

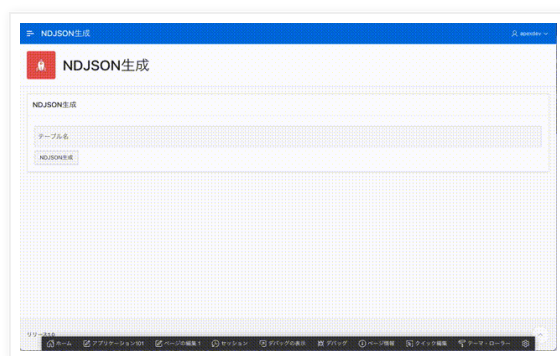
日々是Oracle APEX

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。

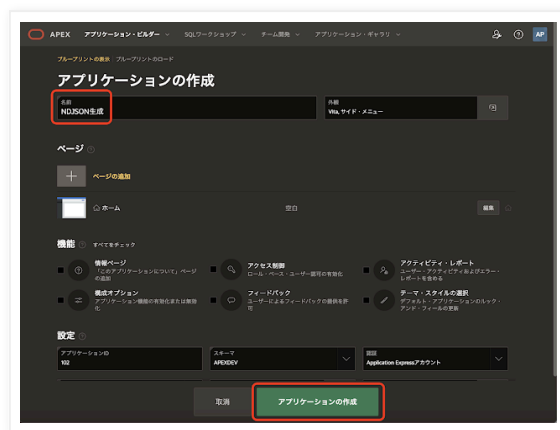
2021年9月9日 木曜日

表の内容をNDJSON形式で出力する

以前に改行で区切ったJSON形式(Newline Delimited JSON)のファイルをロードする記事を書きました。今度は逆に、指定した表の内容をNDJSON形式で出力してみます。

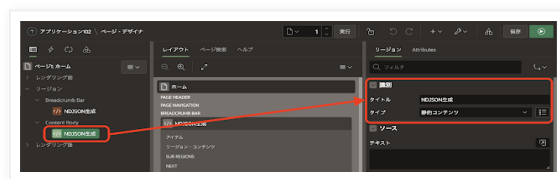


アプリケーション作成ウィザードを起動し、空のアプリケーションを作成します。アプリケーションの名前はNDJSON生成としました。アプリケーションの作成を実行します。

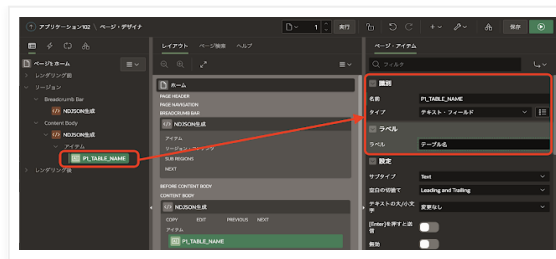


アプリケーションが作成されたら、ページ・デザイナーでホーム・ページを開きます。

Content Bodyにリージョンを作成します。識別のタイトルはNDJSON生成、タイプは静的コンテンツとします。



NDJSON形式でデータを出力する表を指定するページ・アイテムを作成します。識別の名前をP1_TABLE_NAME、タイプはテキスト・フィールドとします。ラベルはテーブル名とします。

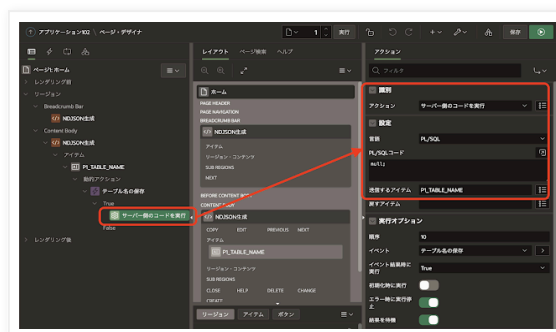


ページ・アイテムP1_TABLE_NAMEに変更があった時点で、その値をセッション・ステートに保存するよう、動的アクションの作成を行います。

識別の名前はテーブル名の保存とします。P1_TABLE_NAMEにたいして動的アクションの作成を実行していると、**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_TABLE_NAME**となっています。

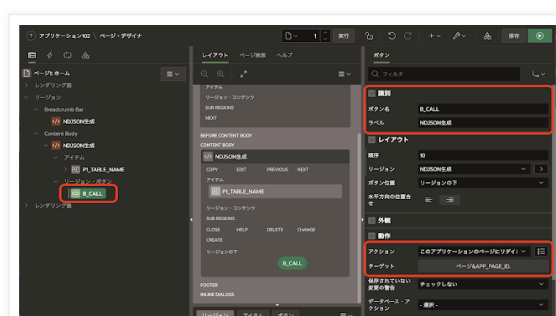


TRUEアクションの識別のアクションは**サーバー側のコードを実行**を選択します。設定のPL/SQLコードとしてはnull;を記述します。サーバー側では実際には何も実行されませんが、**送信するアイテム**として**P1_TABLE_NAME**を指定することにより、このページ・アイテムがセッション・ステートに保存されます。

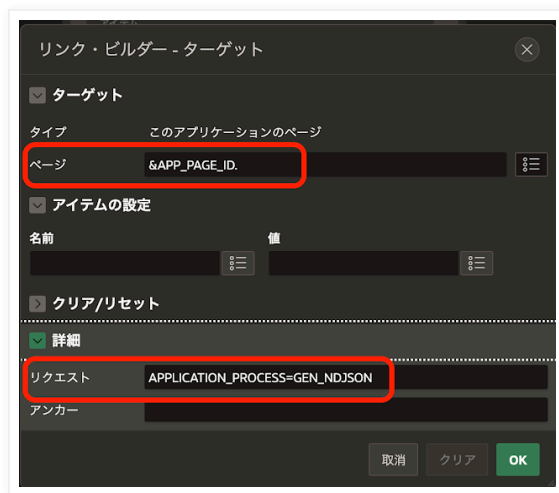


NDJSONの生成を実行するボタンを作成します。

識別のボタン名は**B_CALL**とします。ラベルは**NDJSON生成**とします。動作のアクションとしてこのアプリケーションのページにリダイレクトを選びます。



ターゲットの設定としては、ページに&APP_PAGE_ID.を指定します。&APP_PAGE_ID.は現在開いているページ番号に置き換えられます。リクエストとしてAPPLICATION_PROCESS=GEN_NDJSONを設定します。Ajaxコールバックとして作成されたプロセスGEN_NDJSONを呼び出します。

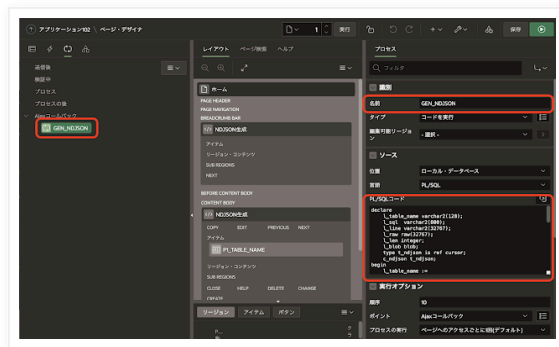


NDJSONを生成するプロセスをAjaxコールバックの位置に作成します。以下のコードを記述します。

```
declare
    l_table_name varchar2(128);
    l_sql varchar2(800);
    l_line varchar2(32767);
    l_raw raw(32767);
    l_len integer;
    l_blob blob;
    type t_ndjson is ref cursor;
    c_ndjson t_ndjson;
begin
    l_table_name := dbms_assert.sql_object_name(:P1_TABLE_NAME);
    l_sql := 'select json_object(*) as l from ' || l_table_name;
    -- 生成したNDJSON形式のデータはBLOBに書き込む。
    dbms_lob.createtemporary(l_blob, TRUE, dbms_lob.session);
    dbms_lob.open(l_blob, dbms_lob.lob_readwrite);
    -- 一行ずつJSONオブジェクトとして取り出す。
    open c_ndjson for l_sql;
    loop
        fetch c_ndjson into l_line;
        exit when c_ndjson%notfound;
        l_line := l_line || chr(10); -- 改行の追加
        l_raw := utl_raw.cast_to_raw(l_line);
        l_len := utl_raw.length(l_raw);
        dbms_lob.writeappend(l_blob, l_len, l_raw); -- BLOBに追記
    end loop;
    close c_ndjson;
    dbms_lob.close(l_blob);
    -- NDJSONの書き込みの終了。
    -- 書き込んだLOBの内容をダウンロード。
    sys.http.init;
    sys.http.p('Content-Length: ' || dbms_lob.getlength(l_blob));
    sys.http.p('Content-Disposition: attachment; filename=' || l_table_name || '.json');
    sys.owa_util.http_header_close;
    sys.wpg_docload.download_file(l_blob);
    apex_application.stop_apex_engine;
```

```
dbms_lob.free_temporary(l_blob);  
end;
```

ページ・アイテムP1_TABLE_NAMEを使ってSQLを構築するため、SQLインジェクションの対応としてDBMS_ASSERT.SQL_OBJECT_NAMEで囲み、安全な文字列であることを保証しています。またSQLの中でjson_object(*)という記法を使っています。これはOracle Database 19c以降で利用可能です。それ以前(例えば現行のapex.oracle.comは18cなので、このままでは動作しません)のバージョンでは、この部分を書き換える必要があります。



以上でアプリケーションの完成です。実行すると記事の先頭のGIF動画のように動きます。

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/generate-ndjson.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: 22:25

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.