

# 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6月16日 木曜日

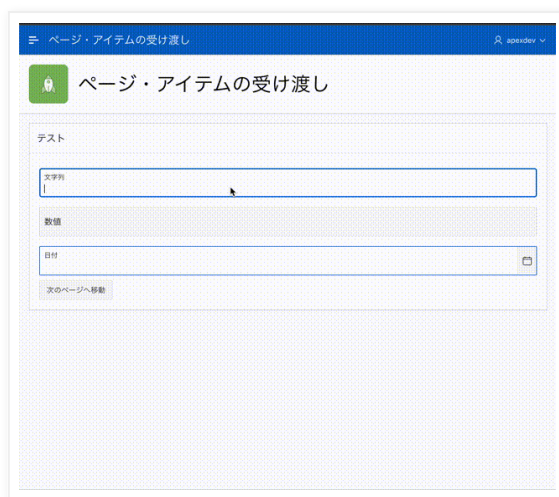
## 沢山のページ・アイテムの値を移動先のページに受け渡す

ページを遷移する際に、元のページに沢山あるページ・アイテムの値を次のページに渡す良い方法は、という相談がありました。

多分、コレクション（[APEX\\_COLLECTION](#)）を介してページ・アイテムの値を受け渡すのが、一番容易な方法でしょう。

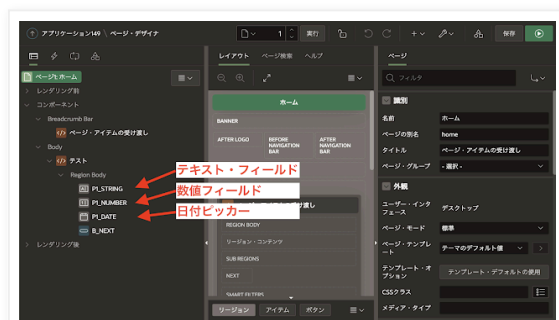
以下、簡単なサンプルを実装してみました。

遷移先のページは、元ページをコピーして作成したので見た目がほとんど同じですが、ページ遷移をした後のページ・アイテムに同じ値が設定されていることが確認できます。



空のアプリケーション（名前はページ・アイテムの受け渡しとしました）を作成します。

ホーム・ページにタイプが静的コンテンツのリージョンを作成し、ページ・アイテムを3つ作成します。ページ・アイテムP1\_STRINGはタイプがテキスト・フィールド、P1\_NUMBERはタイプが数値フィールド、P1\_DATEはタイプが日付ピッカーです。



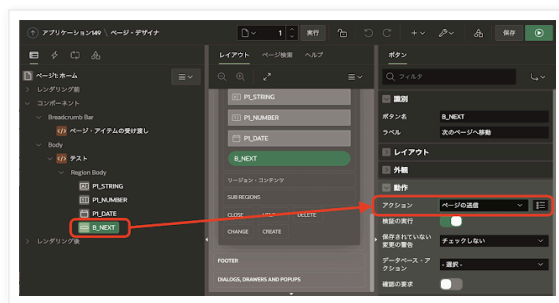
このページをコピーして、移動先のページを作成します。

ページ番号は**2**とします。ページ・アイテムの名前のプレフィックスが**P1\_**から**P2\_**へ変更されます。



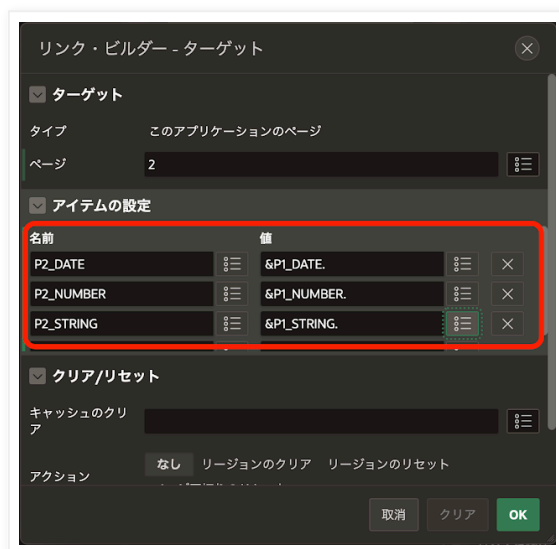
元のページから次のページへ、移動するためのボタンを作成します。

識別の名前は**B\_NEXT**、ラベルは**次のページへ移動**とします。動作のアクションには**ページの送信**を選択します。



ボタンをクリックしてページを移動する場合、**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**を選択することが多いです。引き渡す値は、**リンク・ビルダー・ターゲットのアイテムの設定**に定義します。

例えば以下のような設定になります。今回の例では、**アクション**として**ページの送信**を選択するため、この設定は行いません。



このアクションを選択していると、ボタンを押した時にHTTPのGETリクエストが発行されます。ページが表示されるときに、そのときのP1\_STRING、P1\_NUMBER、P1\_DATEの値を使って、移動先の

URLが生成されます。ページが生成されるときにページ・アイテムの値が未設定であれば、生成されるリンクの値も未設定になります。

```
/ords/r/apexdev/transfer-page-items/target?  
p2_date=&p2_number=&p2_string=&session=111004570014402&cs=1ZxL6vkecnEat5QZYcmAy0E  
mIlgHF1xtyOpjy5001RGOFltrXUeyikviuqiFBtxiXdk25iSCsglj6SwGiwjxrVQ
```

ページが表示された後の変更は、ボタンに含まれている宛先のURLには反映されません。

**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**が想定しているのは、表の一覧のレポートから、データを変更するためのフォームを開くといった、主キーを引数として別ページを開くようなページの遷移です。

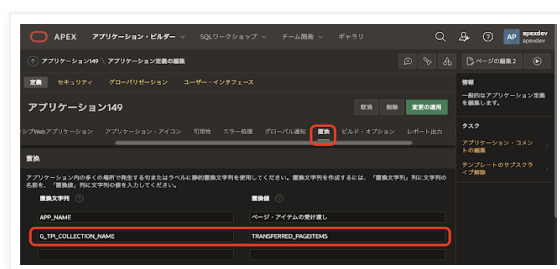
フォームのページから、別のページのフォームに遷移する場合は、ボタンの**アクションにページの送信**を選択します。アクションがページの送信の場合は、HTTPのPOSTリクエストが発行されます。ページに含まれるすべてのページ・アイテムの値はPOSTリクエストのコンテンツとして、サーバーに送信されます。そのためサーバーは、ボタンを押した時点のページ・アイテムの値を受信します。

以下より、APEXコレクションを使ったページ・アイテムの値の受け渡しを実装します。

最初にページ・アイテムの受け渡しに使用するAPEXコレクションの名前を、**アプリケーション定義の置換**として定義します。

**置換文字列**としてG\_TPI\_COLLECTION\_NAME、**置換値**をTRANSFERRED\_PAGEITEMSとします。

コレクション名はアプリケーションの各所で参照されることがあるので、一箇所で変更できるようにしておきます。また、コレクション名を間違えると原因を見つけるのが大変なので、直書きはせず常にバインド変数を介して指定します。



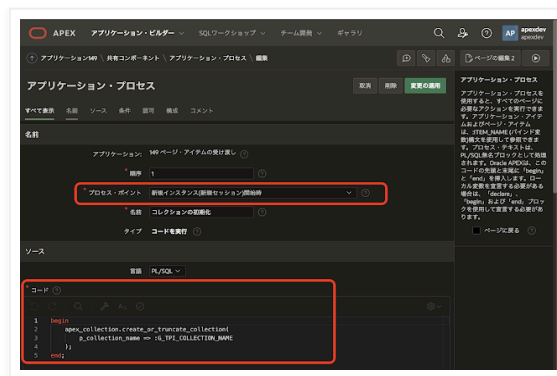
**共有コンポーネントのアプリケーション・アイテム**としてG\_TPI\_SEQ\_IDを作成します。このアプリケーション・アイテムは、コレクションのSEQ\_IDを保存するために使用します。

セッション開始時にコレクションを初期化します。

**共有コンポーネントのアプリケーション・プロセス**を作成します。**プロセス・ポイント**は**新規インスタンス(新規セッション)開始時**、実行するコードは以下になります。

**APEX\_COLLECTION.CREATE\_OR\_TRUNCATE\_COLLECTION**を呼び出します。

```
begin  
  apex_collection.create_or_truncate_collection(  
    p_collection_name => :G_TPI_COLLECTION_NAME  
  );  
end;
```

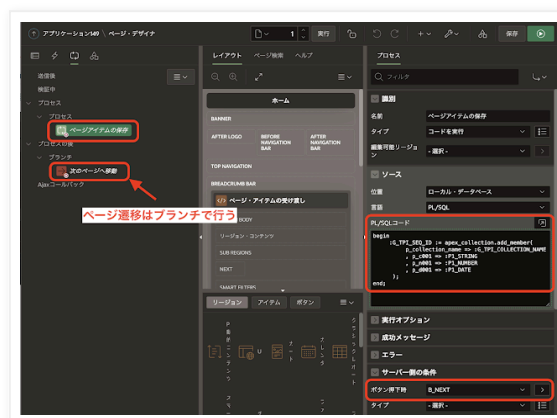


ページ・アイテムを送信するページに、ページ・プロセスを作成します。

実行するコードは以下になります。APEX\_COLLECTION.ADD\_MEMBERを呼び出します。受け渡すページ・アイテムの数だけ、指定する引数を追加します。コレクションにはVARCHAR2、NUMBER、DATE型などで値を保存することができます。ページ・アイテムが保持しているデータは基本的に文字列ですが、コレクションに保存する際に型を適切に選択するとTO\_CHARやTO\_NUMBER関数の呼び出しが不要になります。

ページの遷移はブランチで行います。

```
begin
  :G_TPI_SEQ_ID := apex_collection.add_member(
    p_collection_name => :G_TPI_COLLECTION_NAME
    , p_c001 => :P1_STRING
    , p_n001 => :P1_NUMBER
    , p_d001 => :P1_DATE
  );
end;
```



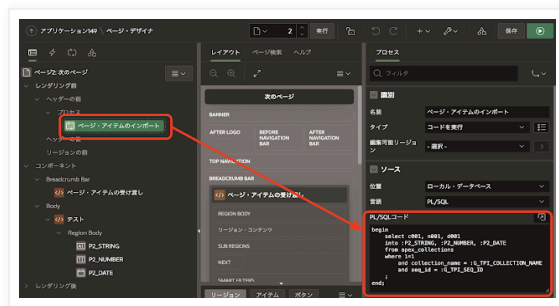
ページ・アイテムを受信するページに、ページ・プロセスを作成します。

プロセスはページのレンダリング前のヘッダーの前に実行します。APEX\_COLLECTIONSビューを検索し、値を取り出します。

実行するコードは以下になります。

```
begin
  select c001, n001, d001
  into :P2_STRING, :P2_NUMBER, :P2_DATE
  from apex_collections
  where 1=1
  and collection_name = :G_TPI_COLLECTION_NAME
  and seq_id = :G_TPI_SEQ_ID;
```

;
end;



APEXコレクションを使用したページ・アイテムの値の受け渡し方法の実装は以上になります。

今回作成したAPEXアプリケーションにエクスポートを以下に置きました。  
<https://github.com/ujnak/apexapps/blob/master/exports/transfer-page-items.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: 14:23

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.