

日々是Oracle APEX

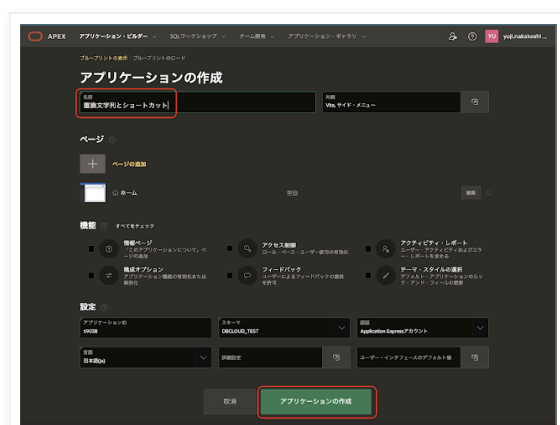
Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2020年6月19日 金曜日

Oracle APEXアプリケーションのグローバル化(9) - 置換文字列とショートカット

Oracle APEXアプリケーションを多言語に対応させるための色々な機能を紹介してきました。このトピックの最後に置換文字列とショートカットを使用したメッセージの翻訳方法を紹介します。

実用的な理解を得るために、これらの機能を使用した簡単なアプリケーションを作成します。まず、**アプリケーション・ビルダー**から**作成**をクリックし、**新規アプリケーション**の作成を行います。アプリケーション作成ウィザードでは、アプリケーションに適切な**名前**を指定し、その他はデフォルトのままで**アプリケーションの作成**を実行します。



3 ページだけの、何も機能がないアプリケーションが作成されます。



共有コンポーネントのグローバル化のセクションにある**テキスト・メッセージ**を開き、以下の4つのメッセージを登録します。

名前	言語	テキスト	JavaScriptで使用
T_HELLO	ja	こんにちは!	いいえ
T_HELLO	en	Hello!	いいえ
T_CONFIRM	ja	本当に?	はい

T_CONFIRM	en	Are you sure?	はい
-----------	----	---------------	----

以上の4つのメッセージを登録すると、以下のようにリストされます。



名前	言語	テキスト	テキストの長さ	JavaScriptで利用	所有者	日付
T_CONFIRM	ja	本当にいいえ？	6	はい	YUJUNAKAGISHI@ORACLE.COM	7月26日
T_HELLO	ja	こんにちは！	6	いいえ	YUJUNAKAGISHI@ORACLE.COM	7月26日
T_HELLO	en	Hello!	6	いいえ	YUJUNAKAGISHI@ORACLE.COM	7月26日
T_CONFIRM	en	Are you sure?	12	いいえ	YUJUNAKAGISHI@ORACLE.COM	7月26日

このようにして登録したメッセージを置換文字列、またはショートカットから利用する方法を実装していきます。

置換文字列による翻訳

登録済みのテキスト・メッセージは、**&APEX_TEXT\$メッセージ名.**という形式の置換文字列として使用することができます。上記の例をとると、以下のように置き換わります。

&APEX_TEXT\$T_HELLO. = こんにちは! (言語として日本語が選択されているとき)

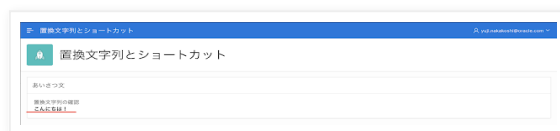
&APEX_TEXT\$T_HELLO. = Hello! (言語として英語が選択されているとき)

ホーム・ページに静的リージョンを追加して確認します。ソースのテキストとしては以下を入力します。

置換文字列の確認
&APP_TEXT\$T_HELLO.



アプリケーションを実行して結果を確認します。



アプリケーションのプライマリ言語が日本語なので、**&APP_TEXT\$T_HELLO.がこんにちは!**へ置き換わっていることが確認できます。

ショートカットによる翻訳

共有コンポーネントにショートカットというコンポーネントが含まれています。これは置換文字列と似ていますが、置き換える文字列を動的に生成することができます。その機能のひとつとして、メッセージの動的翻訳が提供されています。



ショートカットを開いて、動作を確認するために2つのショートカット、T_HELLOとT_CONFIRMを登録します。



登録可能なショートカットのタイプには以下があります。

- PL/SQLファンクション本体
- HTMLテキスト
- エスケープされた特殊文字付きHTMLテキスト
- イメージ
- JavaScriptにより一重引用符がエスケープされたテキスト
- メッセージ
- JavaScriptにより一重引用符がエスケープされたメッセージ

上記の中で、**メッセージ**とJavaScriptにより一重引用符がエスケープされたメッセージが動的な翻訳に使用できるタイプです。

名前をT_HELLO、**タイプ**をメッセージ、**ショートカット**としてT_HELLO(これはテキスト・メッセージとして登録されている名前)を指定して、ショートカットT_HELLOを作成します。



同様に、**名前**をT_CONFIRM、**タイプ**をJavaScriptにより一重引用符がエスケープされたメッセージ、**ショートカット**としてT_CONFIRMを指定して、ショートカットT_CONFIRMを作成します。



これでショートカットが登録されました。

マニュアルの記載によると、ショートカットを使用できる箇所は以下になります。

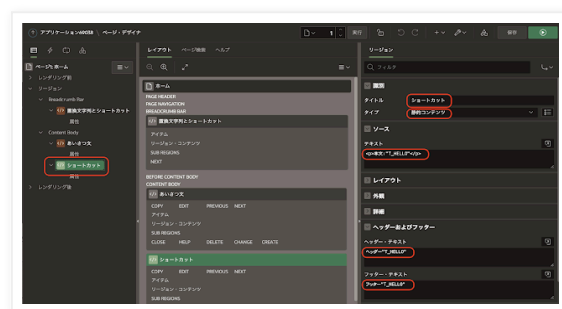
- ・「HTMLテキスト(ショートカットあり)」として定義されたリージョンのリージョン・ソース属性。
- ・リージョンのヘッダーおよびフッターのテキスト属性。
- ・アイテム・ラベル属性、要素の前のテキスト、要素の後のテキストおよびデフォルト値属性。
- ・リージョン・テンプレート属性。

ホーム・ページに**タイトル**を**ショートカット**、**タイプ**が**静的コンテンツ**であるリージョンを追加し、ショートカットによる文字列の表示を確認します。その他の情報は以下を指定します。ショートカット名は**ダブルクォーテーション**で囲みます。

ソースのテキスト：<p>本文:"T_HELLO"</p>

ヘッダー・テキスト：ヘッダー"T_HELLO"

フッター・テキスト：フッター"T_HELLO"



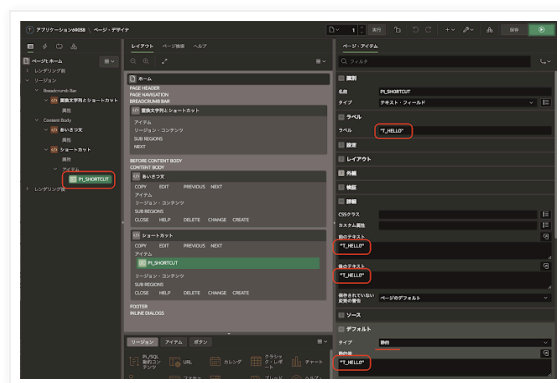
ここで、マニュアルの記載にある「HTMLテキスト(ショートカットあり)」をリージョンに定義します。リージョンの属性を開き、出力形式がHTMLであることを確認し、ショートカットを開くをONにします。



設定が完了したらアプリケーションを実行し、結果を確認します。ショートカットが日本語のメッセージに置き換わっていることが確認できます。



ページ・アイテムについてもショートカットが使えます。**P1_SHORTCUT**というテキスト・フィールドを追加して、置き換え可能な箇所にショートカットを指定してみます。**ラベル**、**前のテキスト**、**後のテキスト**、**タイプ**を静的としたデフォルト値に**"T_HELLO"**を指定します。

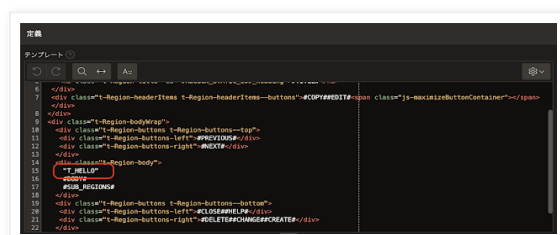


アプリケーションを実行して、表示がどうなるかを確認します。



指定可能な箇所のすべてで、ショートカットが置き換わっていることが確認できます。

リージョン・テンプレートへのショートカットの適用については実際の作業は行いません。以下のように、リージョン・テンプレートの定義に直接ショートカットを埋め込むことができます。



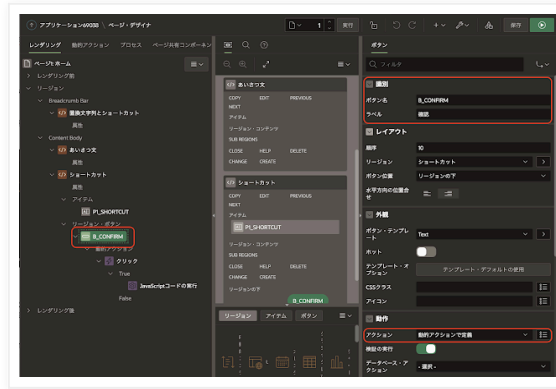
JavaScriptでのショートカットの利用は以下の手順にそって行います。

最初にショートカットをグローバル変数に設定します。ページのプロパティにある**ファンクション**および**グローバル変数の宣言**に以下を記述します。

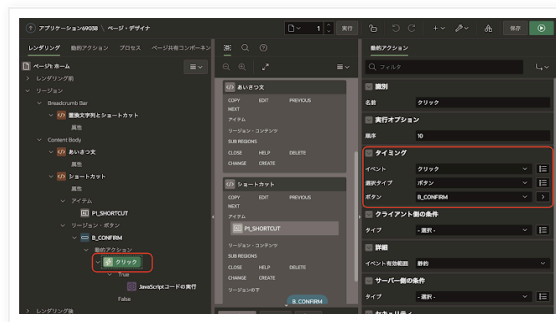
```
var msg_confirm = "T_CONFIRM";
```

JavaScriptにてmsg_confirmを表示するダイアログを開くことで設定の確認を行います。

動的アクションを実行するボタンを作成します。**アクション**は**動的アクション**で定義になります。

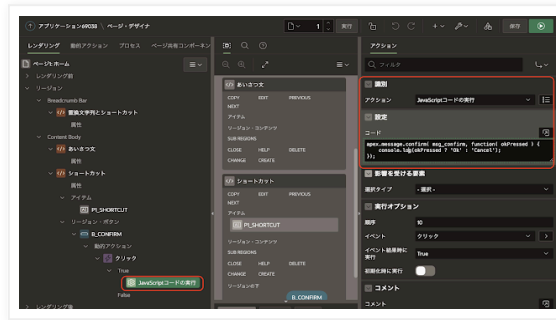


ボタンをクリックしたタイミングで、動的アクションを実行します。

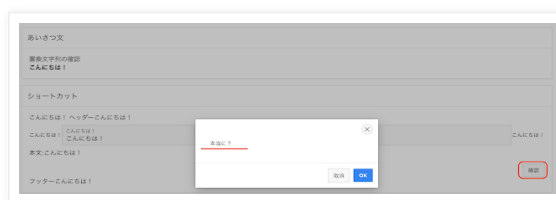


アクションとしてJavaScriptコードの実行を選び、実行するアクションのJavaScriptのコードとして以下を指定します。

```
apex.message.confirm( msg_confirm, function( okPressed ) {
    console.log(okPressed ? 'Ok' : 'Cancel');
});
```



以上で実装は完了です。アプリケーションを実行し、追加したボタンをクリックします。msg_confirm、元々はT_CONFIRMとして定義されたショートカット(そのショートカットはT_CONFIRMとして定義された翻訳済みのテキスト・メッセージ)が表示されていることが確認できます。

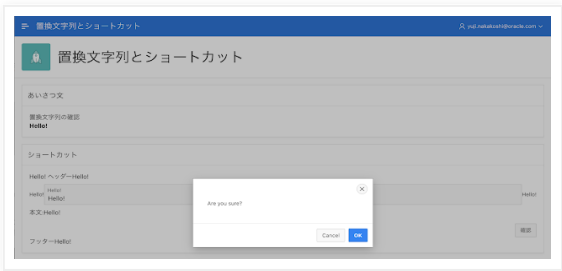


最後にアプリケーションの言語を英語に変更して、置換文字列やショートカットが英語に切り替わるか確認します。この確認のためだけにアプリケーション言語を追加するのも手間なので、アプリ

ケーションのプライマリ言語を英語(en)に変更します。



置換文字列およびショートカットで指定された箇所がすべて英語になっていることが確認できます。



以上でOracle APEXで作成したアプリケーションを多言語に対応させるために利用可能な機能についての解説は終了です。

完

Yuji N. 時刻: 19:36

共有

ウェブ バージョンを表示

自己紹介

Yuji N.

日本オラクル株式会社に勤務していて、Oracle APEXのGroundbreaker Advocateを拝命しました。こちらの記事につきましては、免責事項の参照をお願いいたします。

詳細プロフィールを表示