

日々是Oracle APEX

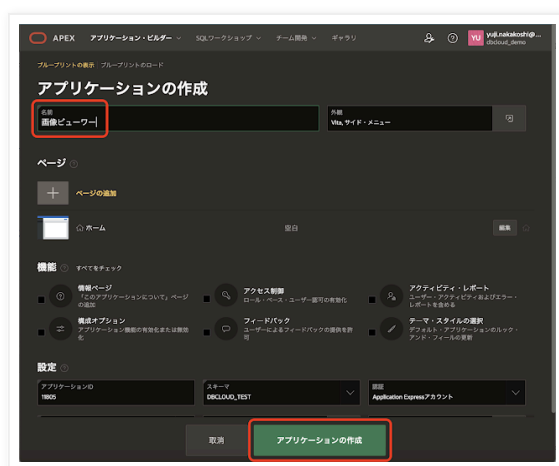
Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2021年9月8日 水曜日

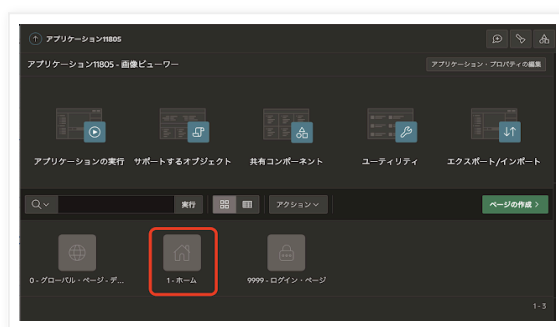
APEXの一時表を使った画像ビューワー・アプリの作成

Oracle APEXのページ・アイテムの**タイプ**に**ファイル参照...**というのがあります。このページ・アイテムを使って、選択した画像ファイルを表示するアプリケーションを作成してみます。

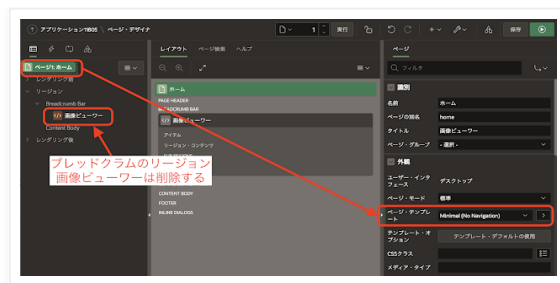
最初に空のアプリケーションを作成します。**アプリケーション作成ウィザード**を起動します。名前を**画像ビューワー**として、**アプリケーションの作成**を実行します。



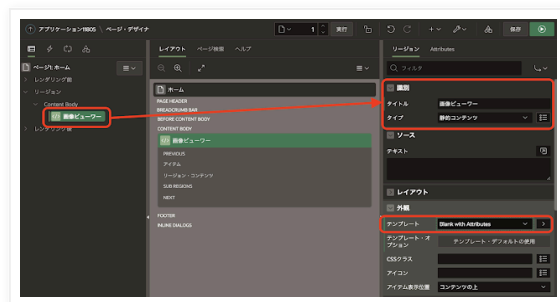
アプリケーションが作成されたら、**ページ・デザイナー**にて**ホーム・ページ**を開きます。ホーム・ページに画像のアップロードと表示の機能を実装します。



アプリケーションとしてはホーム・ページしか含まれないため、ナビゲーション・メニューやブレッডクラムは不要です。**ページ・プロパティ**の**外観**の**ページ・テンプレート**として**Minimail (No Navigation)**を選択し、ブレッডクラムの**リージョン画像ビューワー**を削除します。



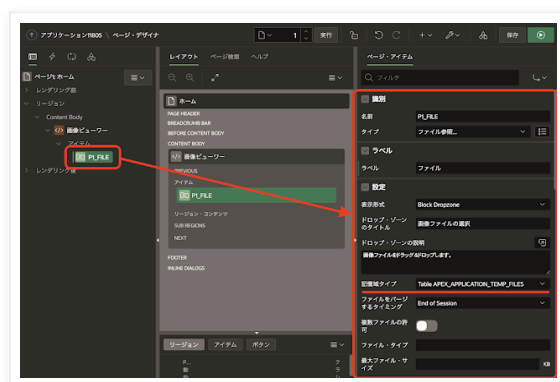
Content Bodyでリージョンの作成を実行します。識別のタイトルは画像ビューワーとし、タイプは静的コンテンツとします。リージョンとしての装飾は不要なので、外観のテンプレートとしてBlank with Attributesを選択します。



表示するファイルをローカルのファイル・システムから選択するためのページ・アイテムを作成します。

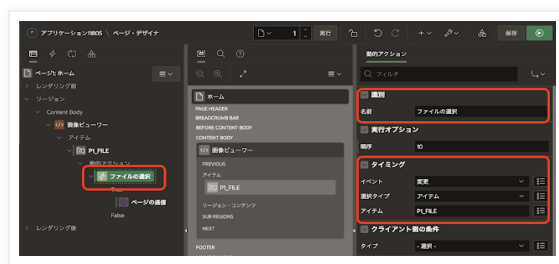
識別の名前をP1_FILE、タイプとしてファイル参照...を選択します。重要な設定は設定の記憶域タイプで、これにはTable APEX_APPLICATION_TEMP_FILESを選択します。この記憶域タイプが選択されている場合、ローカルのファイル・システムからアップロードされるファイルは、Oracle APEXが用意している表APEX_APPLICATION_TEMP_FILESに書き込まれます。記憶域タイプとしてBLOB column specified in Item Source attributeを選んだ場合は、BLOB列を持つ表をあらかじめ作成しておく必要があります。アップロードするファイルを永続的に保存する場合は、後者の設定を行った方がよいでしょう。

それ以外の設定は操作方法や見かけに関するものなので、必ずしもこの通りではなくても構いません。今回の手順では、ラベルとしてファイル、設定の表示形式はBlock Dropzone、ドロップ・ゾーンのタイトルとして画像ファイルの選択、ドロップ・ゾーンの説明として「画像ファイルをドラッグ&ドロップします。」を設定しています。

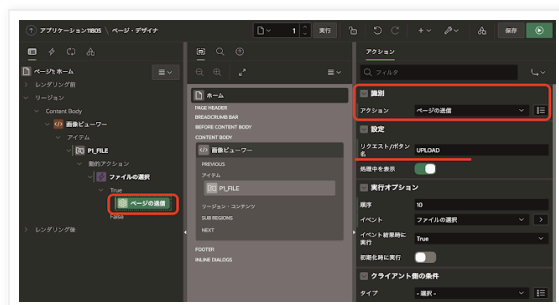


ページ・アイテムP1_FILEで選択されたファイルは、ページの送信を行うまではアップロードされません。ページ・アイテムP1_FILEの値の変更が行われたとき(ファイルが選択されたときになります)に、動的アクションによってページの送信を実行します。

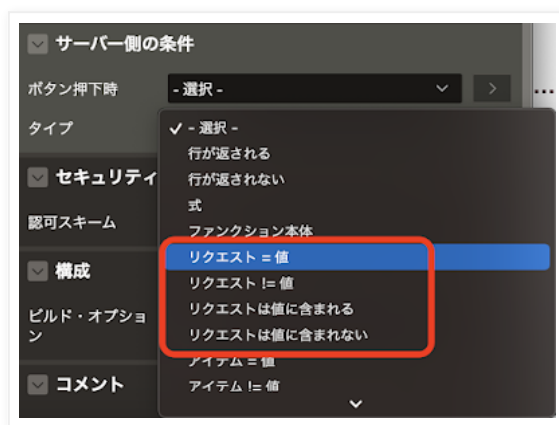
ページ・アイテムP1_FILEで動的アクションの作成を実行します。識別の名前はファイルの選択とします。**タイミング**はデフォルトで、**イベント**が変更、**選択タイプ**が**アイテム**、**アイテム**が**P1_FILE**になります。



TRUEアクションの識別のアクションとして、**ページの送信**を選択します。これで選択したファイルのアップロードが行われます。今回のアプリケーションでは使用しませんが、**設定のリクエスト/ボタン名**は**UPLOAD**としています。この名前がリクエストの値として送信されるので、サーバー側に作成する**検証**や**プロセス**の**サーバー側の条件**で使用することができます。



サーバー側の条件の**タイプ**には、**リクエスト**を参照するものがあります。これらの条件として上記で設定した**リクエスト/ボタン名**を使うことができます。



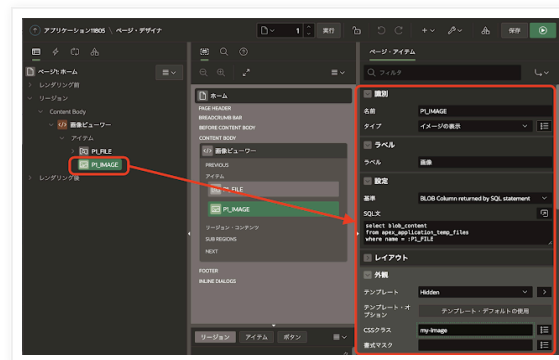
以上でファイルを選択すると、サーバー側に選択されたファイルがアップロードされるようになりました。アップロードされたファイルは表APEX_APPLICATION_TEMP_FILESより参照できます。

アップロードされたファイルを表示するページ・アイテムを作成します。

識別の名前を**P1_IMAGE**とします。**タイプ**として**イメージの表示**を選択します。**ラベル**の**ラベル**は**画像**、**設定の基準**として**BLOB Column returned by SQL statement**を選択し、**SQL文**として以下を記述します。表APEX_APPLICATION_TEMP_FILESよりアップロードされたファイルの内容である**BLOB列**を検索しています。

```
select blob_content
from apex_application_temp_files
where name = :P1_FILE
```

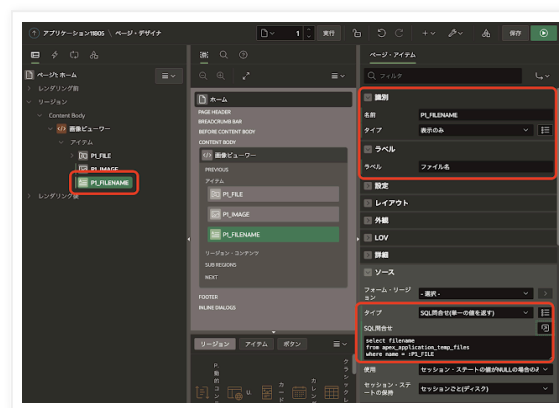
外観のテンプレートはHiddenを選択し、CSSクラスとしてmy-imageを設定しておきます。CSSクラス自体の記述は後ほど、ページ・プロパティに含めます。



アップロードされたファイルの、ファイル名を表示するページ・アイテムを作成します。

識別の名前をP1_FILENAME、タイプとして表示のみを選択します。ラベルのラベルはファイル名とします。ソースのタイプとしてSQL問合せ(単一の値を返す)を選択し、SQL問合せとして以下を記述します。

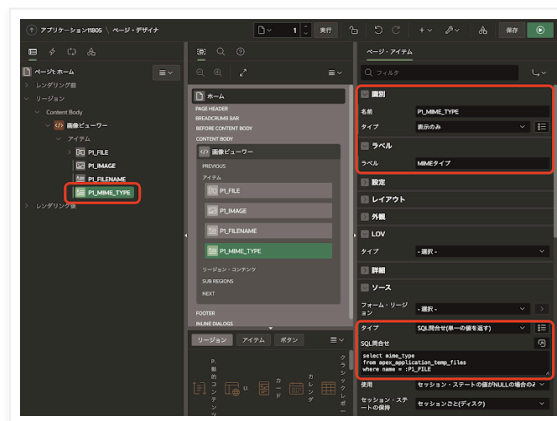
```
select filename
from apex_application_temp_files
where name = :P1_FILE
```



同様にアップロードされたファイルの、MIMEタイプを表示するページ・アイテムを作成します。

識別の名前をP1_MIME_TYPE、タイプとして表示のみを選択します。ラベルのラベルはMIMEタイプとします。ソースのタイプとしてSQL問合せ(単一の値を返す)を選択し、SQL問合せとして以下を記述します。

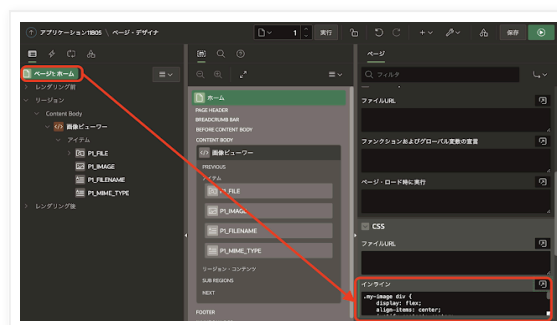
```
select mime_type
from apex_application_temp_files
where name = :P1_FILE
```



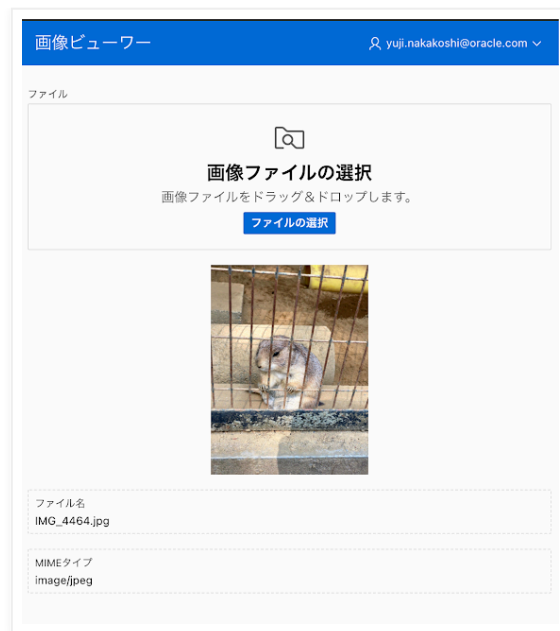
画像の表示を画面にフィットさせるため、ページ・プロパティのCSSのインラインに以下を記述します。

```
.my-image div {
    display: flex;
    justify-content: center;
}
```

```
.my-image img {
    width: 30%;
    height: auto;
}
```

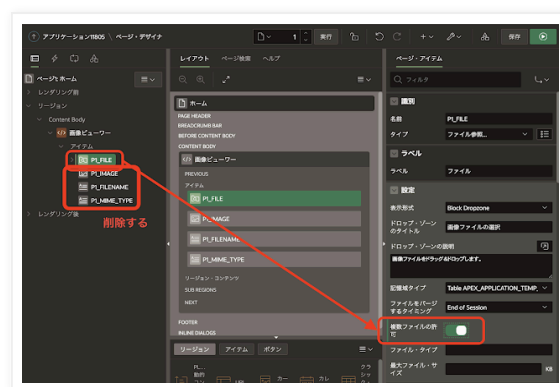


以上で完成です。ページを実行し画像ファイルをアップロードすると、以下のように表示されます。



画像の表示にカード・リージョンを使うと、複数の画像を表示することができます。

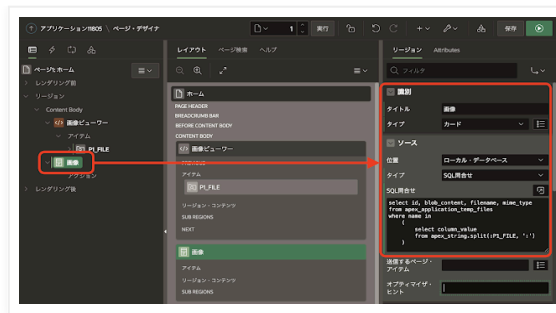
ページ・アイテムP1_FILEの設定の**複数ファイルの許可**をONに変更します。ページ・アイテムP1_IMAGE、P1_FILENAME、P1_MIME_TYPEは削除します。



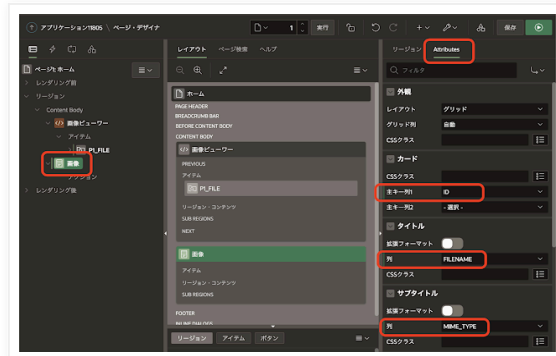
カード・リージョンを作成します。

識別の名前は**画像**とします。**タイプ**は**カード**です。**ソースの位置**は**ローカル・データベース**で、**タイプ**を**SQL問合せ**とします。**SQL問合せ**として以下を記述します。

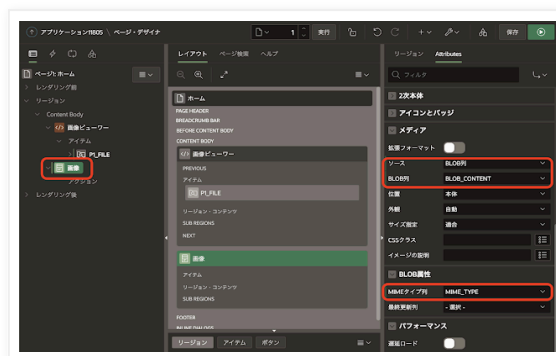
```
select id, blob_content, filename, mime_type
from apex_application_temp_files
where name in
(
    select column_value
    from apex_string.split(:P1_FILE, ':')
)
```



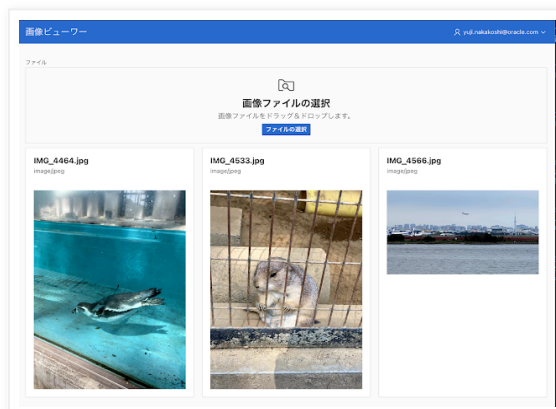
カード・リージョンのAttributesとして、カードの主キー列1にID、タイトルの列にFILENAME、サブタイトルの列としてMIME_TYPEを指定します。



メディアのソースをBLOB列とし、BLOB列にBLOB_CONTENTを指定します。BLOB属性のMIMEタイプ列にMIME_TYPEを指定します。



以上で複数ファイルのアップロードと表示を行うアプリケーションは作成できました。ページを実行し、複数の画像ファイルを選択すると、以下のように表示されます。



APEXの一時表APEX_APPLICATION_TEMP_FILESの使い方の紹介は以上になります。

ひとつの画像を表示するアプリケーションのエクスポートは以下です。

<https://github.com/ujnak/apexapps/blob/master/exports/image-viewer-single.sql>

複数の画像を表示するアプリケーションのエクスポートは以下です。

<https://github.com/ujnak/apexapps/blob/master/exports/image-viewer-multi.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: 14:14

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.
