引数

\$msg 出力するメッセージを文字列で与える。

戻り値

なし

説明

デスクトップに「Logout.txt」というファイルがあればファイルを開いて、最後の行にメッセージを追記する。「Logout.txt」というファイルがなければファイルを作成して、ファイルの頭からメッセージを出力する。

プリントメソッド

JCL_prt_NumOfPages

印刷されるページ数を計算する。

```
$numOfPages:=JCL_prt_NumOfPages (->inArrayPtr;$maxRowsInPage)
```

引数

->inArrayPtr 配列をポインタで与える。

\$maxRowsInPage ページ辺りの最大印刷行数。

戻り値

印刷されるページ数を返す。

説明

配列「inArrayPtr」の要素数とページ辺りの最大印刷行数から、印刷されるページ数を計算して返す。

JCL_prt_PageBreak

印刷時に改ページを出力。

JCL_prt_PageBreak (\$nombre;\$numOfPages)

引数

\$nombre 現在のページ番号。

\$numOfPages 総ページ数。

説明

総ページ数「\$numOfPages」と現在のページ数「\$nombre」を比較して、最後のページはここで終わりのページブレイク、最後のページでない場合は継続のページブレイクを出力する。