

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。

2022年3月25日 金曜日

PL/SQL APIのAPEX_ITEMを使う

Oracle APEXのPL/SQL APIとしてパッケージ[APEX_ITEM](#)が提供されています。APIのマニュアルを参照すると、**APEX_ITEM (Legacy)**となっており、また、以下のように記載されています。

This API is designated as legacy.

You can use the APEX_ITEM package to create form elements dynamically based on a SQL query instead of creating individual items page by page.

そのため、使ってはいけない気になります。しかし、Legacyだといっているだけで、サポートをやめているわけではありません。保証はできませんが、Oracle APEXのPL/SQL APIの動作はオラクル・データベースに依存しているので、オラクル・データベース自体の仕様変更がない限りは、PL/SQL APIがデサポートになるというのはあまり考えられません。そして、過去に動いていたSQLやPL/SQLが動かなくなるような仕様変更は、（不具合でもない限り）あまり聞いたことがありません。

Oracle APEXであれば、例えば[Tabular Form](#)はLegacyとなっています。マニュアルには以下の注意書きがあります。APEX_ITEMはここまで強く書かれていません。

Note: A tabular form is a legacy application component. Although existing legacy tabular forms are supported, the creation of new legacy tabular forms has been desupported. Oracle recommends developers create interactive grids instead.

対話グリッドの利用が推奨ですが、すでに作られているものについてはサポートを継続すると記載されています。この他にもJavaScript APIに含まれる[Non-namespace APIs](#)もJavaScriptが一般的に使われるようになる前に開発したもので、推奨はNamespaceがあるAPIです。しかし、サポートをやめるという話はありません。

[Oracle Multimediaのように無くなる機能](#)はあるので100%大丈夫ということはありませんが、開発チームには「きちんと代替できる機能が提供されるまでは、既存の機能を除くことはない。」という方針があるようです。

さて、現在提供されているコンポーネントでは代替できない実装として、クラシック・レポートとAPEX_ITEMを組み合わせた更新可能なレポートがあります。以下のようなアプリケーションになります。

更新可能レポート

更新可能レポート

従業員

Empno	Tl	Ename	Job	Mgr	Hiredate	Sal	Comm	Empno
7800		クラーク	MANAGER	7839	1980/12/18	1600	1400	7800
7801		ケルナー	MANAGER	7839	1981/02/09	1600	1400	7801
7811		クラーク	MANAGER	7839	1981/02/02	1200	1400	7811
7844		クラーク	MANAGER	7839	1981/04/02	2974		7844
7869		クラーク	MANAGER	7839	1981/06/09	1200	1400	7869
7876		クラーク	MANAGER	7839	1981/06/09	2850		7876
7882		クラーク	MANAGER	7839	1981/06/09	2450		7882
7889		クラーク	MANAGER	7839	1981/12/09	3000		7889
7839		クラーク	MANAGER	7839	1981/01/07	5000		7839
7844		クラーク	MANAGER	7839	1981/02/09	1500		7844
7876		クラーク	MANAGER	7839	1981/06/09	1500		7876
7882		クラーク	MANAGER	7839	1981/06/09	1500		7882
7889		クラーク	MANAGER	7839	1981/12/09	1500		7889
7839		クラーク	MANAGER	7839	1981/01/07	5000		7839
7844		クラーク	MANAGER	7839	1981/02/09	1500		7844
7876		クラーク	MANAGER	7839	1981/06/09	1500		7876
7882		クラーク	MANAGER	7839	1981/06/09	1500		7882
7889		クラーク	MANAGER	7839	1981/12/09	1500		7889

1 / 14

実装方法について、以下より簡単に紹介します。

サンプル・データセットのEMP/DEPTに含まれる表EMPを使います。

SQLワークショップのユーティリティのサンプル・データセットを開き、EMP/DEPTをインストールします。アプリケーションの作成は不要です。

APEX アプリケーションビルダー SQLワークショップ

サンプル・データセット

アクション	名前	説明	言語	ユーザー	インストール	リフレッシュ
インストール	EMP / DEPT	英語、中国語、カナル、フランス語、ドイツ語、スペイン語、ポルトガル語、イタリア語、ロシア語、韓国語、日本語。	英語	SYSTEM	インストール	リフレッシュ
インストール	EMP / DEPT	英語、中国語、カナル、フランス語、ドイツ語、スペイン語、ポルトガル語、イタリア語、ロシア語、韓国語、日本語。	英語	SYSTEM	インストール	リフレッシュ
インストール	EMP / DEPT	英語、中国語、カナル、フランス語、ドイツ語、スペイン語、ポルトガル語、イタリア語、ロシア語、韓国語、日本語。	英語	SYSTEM	インストール	リフレッシュ
インストール	EMP / DEPT	英語、中国語、カナル、フランス語、ドイツ語、スペイン語、ポルトガル語、イタリア語、ロシア語、韓国語、日本語。	英語	SYSTEM	インストール	リフレッシュ
インストール	EMP / DEPT	英語、中国語、カナル、フランス語、ドイツ語、スペイン語、ポルトガル語、イタリア語、ロシア語、韓国語、日本語。	英語	SYSTEM	インストール	リフレッシュ
インストール	EMP / DEPT	英語、中国語、カナル、フランス語、ドイツ語、スペイン語、ポルトガル語、イタリア語、ロシア語、韓国語、日本語。	英語	SYSTEM	インストール	リフレッシュ

アプリケーション作成ウィザードを起動し、空のアプリケーションを作成します。名前は更新可能レポートとします。

アプリケーションの作成を実行します。

APEX アプリケーションビルダー SQLワークショップ

アプリケーションの作成

名前: 更新可能レポート

タイプ: Web, サイドメニュー

ページ: + ページの追加

ホーム: 空白

機能:

- 検索ページ: 検索ページ (このアプリケーションについて、ページ検索)
- アクセス制御: ロール、パス、ユーザーごとの権限
- フィードバック: ユーザーによるフィードバックの機能
- アクセシビリティレポート: このアプリケーションのアクセシビリティレポート

設定:

- テーマ・スタイルの選択: テーマ・スタイルの選択 (このアプリケーションのテーマ・スタイルの選択)

実行: 実行

アプリケーションが作成されたら、ページ・デザイナーにてホーム・ページを開きます。

Content Bodyにリージョンを作成します。

識別の名前は従業員とします。タイプとしてクラシック・レポートを選択します。ソースのタイプとしてSQL問合せを選択し、SQL問合せとして以下を入力します。

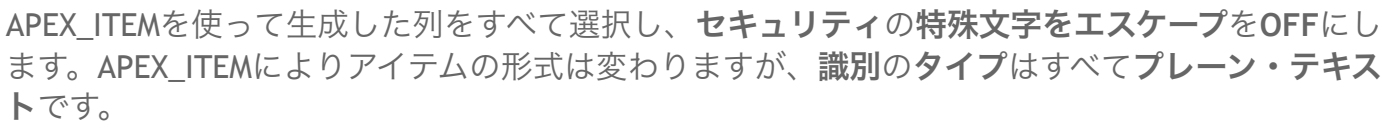
select

```

apex_item.text(
    p_idx => 1
    , p_attributes => 'class="apex_disabled"'
    , p_value => empno
) empno
, ename
, apex_item.select_list_from_query(
    p_idx => 2
    , p_value => job
    , p_query => 'select distinct job d, job r from emp'
    , p_show_extra => 'NO'
    , p_item_label => 'JOB'
) JOB
, apex_item.select_list_from_query(
    p_idx => 3
    , p_value => mgr
    , p_query => 'select ename d, empno r from emp'
    , p_null_value => ''
    , p_null_text => 'マネージャーなし'
    , p_show_extra => 'NO'
    , p_item_label => 'MGR'
) MGR
, apex_item.date_popup2(
    p_idx => 4
    , p_value => hiredate
    , p_item_label => 'HIREDATE'
) HIREDATE
, apex_item.text(
    p_idx => 5
    , p_value => to_char(sal)
    , p_size => 8
    , p_item_label => 'SAL'
) sal
, apex_item.text(
    p_idx => 6
    , p_value => to_char(comm)
    , p_size => 8
    , p_item_label => 'COMM'
) comm
, apex_item.select_list_from_query(
    p_idx => 7
    , p_value => deptno
    , p_query => 'select dname d, deptno r from dept'
    , p_show_extra => 'NO'
    , p_item_label => 'DEPTNO'
) DEPTNO
from EMP

```

プロシージャAPEX_ITEM.TEXTなどを呼び出し、クラシック・レポートの列としてアイテムを生成し、更新可能にしています。



更新処理を組み込みます。

レポートのリージョン従業員に送信ボタンを作成します。

識別のボタン名はB_SUBMIT、ラベルは送信とします。動作のアクションはデフォルトのページの送信とします。



左ペインで**プロセス・ビュー**を開き、実際のデータベースの更新処理を**プロセス**として実装します。

プロセスを作成します。

識別の名前は**従業員の更新**とします。**タイプ**は**コードを実行**です。**ソース**のPL/SQLコードとして以下を記述します。

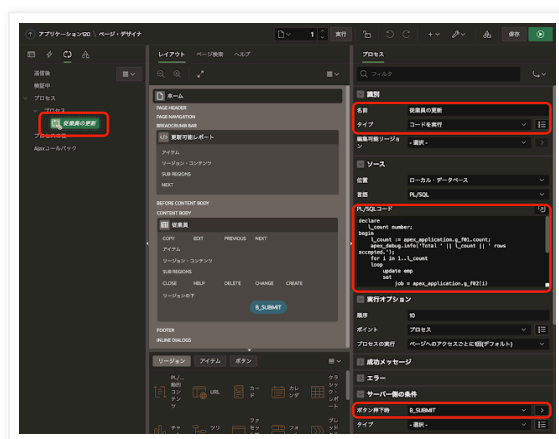
```
declare
    l_count number;
begin
    l_count := apex_application.g_f01.count;
    apex_debug.info('Total ' || l_count || ' rows accepted.');
```

```
for i in 1..l_count
loop
    update emp
    set
        job = apex_application.g_f02(i)
        , mgr = apex_application.g_f03(i)
        , hiredate = apex_application.g_f04(i)
        , sal = apex_application.g_f05(i)
        , comm = apex_application.g_f06(i)
        , deptno = apex_application.g_f07(i)
    where
        empno = apex_application.g_f01(i);
end loop;
```

```
end;
```

レポートに指定した値は、**APEX_APPLICATION.G_Fxx**に配列として送信されます。**xx**の数値はAPEX_ITEMの呼び出しで、引数**p_idx**として指定した数値になります。

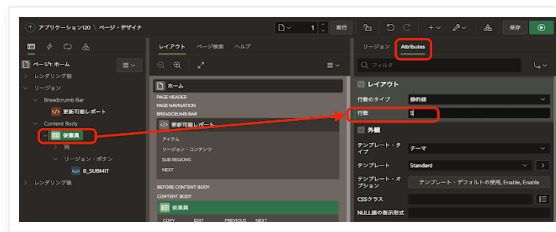
サーバー側の条件として、**ボタン押下時**に**B_SUBMIT**を選択します。



以上でアプリケーションは完成です。

対話グリッドとは異なり、画面に表示されている行だけが更新の対象になります。ページ送りには対応していません。

例えば、**レイアウトの行数**を**5**に変更します。

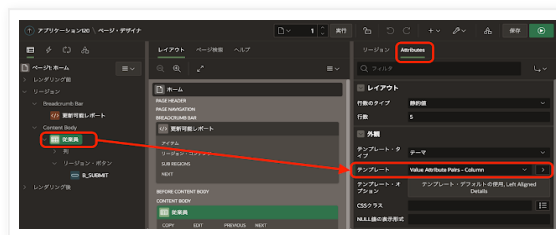


クラシック・レポートには、最大5行が表示されます。送信ボタンを押したときに更新されるのは、その時点で表示されている5行だけです。

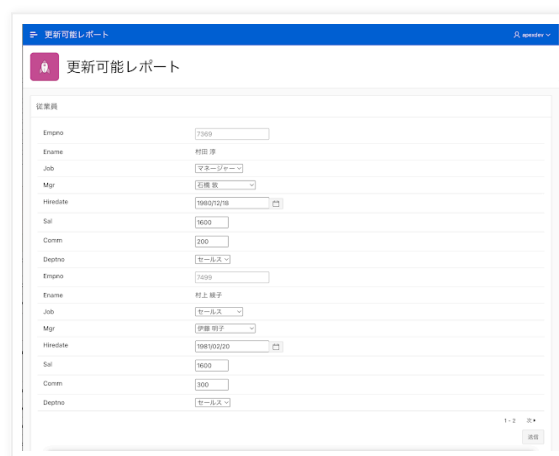


機能的には対話グリッドの方が豊富ですが、クラシック・レポートによる更新フォームは見かけを変更する自由度が高いです。

例として、クラシック・レポートの**外観のテンプレート**を**Value Attribute Pairs - Column**に変更してみます。



縦方向に列が表示されますが、更新可能レポートとしての動作は変わりません。（画面を見やすくするために行数を2に変更しています。）



クラシック・レポートのテンプレートをカスタマイズすることにより、より特別な見た目の更新可能なレポートを作成することができます。

今回作成したアプリケーションのエクスポートを以下に置きました。

<https://github.com/ujnak/apexapps/blob/master/exports/updatable-classic-report.sql>

Oracle APEXのアプリケーション作成の参考になれば幸いです。

完

Yuji N. 時刻: 14:49

共有

<

ホーム

>

[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](#)

Powered by Blogger.
