

日々是Oracle APEX

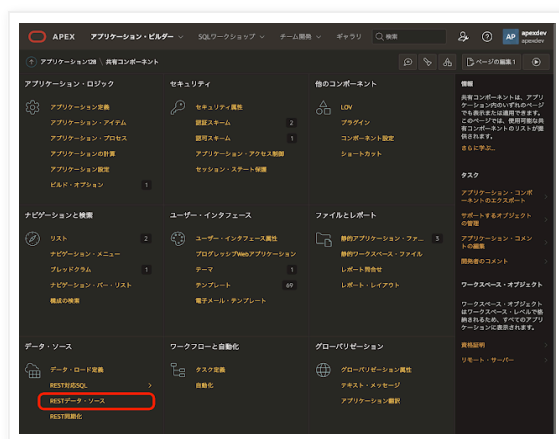
Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2023年1月19日 木曜日

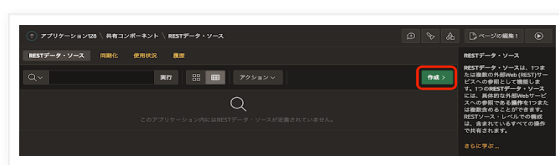
Microsoft OneDriveを操作するAPEXアプリの作成(4) - RESTデータ・ソース

OneDriveにあるドライブ、ルート直下のファイルまたはフォルダ（DriveItems）の一覧、フォルダに含まれるDriveItemsを取得するRESTデータ・ソースを作成し、それらをソースとした対話モード・レポートを作成します。

共有コンポーネントのRESTデータ・ソースを開きます。



作成をクリックします。



RESTデータ・ソースの作成は最初からとします。

次へ進みます。



RESTデータ・ソース・タイプとして簡易HTTPを選択します。

作成するRESTデータ・ソースの名前はMy Drivesとします。

URLエンドポイントとして、以下を指定します。所有しているドライブの一覧を取得します。

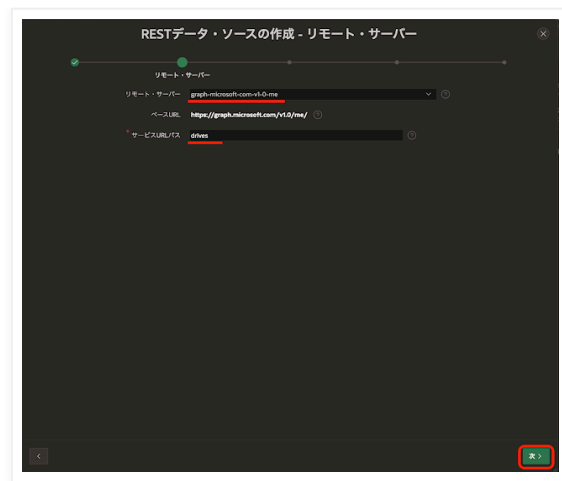
`https://graph.microsoft.com/v1.0/me/drives`

次へ進みます。



入力したURLエンドポイントより、リモート・サーバーやサービスURLパスが導出されます。

変更は不要です。そのまま次へ進みます。



ページ区切りタイプはページ区切りなしから変更しません。

Microsoft Graph APIはODataに準拠していることより、ページングをサポートしていると考えられます。そういう仕様も考慮する場合は、ODataプラグインの利用を検討すべきでしょう。

次へ進みます。



認証の設定に進みます。

認証が必要ですをONにします。資格証明として認証スキームで使用している**MS Azure AD Cred for OneDrive**を選択します。

OAuthトークンURLですが指定が必須になっています。**APEX_WEB_SERVICE.MAKE_REST_SERVICE**を呼び出す際には、そのような指定は無くても応答は返ってきているため、なぜこの値が必須なのか不明です。

適当な値を設定しても動くようですが、もっとも**それらしい値**を設定するよう努力します。

最初にトークンURLを確認します。Open ID Connectの検出URLをブラウザより直接呼び出します。

<https://login.microsoftonline.com/common/v2.0/.well-known/openid-configuration>

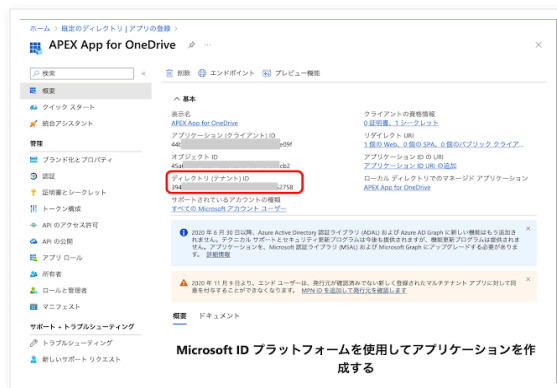
JSONの応答より、**token_endpoint**を確認します。

```
token_endpoint: "https://login.microsoftonline.com/tenantId/oauth2/token",
```

token_endpointは以下になります。

<https://login.microsoftonline.com/common/oauth2/v2.0/token>

commonの部分を、このアプリケーションのテナントIDで置き換えます。Azure Active Directoryのアプリのページを開き、ディレクトリ（テナント）IDをコピーします。



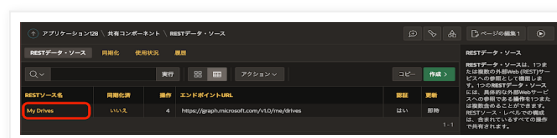
OAuthトークンURLに、commonをテナントIDで置き換えたURLを指定します。

検出をクリックすると、認証のエラーが発生します。そのため、RESTソースの手動作成をクリックします。

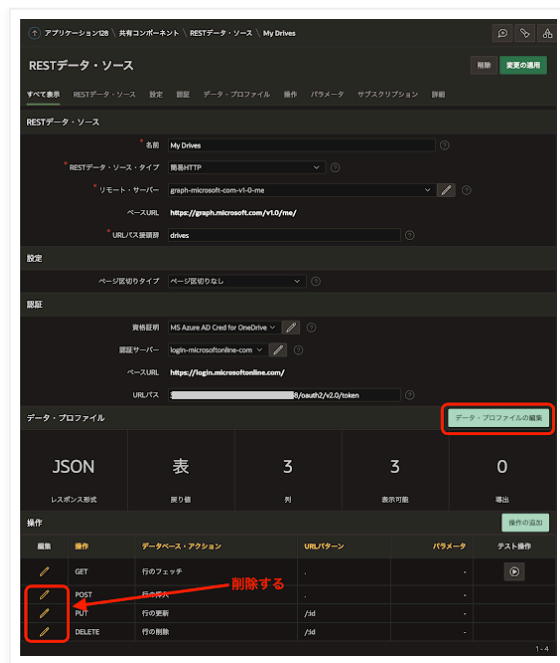


RESTデータ・ソースとしてMy Drivesが作成されます。

まだ設定は完了していないため、RESTデータ・ソースMy Drivesを開いて編集します。



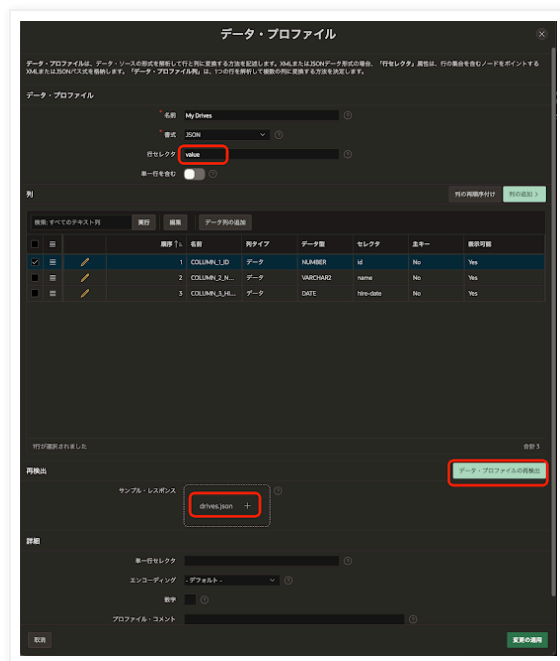
操作のPOST、PUT、DELETEは使用しないため、編集の鉛筆アイコンをクリックして削除します。その後、データ・プロファイルの編集をクリックします。



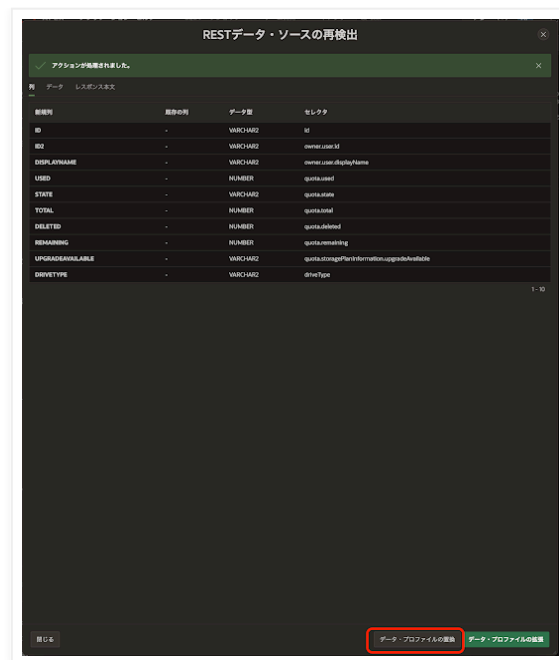
データ・プロファイルを編集するダイアログが開きます。

データ・プロファイルの行セクタとしてvalueを指定します。再検出のサンプル・レスポンスとしてGraph APIをテストしたときにJSON形式の応答を保存したファイルdrives.jsonを選択します。

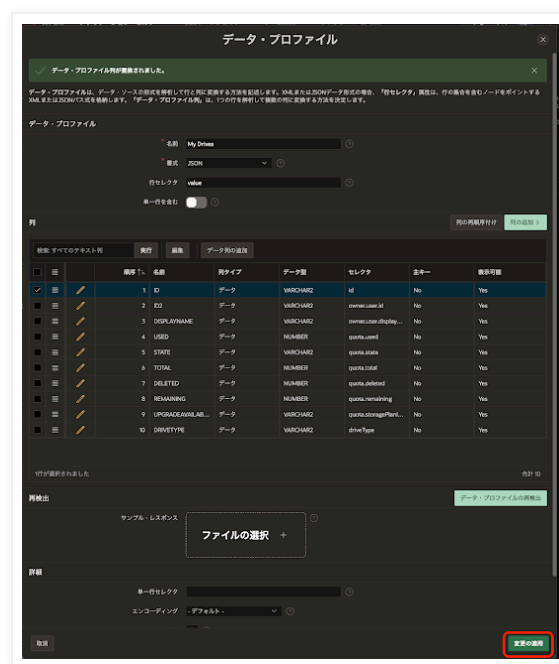
以上の指定を行い、データ・プロファイルの再検出をクリックします。



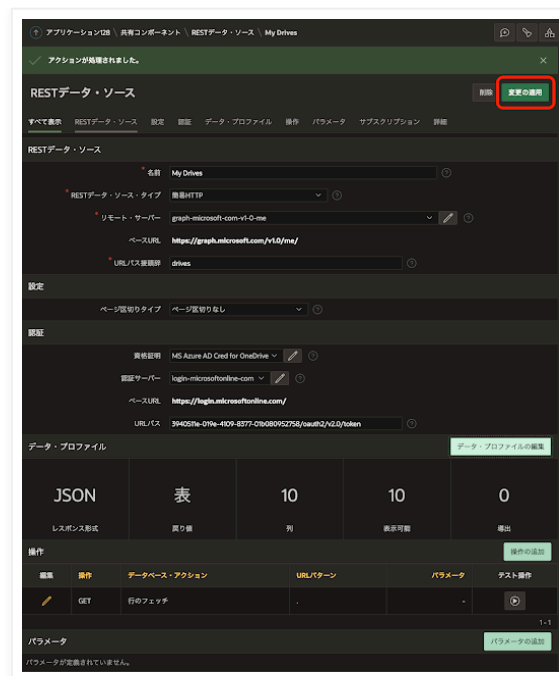
サンプル・レスポンスより列が検出されます。データ・プロファイルの置換をクリックします。



データ・プロファイルの列が置き換わります。**変更の適用**をクリックします。



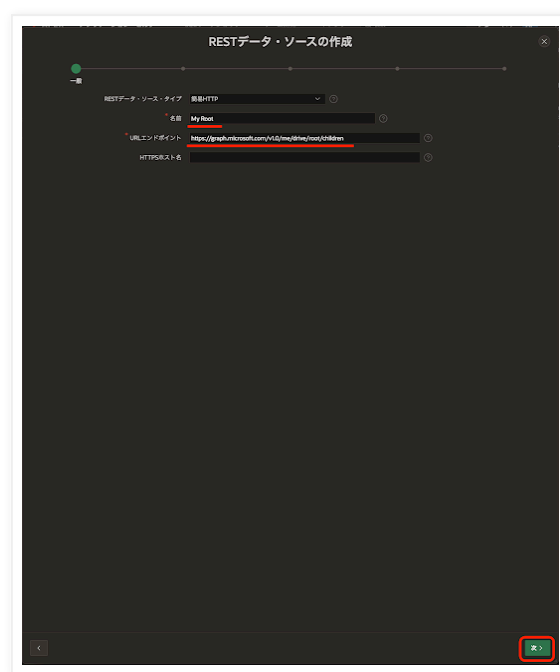
変更の適用をクリックし、RESTデータ・ソースMy Drivesの編集を完了します。



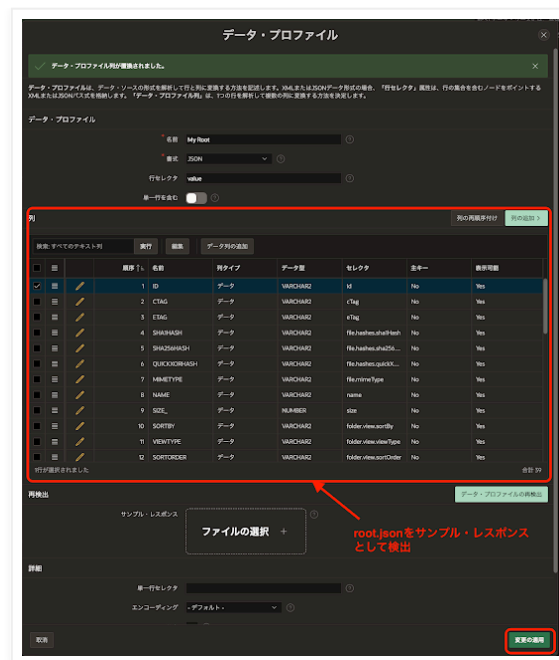
同様の手順で、ドライブのルート直下のファイルやフォルダを取得するRESTデータ・ソースを作成します。

RESTデータ・ソースの名前は**My Root**とします。URLエンドポイントの指定は以下になります。

https://graph.microsoft.com/v1.0/me/drive/root/children



データ・プロファイルのサンプル・レスポンスとして使用するファイルは、**root.json**として作成しているので、それを使用します。



RESTデータ・ソースとしてMy Drives、My Rootが作成されました。

フォルダを選択した上で、ファイルの一覧を表示するRESTデータ・ソースを作成するにはフォルダのIDが必要です。この2つのRESTデータ・ソースを使った対話モード・レポートを作成し（特にMy Root）、次に作成するRESTデータ・ソースに必要なフォルダIDを表示させます。

ページの作成を実行します。

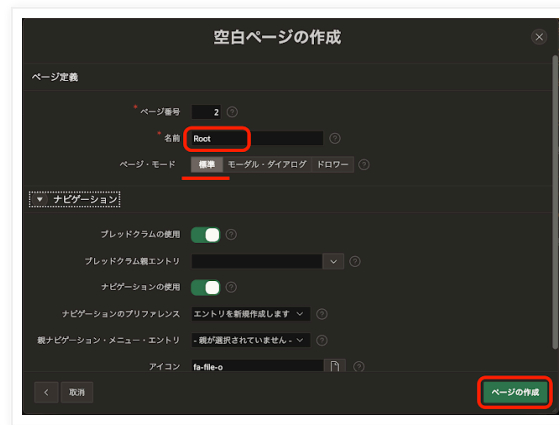


空白ページを選択します。



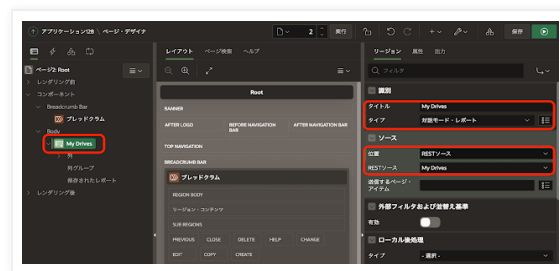
ページ定義の名前はRoot、ページ・モードは標準、ナビゲーションはデフォルトから変更しません。デフォルトでは、ブレッドクラム、ナビゲーション・メニューともに作成されます。

ページの作成をクリックします。

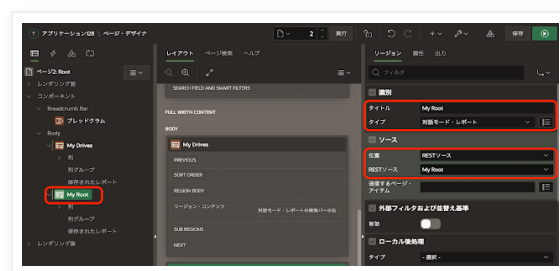


ページが作成されたら、RESTデータ・ソースMy Drivesをソースとする、対話モード・レポートのリージョンを作成します。

識別のタイトルはMy Drives、タイプとして対話モード・レポートを選択します。ソースの位置はRESTソース、RESTソースとしてMy Drivesを選択します。



同様にRESTソースがMy Rootの対話モード・レポートを作成します。



以上でページは完成です。

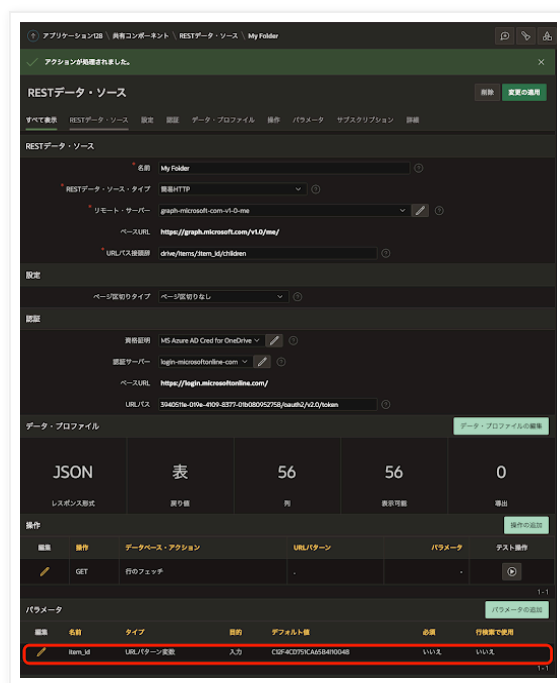
変更を保存し、ページを実行します。

ルート直下にあるフォルダの行を見つけ、Id値を取得します。



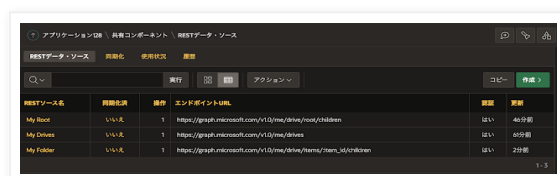
これ以降は、My DrivesやMy Rootの作成手順と同じです。

最後にRESTデータ・ソースの**パラメータ**としてitem_idの設定が含まれていることを確認します。



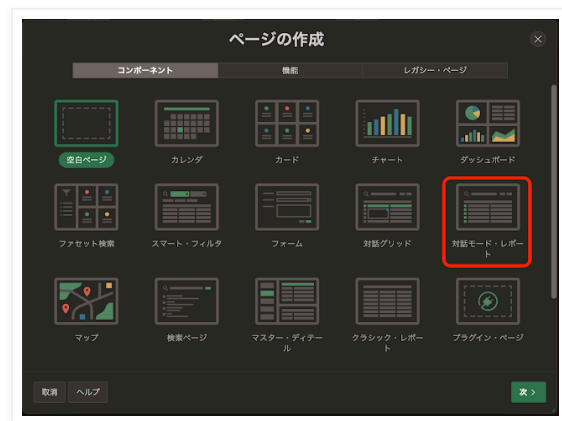
以上でRESTデータ・ソース**My Folder**が作成されました。

今回のアプリで使用するRESTデータ・ソース、My Drives、My Root、My Folderの3つが作成されました。



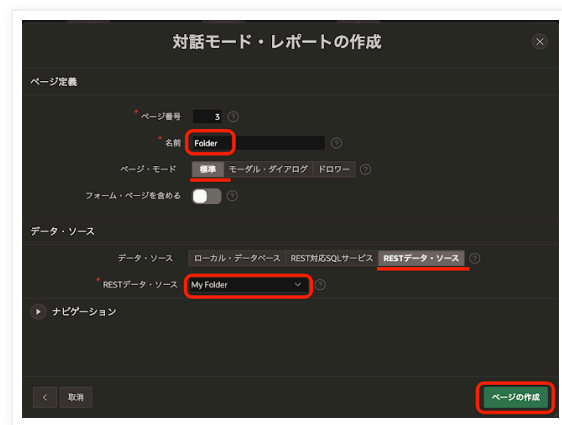
RESTデータ・ソース**My Folder**を使った対話モード・レポートのページを作成します。

ページの作成を実行し、**対話モード・レポート**を選択します。



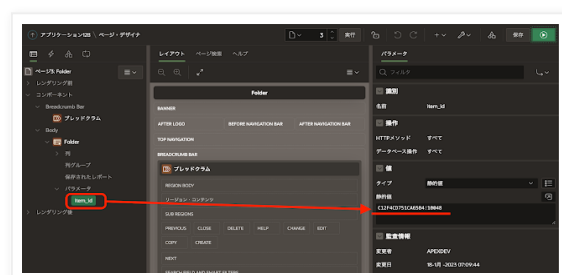
ページ定義の名前はFolder、ページ・モードは標準を選択します。データ・ソースとしてRESTデータ・ソースを選択し、RESTデータ・ソースはMy Folderとします。ナビゲーションはデフォルトから変更しません。

ページの作成をクリックします。



対話モード・レポートのページが作成されます。

レポートにはパラメータitem_idが含まれています。デフォルトでは静的値として、サンプル・レスポンスの出力に使用したIDが設定されています。



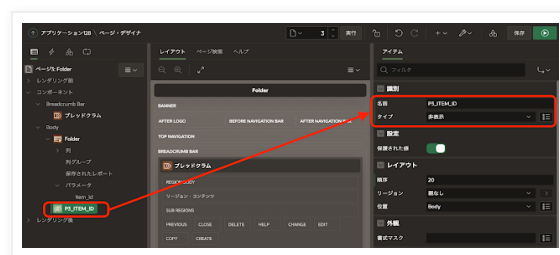
ページを実行すると、指定されたフォルダに含まれるアイテムの一覧が表示されます。

[illegible]

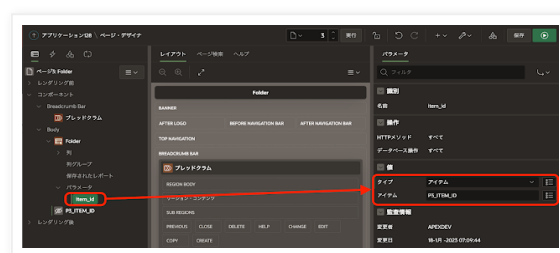
一覧を表示するフォルダを選択できるようにします。

ページ・アイテムを作成します。このページ・アイテムに、一覧表示の対象とするフォルダのアイテムIDを保持します。

識別の名前はP3_ITEM_ID、タイプは非表示とします。



対話モード・レポートのパラメータitem_idを選択し、値のタイプをアイテムに変更します。アイテムとしてP3_ITEM_IDを選択します。



ルート直下のフォルダやファイルを一覧しているレポート**My Root**に表示されている**フォルダ名**をクリックして、このMy Folderのレポートを開かれるようにリンクを追加します。

対話モード・レポートMy Rootを選択し、ローカル後処理のタイプをSQL問合せに変更します。

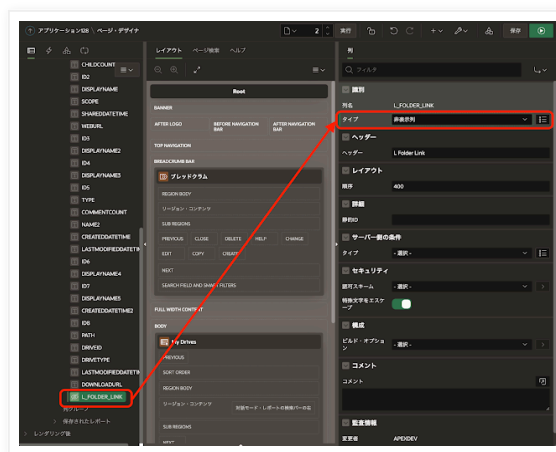
変更後に表示されるSQL問合せに、列**L FOLDER LINK**として以下を追加します。

```
,apex_page.get_url(
    p_page => 3
    ,p_items => 'P3_ITEM_ID'
    ,p_values => ID
```

) L_FOLDER_LINK

```
select ID,
       CTAG,
       ETAG,
       SHA1HASH,
       SHA256HASH,
       QUICKXORHASH,
       MIMETYPE,
       NAME,
       SIZE_,
       SORTBY,
       VIEWTYPE,
       SORTORDER,
       CHILDCOUNT,
       ID2,
       DISPLAYNAME,
       SCOPE,
       SHAREDATETIME,
       WEBURL,
       ID3,
       DISPLAYNAME2,
       ID4,
       DISPLAYNAME3,
       ID5,
       TYPE,
       COMMENTCOUNT,
       NAME2,
       CREATEDATETIME,
       LASTMODIFIEDDATETIME,
       ID6,
       DISPLAYNAME4,
       ID7,
       DISPLAYNAME5,
       CREATEDATETIME2,
       ID8,
       PATH,
       DRIVEID,
       DRIVETYPE,
       LASTMODIFIEDDATETIME2,
       DOWNLOADURL
,apex_page.get_url(
  p_page => 3
  ,p_items => 'P3_ITEM_ID'
  ,p_values => ID
) L_FOLDER_LINK
from #APEX$SOURCE_DATA#
```

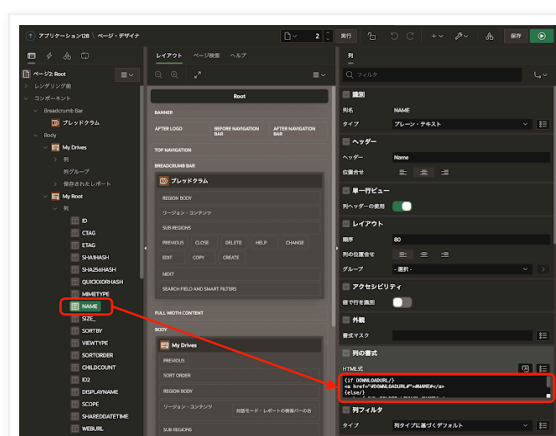
追加した列L_FOLDER_LINKはレポートに表示する必要はないため（また、OneDriveから取得できている列と間違えないように）、識別の**タイプ**を**非表示**に変更します。



OneDriveが返す一覧にはDOWNLOADURLという属性があります。実体のあるファイルの場合、このURLにアクセスすることでファイルをダウンロードできます。DOWNLOADURLに値がない場合にフォルダと認識して、レポートMy Folderを表示させます。

レポートMy Rootの列NAMEを選択し、**列の書式**のHTML式として以下を記述します。

```
{if DOWNLOADURL/}
<a href="#DOWNLOADURL#">#NAME#</a>
{else/}
<a href="#L_FOLDER_LINK#">#NAME#</a>
{endif/}
```



以上で、対話モード・レポートMy Rootの列NAMEをクリックすることで、それがフォルダであればMy Folderのレポートを表示し、ファイルであればダウンロードが実行されるようになりました。

同様の変更をレポートFolderに対しても実施します。

レポート**Folder**を選択し、**ローカル後処理**のSQLとして列**L_FOLDER_LINK**を追加します。表示する対象は自分自身なので、つねに同じページに存在します。そのため**p_page**には**:APP_PAGE_ID**を指定しています。

```
,apex_page.get_url(  
  p_page => :APP_PAGE_ID  
  ,p_items => 'P3_ITEM_ID'  
  ,p_values => ID  
) L_FOLDER_LINK
```

```
select ID,  
       CTAG,  
       ETAG,  
       SHA1HASH,  
       SHA256HASH,  
       QUICKXORHASH,  
       MIMETYPE,  
       NAME,  
       SIZE_,  
       WIDTH,  
       HEIGHT,  
       ISO,  
       FNUMBER,  
       CAMERAMAKE,  
       CAMERAMODEL,  
       FOCALENGTH,  
       TAKENDATETIME,  
       EXPOSURENUMERATOR,  
       EXPOSUREDENOMINATOR,  
       WIDTH2,  
       FOURCC,  
       HEIGHT2,  
       BITRATE,  
       DURATION,  
       FRAMERATE,  
       AUDIOFORMAT,  
       AUDIOCHANNELS,  
       AUDIOBITSPERSAMPLE,  
       AUDIOSAMPLESPERSECOND,  
       ID2,  
       DISPLAYNAME,  
       SCOPE,  
       SHAREDATETIME,  
       WEBURL,  
       ALTITUDE,  
       LATITUDE,  
       LONGITUDE,  
       ID3,
```



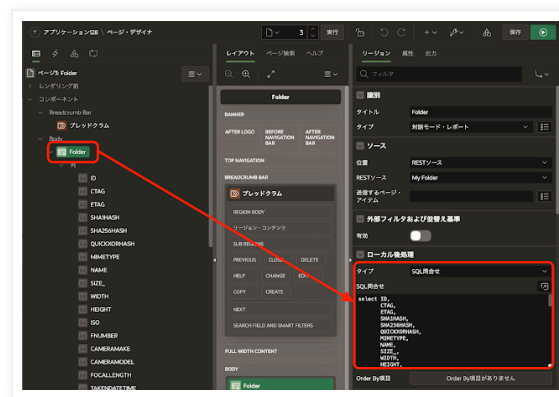
```

DISPLAYNAME2,
ID4,
DISPLAYNAME3,
COMMENTCOUNT,
CREATEDDATETIME,
LASTMODIFIEDDATETIME,
ID5,
DISPLAYNAME4,
ID6,
DISPLAYNAME5,
CREATEDDATETIME2,
ID7,
NAME2,
PATH,
DRIVEID,
DRIVETYPE,
LASTMODIFIEDDATETIME2,
DOWNLOADURL
),apex_page.get_url(
    p_page => :APP_PAGE_ID
    ,p_items => 'P3_ITEM_ID'
    ,p_values => ID
) L_FOLDER_LINK
from #APEX$SOURCE_DATA#

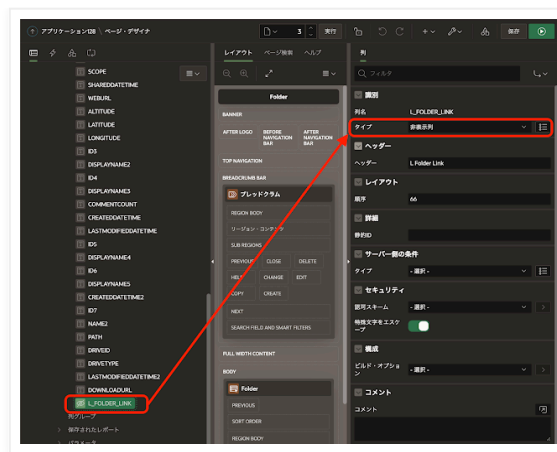
```

modify-local-folder.sql hosted with ❤ by GitHub

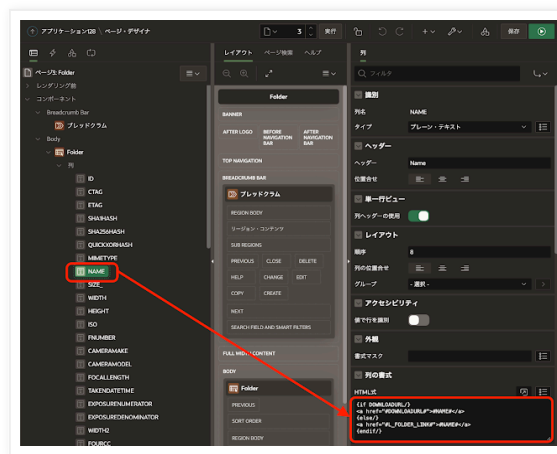
[view raw](#)



列L_FOLDER_LINKの識別のタイプを非表示に変更します。



列**NAME**の列の書式のHTML式を記述します。記述する内容はレポートMy Rootの列NAMEと同一です。



以上でRESTデータ・ソースの作成と、作成したRESTデータ・ソースを使用した対話モード・レポートの作成は完了です。

次の記事では、ファイルの作成、更新（アップロード）と削除を実装します。

続く

Yuji N. 時刻: 17:44

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

