

日々はOracle APEX

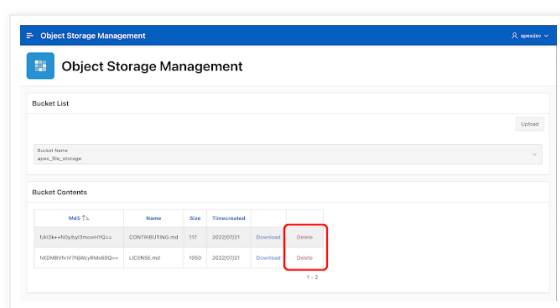
Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2020年2月28日 金曜日

APEXからOCIオブジェクト・ストレージを操作する(9) - ファイルの削除

OCIオブジェクト・ストレージ上にあるオブジェクトを削除するリンクを、ホーム画面のレポートに追加します。

元ブログの内容をAPEX 22.2のActionインターフェースを使う実装に変えています。



オブジェクトを削除するリンクの追加

ダウンロード・リンクと同様に、仮想列をリージョンBucket Contentsに追加します。手順は同じです。

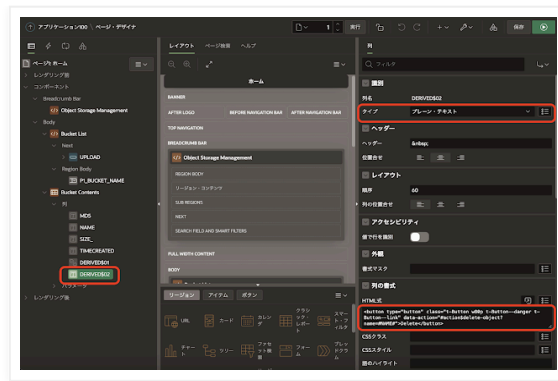
仮想列DERIVED\$02が作成されるので、その識別のタイプとしてプレーン・テキストを指定し、式の書式のHTML式に以下を設定します。

```
<button type="button" class="t-Button w80p t-Button--danger t-Button--link" data-action="#action
```

delete-object-button.html hosted with ❤ by GitHub

[view raw](#)

カスタム属性data-actionからアクションdelete-objectを呼び出すように定義しています。引数nameとしてオブジェクト名を渡しています。



アクションdelete-objectを定義します。以下のコードを、ページ・プロパティのJavaScriptのページ・ロード時に実行に記述します。

```
apex.actions.add([
    {
        name: "delete-object",
        action: function( event, element, args ) {
            apex.message.confirm(
                "Are you sure?",
                (okPressed) => {
                    if (okPressed) {
                        let result = apex.server.process(
                            "DELETE_OBJECT",
                            {
                                x01: $v("P1_BUCKET_NAME"),
                                x02: args.name
                            }
                        );

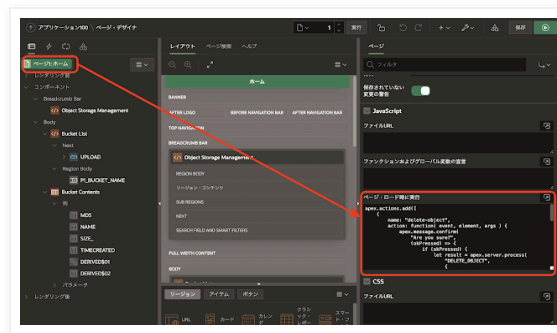
                        result.done(
                            (data) => {
                                apex.message.showPageSuccess(
                                    'Object deleted successfully.',
                                );
                                apex.event.trigger(
                                    "#R_BUCKET_CONTENTS",
                                    "apexrefresh"
                                );
                            }
                        ).fail(
                            (jqXHR, textStatus, errorThrown) => {
                                apex.message.alert('Failed to delete object.');
                            }
                        );
                    }
                }
            );
        }
    }
]);
```

```
}
```

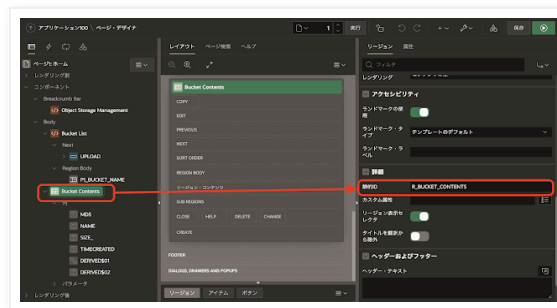
```
});
```

delete-object.js hosted with ❤️ by GitHub

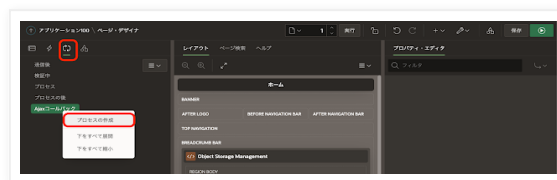
[view raw](#)



リージョンBucket Contentsの詳細の静的IDとしてR_BUCKET_CONTENTSを設定します。リフレッシュ対象のリージョンを指定するために使用します。



Ajaxのコールバックを作成するため、左ペインの表示にプロセス・ビューを選んで、Ajaxコールバックの上でコンテキスト・メニューを開き、プロセスの作成を実行します。



識別の名前としてDELETE_OBJECT(この名前で処理が呼び出されるので必ずDELETE_OBJECTを指定してください)、識別のタイプはコードを実行、ソースの位置はローカル・データベースで、PL/SQLコードとして以下を記述します。元のブログのコードは日本語の扱いに不備があったので、その部分を修正しています。

```
declare
  c_bucket_name apex_application.g_x01%type :=
    apex_application.g_x01;
  c_object_name apex_application.g_x02%type :=
    apex_application.g_x02;
  l_request_url varchar2(32767);

  l_response clob;
begin
  l_request_url := :G_BASE_URL || 'b/' || c_bucket_name
```

```

|| '/o/' || utl_url.escape(c_object_name, false, 'AL32UTF8');

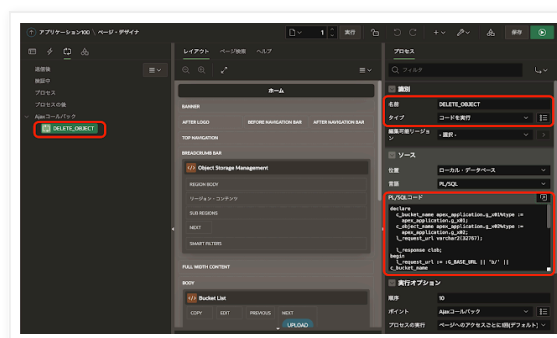
l_response := apex_web_service.make_rest_request(
  p_url => l_request_url
  , p_http_method => 'DELETE'
  , p_credential_static_id => :G_OCI_WEB_CREDENTIAL
);

if apex_web_service.g_status_code != 200 then
  owa_util.status_line(
    nstatus => apex_web_service.g_status_code
  );
  sys.http.p('{ "message": "Failed" }');
else
  sys.http.p('{ "message": "Success" }');
end if;
end;

```

deleteFile.sql hosted with ❤ by GitHub

[view raw](#)



これでオブジェクトの削除機能が実装できました。

アプリケーションを実行して、ファイルのアップロード、ダウンロード、削除などを実行してみましょう。

また、元となるブログの内容はここで終了です。

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/apex-with-oci-object-storage.zip>

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: 19:31

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.
