

日々是Oracle APEX

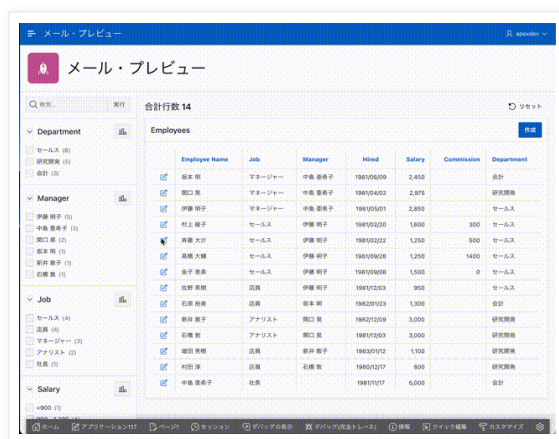
Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2022年5月19日 木曜日

APEX_MAIL.PREPARE_TEMPLATEを呼び出し、送信するメールをプレビューする

メール送信前に、電子メール・テンプレートの穴埋めを行なった結果を画面に表示して確認できないか、との相談がありました。調べてみたところ、プロシージャAPEX_MAIL.PREPARE_TEMPLATEを使うと、送信するメールのプレビューができそうです。

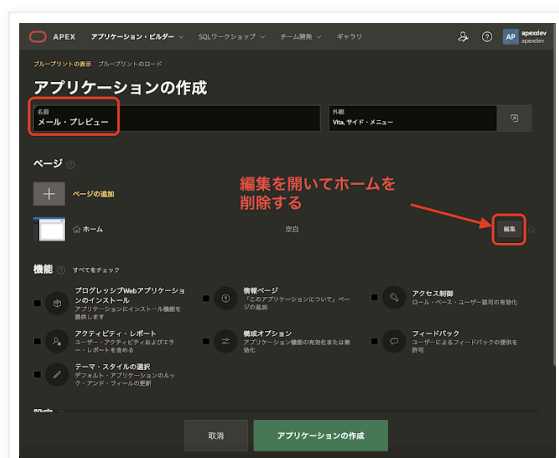
試しに実装してみました。



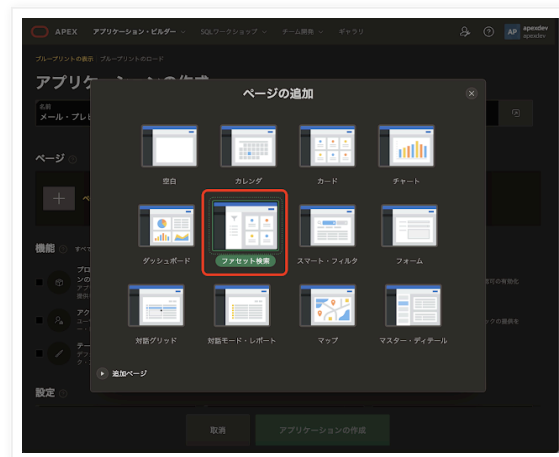
アプリケーションの作成は、Always FreeのAutonomous Database上のAPEX 21.2を使って行います。また、ソースとなるデータとして、**サンプル・データセット**のEMP/DEPTに含まれる表EMPを使用します。

以下より、サンプルの実装を紹介します。

アプリケーション作成ウィザードを起動します。名前は**メール・プレビュー**とします。ホーム・ページは、**編集**をクリックして設定画面を開き、**削除**します。



ページの追加を実行し、**ファセット検索**を選択します。

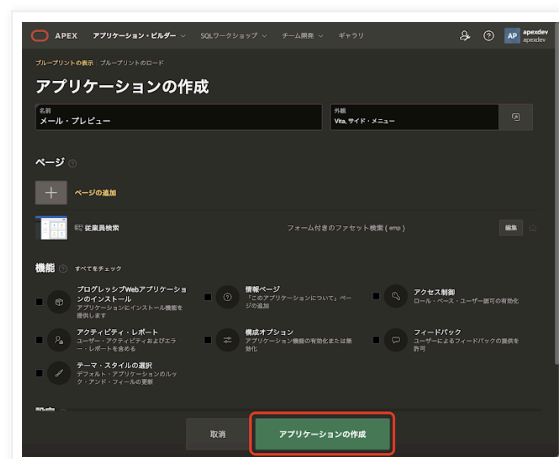


ページ名は**従業員検索**とします。レポートの表示形式は**レポート**、**表**として**EMP**を選択します。**フォームを含める**に**チェック**を入れ、表EMPの編集ページも作成します。**この編集ページを、電子メールのプレビューのページに作り直します。**

ページの追加をクリックします。

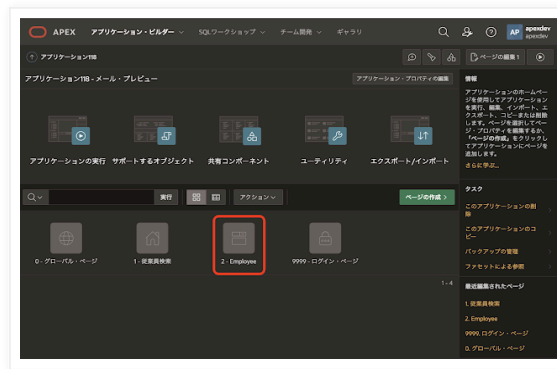


アプリケーションの作成を実行します。

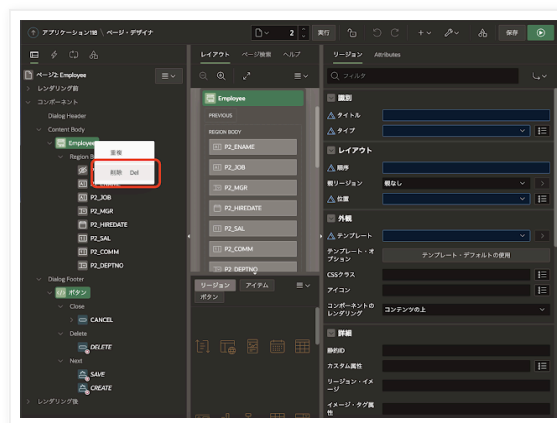


アプリケーションが作成されます。

フォームのページ**Employees**を開き、メールのプレビューに作り直します。



リージョン**Employees**と**ボタン**を削除します。これらのリージョンに配置されているページ・アイテムやボタンも削除します。



すべてのリージョンが無くなったところで、**Content Body**に新たにリージョンを作成します。

新規に作成したリージョンの識別の名前は**プレビュー**とし、**タイプ**に**PL/SQL動的コンテンツ**を選択します。**ソースのPL/SQLコード**として、以下を記述します。コード中で使っている電子メール・テンプレート**ORDERMAIL**は、この後に作成します。

```
declare
    l_html clob;
    l_text clob;
    l_placeholders clob;
    l_subject varchar2(4000);
begin
    -- 表EMPの1行をJSONのプレースホルダーとして取り出す。
    select json_object(*) into l_placeholders
    from (
        select empno, ename, mgr, job, to_char(hiredate,'DL') hiredate, sal
        from emp where empno = :P2_EMPNO
    );
    -- テンプレートの穴埋めを行う。
    apex_mail.prepare_template(
        p_static_id => 'ORDERMAIL'
        , p_placeholders => l_placeholders
        , p_application_id => :APP_ID
```

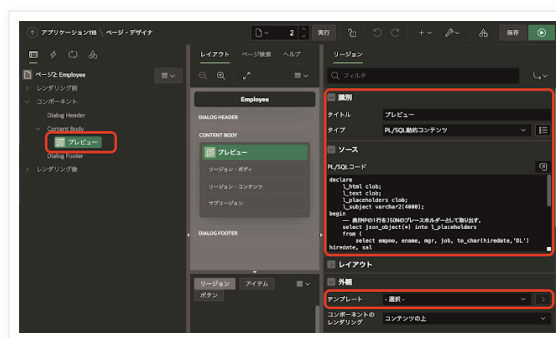
```

, p_subject => l_subject
, p_html => l_html
, p_text => l_text
);
-- リージョンにHTMLを印刷する。
http.p(l_html);
end;
```

preview_email_template.pls hosted with ❤ by GitHub

[view raw](#)

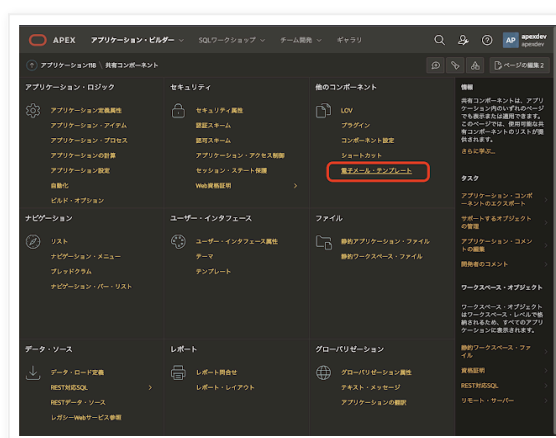
プレビューに影響を与える装飾をできるだけ無くするため、**外観のテンプレート**を**- 選択 -**に変更して、テンプレートを使わないようにします。



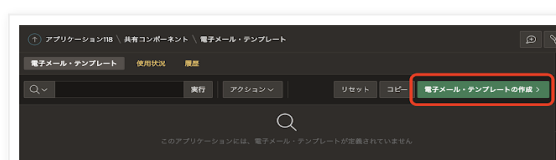
リージョン**プレビュー**にページ・アイテム**P2_EMPNO**を作成します。**タイプ**は**非表示**とします。このページ・アイテムに指定された従業員番号の従業員の情報を使って、電子メール・テンプレートの穴埋めを行います。

以上で、プレビューを行うページは作成できました。

共有コンポーネントの**電子メール・テンプレート**を開いて、テンプレート**ORDERMAIL**を作成します。



電子メール・テンプレートの作成を実行します。



テンプレート名をORDERMAIL（これはもっと読みやすい名前にできます）、静的識別子をORDERMAIL（これはコード中で使用しているので変更不可です）とします。電子メールの件名、HTMLフォーマットのヘッダー、本文、フッター、プレーン・テキスト・フォーマットのコンテンツを以下を参考に埋めていきます。（APIの動作さえ確認できればよかったので、正直、かなり手抜きです）。

件名

#ENAME#様、オーダーを確認しました。

ヘッダー

<b style="font-size: 24px;">マイ・アプリケーション

本文

ようこそ、#ENAME#さん、

オーダーありがとうございます。

オーダー詳細

<table width="100%">

<tr>

<th align="left">オーダー日</th>

<td>#HIREDATE#</td>

</tr>

<tr>

<th align="left">オーダー番号</th>

<td>#EMPNO#</td>

</tr>

<tr>

<th align="left">出荷先</th>

<td>#JOB#</td>

</tr>

<tr>

<th align="left" valign="top">出荷先住所</th>

<td>

東京都港区

北青山1丁目

</td>

</tr>

<tr>

<th align="left" valign="top">オーダー・アイテム</th>

<td>ラグビー・ボール</td>

</tr>

<tr>

<th align="left">オーダー合計</th>

<td>#SAL#</td>

</tr>

```
</table>
<br>
<br>
オーダーを変更しますか。<a href="#ORDER_URL#">お客様のオーダー#ENAME#はここで管理しています。</a>
```

フッター

```
<a href="#MY_APPLICATION_LINK#">「マイ・アプリケーション」にアクセスし、電子メール・プリファレンスを管理します</a>。
```

プレーン・テキスト・フォーマット

ようこそ、#ENAME#さん、

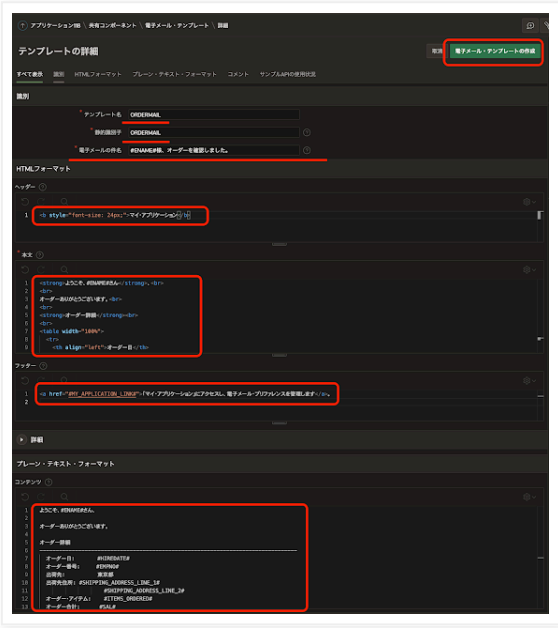
オーダーありがとうございます。

オーダー詳細

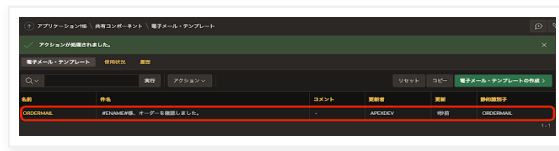
オーダー日:	#HIREDATE#
オーダー番号:	#EMPNO#
出荷先:	東京都
出荷先住所:	#SHIPPING_ADDRESS_LINE_1# #SHIPPING_ADDRESS_LINE_2#
オーダー・アイテム:	#ITEMS_ORDERED#
オーダー合計:	#SAL#

オーダーを変更しますか。お客様のオーダー#ORDER_NUMBER#はここで管理しています: #ORDER_URL#

テンプレートが決まったら、電子メール・テンプレートの作成を実行します。



電子メール・テンプレートORDERMAILが作成されます。



以上でアプリケーションは完成です。アプリケーションを実行すると、本記事の先頭にあるGIF動画のような動作をします。

今回のサンプルで表示しているのは、**HTML本文のみ**です。**件名**や**プレーン・テキスト**はプレビューに含めていません。それでも、確認した範囲ではAPEX_MAIL.PREPARE_TEMPLATEより得られたHTML本文は、十分、プレビューの役に立ちそうです。

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/email-preview.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: 22:45

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.