

日々は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。

2023年5月19日 金曜日

リージョン・タイプURLを使ってみる

Oracle APEXで利用できるリージョンのタイプにURLというものがあります。これは外部のURLを参照して、リージョンの内容として表示するというものです。**組入れモード**として、**IFrame**、**Inline(no escaping)**、**Inline(escape special characters)**の3種類を選べます。

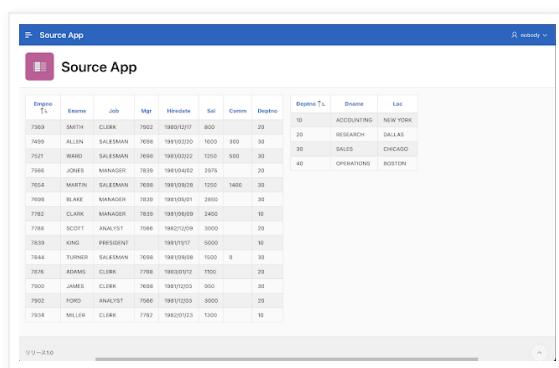
参照される側として、以下のアプリケーションを作成します。

サンプル・データセットのEMP/DEPTに含まれる表EMPをソースとしたクラシック・レポートを画面左に配置し、表DEPTのクラシック・レポートを画面右に配置します。

参照される側のアプリケーションは、APEXアプリケーションではないことが一般的です。クリックジャッキングといった脆弱性の元になるため、**APEXアプリケーションはデフォルトでiFrameへの埋め込みを禁止しています。**

アプリケーション名は**Source App**としています。アプリケーションのエクスポートは以下になります。

<https://github.com/ujnak/apexapps/blob/master/exports/source-app.zip>



Empno	Ename	Job	Mgr	Hiredate	Sal	Comm	Deptno
7369	SMITH	CLERK	7662	1980/12/10	800		20
7499	ALLEN	SALESMAN	7698	1981/02/20	1600	300	30
7521	WARD	SALESMAN	7698	1981/02/22	1200	500	30
7566	JONES	MANAGER	7698	1981/04/02	2975		20
7654	MARTIN	SALESMAN	7698	1981/09/28	1200	1400	30
7698	BLAKE	MANAGER	7633	1981/05/01	2850		30
7782	CLARK	MANAGER	7633	1981/06/09	2450		10
7788	SCOTT	ANALYST	7566	1982/07/09	3000		20
7839	KING	PRESIDENT		1981/11/17	5000		10
7844	TURNER	SALESMAN	7698	1981/09/08	1500	0	30
7876	ADAMS	CLERK	7788	1983/07/12	1500		20
7900	JAMES	CLERK	7698	1981/12/03	950		30
7902	FORD	ANALYST	7566	1981/12/03	3000		20
7934	MILLER	CLERK	7782	1982/07/23	1300		10

このアプリケーション全体をiFrameに取り込むアプリケーションを作成します。

アプリケーション名は**iFrame App**としています。アプリケーションのエクスポートは以下になります。

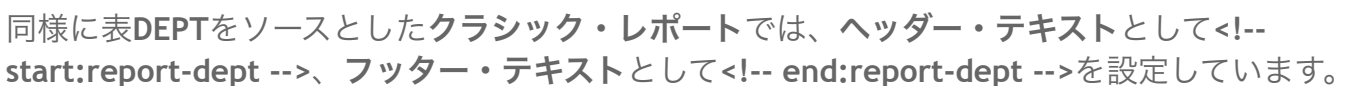
<https://github.com/ujnak/apexapps/blob/master/exports/iframe-app.zip>

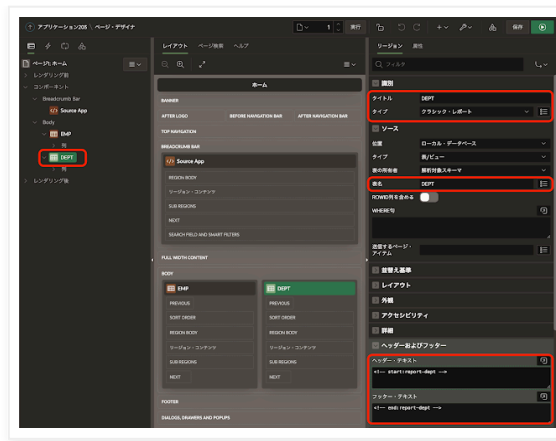


<https://github.com/ujnak/apexapps/blob/master/exports/inline-app.zip>



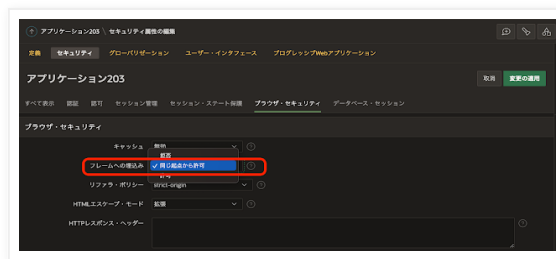
Inlineでの組み込みの範囲を指定するため、ヘッダーおよびフッターのヘッダー・テキストとして<!-- start:report-emp -->、フッター・テキストとして<!-- end:report-emp -->を設定しています。コメントであるため、画面には表示されません。



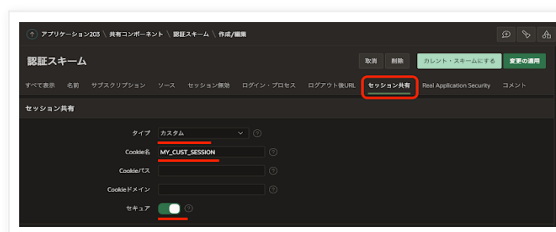


アプリケーション定義のセキュリティのブラウザ・セキュリティのセクションに含まれるフレームへの埋込みを、デフォルトの拒否から同じ起点から許可に変更します。これはX-Frame-Optionsヘッダーに指定する値です。

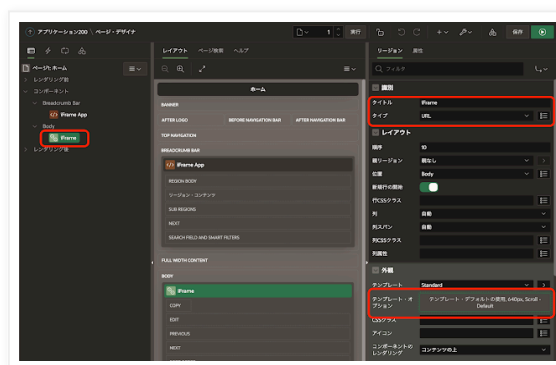
組込む方法が異なるため、この設定はInlineでの組込みに影響を与えません。



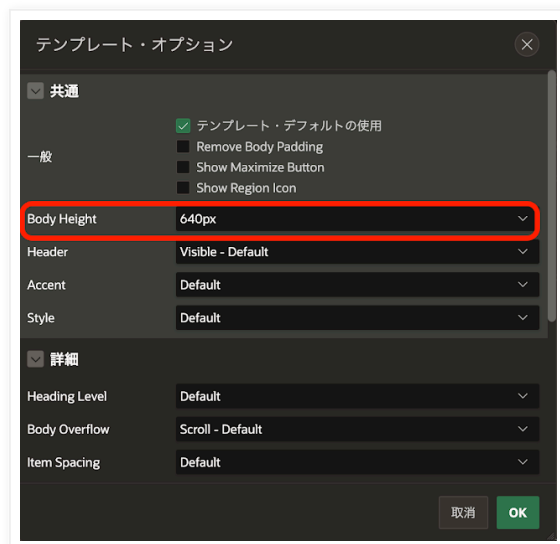
iFrame AppとSource Appが同じワークスペースに作成されている場合、認証スキームのセッション共有で同じCookie名に設定にしていると、iFrame AppとSource Appでセッションを共有できます。つまり、iFrame AppがSource Appを呼び出す際に、Source Appのログイン処理をスキップできます。



Source Appを埋め込むiFrame Appには、識別のタイプがURLであるリージョンだけが作成されています。



URLリージョンの高さは自動調整されないためテンプレート・オプションを使って、**Body Height**として**640px**を設定しています。

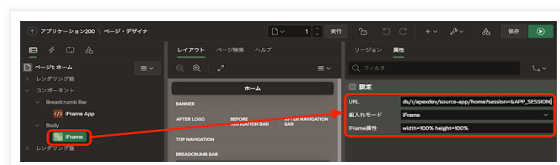


プロパティ・エディタでリージョンの属性を開きます。

設定のURLにSource Appのホーム・ページを指すURLを入力します。**session=&APP_SESSION.**として**セッションID**を渡すことによりセッションが引き継がれ、ログイン処理がスキップされます。

https://ホスト名/ords/r/apexdev/source-app/home?session=&APP_SESSION.

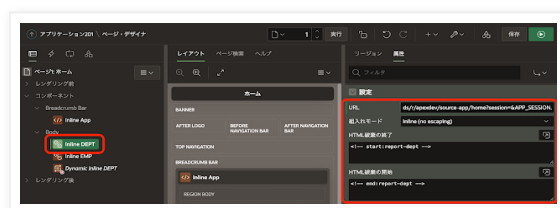
組入れモードはIFrame、IFrame属性として**width=100% height=100%**を設定しています。



IFrame（インラインフレーム要素）そのものが実装されていると考えてよいでしょう。

Inline Appでは、URLリージョンの組入れモードをInline (no escaping)にしています。

表DEPTを表示するリージョンでは、設定のHTML破棄の終了として`<!-- start:report-dept -->`（この文字列が現れるまでは、HTMLテキストを破棄する - つまりここからが表示を開始する）、HTML破棄の開始として`<!-- end:report-dept -->`（ここからのHTMLテキストは破棄する - つまりここで表示を終了する）を設定しています。



表EMPを表示するリージョンのHTML破棄の終了は`<!-- start:report-emp -->`、HTML破棄の開始は`<!-- end:report-emp -->`です。



組入れモードがIFrameの場合は、設定されたURLはブラウザがアクセスします。Inlineの場合は、バックエンドのデータベースがアクセスします。これは動的コンテンツのリージョンで、以下のコードを実行しているのと同様です。

```
declare
    l_response clob;
    l_begin pls_integer;
    l_end pls_integer;
    C_START constant varchar2(40) := '<!-- start:report-dept -->';
    C_END constant varchar2(40) := '<!-- end:report-dept -->';
begin
    l_response := apex_web_service.make_rest_request(
        p_url => :G_APEX_PATH || 'r/apexdev/source-app/home'
        ,p_http_method => 'GET'
    );
    l_begin := instr(l_response, C_START) + length(C_START);
    l_response := substr(l_response, l_begin);
    l_end := instr(l_response, C_END) - 1;
    l_response := substr(l_response, 1, l_end);
    -- return apex_escape.html(l_response); -- エスケープあり
    return l_response; -- エスケープなし
end;
```

plsqli-inline.sql hosted with ❤ by GitHub

[view raw](#)

データベース・サーバーによってHTTPリクエストが発行されているため、ソースとなっているAPEXアプリケーションは認証なし（またはパブリック・ページ）になっている必要があります。

リージョン・タイプURLの紹介は以上になります。

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: 18:52

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](#)

Powered by Blogger.
