

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10月26日水曜日

ページ・アイテムに含まれるコロンの扱いについて

ページ・アイテムの値に含まれるコロンの扱いについて、調べてみました。

ふたつのページ・アイテムP1_DATE1、P1_DATE2があり、送信ボタンでページを送信します（時刻以外のデータを後で投入するため、タイプはテキスト・フィールドとします）。

ブランチを作成して別ページに遷移します。ページ・アイテムの値を引き継ぐように、**ブランチのターゲット**を設定します。

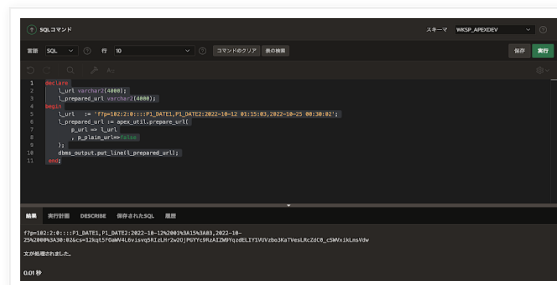
結果を確認します。ページ・アイテムP1_DATE2の秒の値が引き継がれていません。

内部動作を確認してみます。以下のPL/SQLスクリプトをSQLコマンドにて実行します。

```

declare
  l_url varchar2(4000);
  l_prepared_url varchar2(4000);
begin
  l_url := 'f?p=102:2:0::::P1_DATE1,P1_DATE2:2022-10-12 01:15:03,2022-10-25 00:30:02';
  l_prepared_url := apex_util.prepare_url(
    p_url => l_url
    , p_plain_url=>false
  );
  dbms_output.put_line(l_prepared_url);
end;

```



APEX_UTIL.PREPARE_URLのP_URLに与える値は以下です。

f?p=102:2:0::::P1_DATE1,P1_DATE2:2022-10-12 01:15:03,2022-10-25 00:30:02

以下の結果が得られます。

f?p=102:2:0::::P1_DATE1,P1_DATE2:2022-10-12%2001%3A15%3A03,2022-10-25%2000%3A30:02&cs=12kqt5FOaWV4L6visvq5RIzLHr2w2OjPGYYc9RzAlZW9YqzdELIY1VUVzbo3KaTVesLRcZdC0_cSWVxikLnsVdw

チェックサムを除くと以下になります。

f?p=102:2:0::::P1_DATE1,P1_DATE2:2022-10-12%2001%3A15%3A03,2022-10-25%2000%3A30:02

マニュアルでは、ページ・アイテムにカンマが含まれる可能性がある場合は、バックスラッシュ(\)でページ・アイテムの値を囲むように案内されています。

https://docs.oracle.com/cd/F70953_01/html/db/understanding-legacy-url-syntax.html#GUID-D5510BFC-AF94-4AAD-8B82-FF780A47601D

itemValues	<p>URLでセッション・ステートを設定するために使用されるアイテム値のリストです。アイテム値にカンマを渡すには、文字をバックスラッシュで囲んでください。次に例を示します。</p> <div> <pre>\123,45\</pre> </div> <p>backslash comma (\,)以外のすべての文字列は、バックスラッシュで囲むことができます。</p>
------------	--

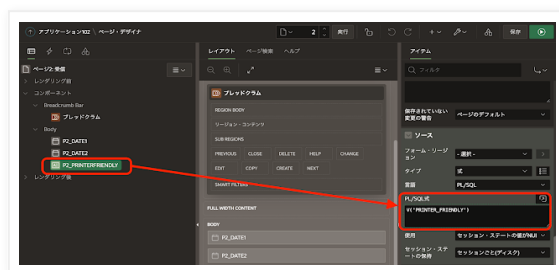
しかし、コロンについてはそのような記述はありません。また、カンマとは異なり、つねに区切り文字として扱われているわけでもありません。ページ・アイテムP1_DATE1については、秒の値も引き継がれています。

調べたところ、**最後のコロン以降はPrinterFriendlyとして認識する**ことが理由でした。

PrinterFriendly	<p>ページを「印刷用」モードでレンダリングするかどうかを決定します。PrinterFriendlyが「はい」に設定されている場合は、ページが「印刷用」モードでレンダリングされます。PrinterFriendlyの値は、リージョンなどの要素をページから削除して印刷される出力を最適化するための条件のレンダリングに使用されます。</p> <p>印刷用プリファレンスを参照するには、次の構文を使用します。</p> <pre>V('PRINTER_FRIENDLY')</pre> <p>参照すると、印刷用テンプレートを使用してページが表示されます。APEXエンジンは、HTMLフォーム・フィールド内のすべてのテキストをテキストとして表示します。印刷用テンプレートには、#FORM_OPEN#タグまたは#FORM_CLOSE#タグは必要ありません。目的は、表をほとんど使用しないで、印刷に適した書式で情報を表示できるようにすることです。</p>
-----------------	---

ターゲットのページでPrinterFriendlyの値を確認するために、ページ・アイテムを追加します。**ソースのタイプ**を式として、**PL/SQL式**に以下を記述します。

V('PRINTER_FRIENDLY')



ボタンをひとつ追加し、こちらのボタンで実行されるブランチのターゲットは以下のように、ページ・アイテムをバックスラッシュで囲みます。

リンク・ビルダー - ターゲット

ターゲット

タイプ: このアプリケーションのページ

ページ: 2

アイテムの設定

名前: P2_DATE1

値: \\\$P1_DATE1\\

クリア/リセット

キャッシュのクリア: なし

アクション: リージョンのクリア, リージョンのリセット, ページ区切りのリセット

取消 クリア OK

バックスラッシュ付きのブランチを呼び出してみます。

ページ・アイテムの区切り文字

Date1: 2022-10-12 01:15:03

Date2: 2022-10-25 00:30:02

送信

バックスラッシュ付き送信

今度はP1_DATE2の秒の値も正しく渡されています。

受信

Date1: 2022-10-12 01:15:03

Date2: 2022-10-25 00:30:02

Printer Friendly: NO

秒が渡されている

P1_DATE2の秒の部分がPrinterFriendlyとして扱われていることを確認するため、秒の部分にYESを記載して送信してみます。

ページ・アイテムの区切り文字

Date1: 2022-10-12 01:15:03

Date2: 2022-10-25 00:30:YES

送信

バックスラッシュ付き送信

P1_DATE2の秒の値は00になりますが、Printer FriendlyがYESになっていることが確認できます。

受信

Date1: 2022-10-12 01:15:03

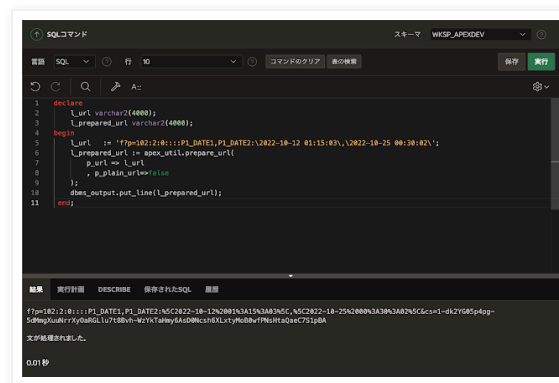
Date2: 2022-10-25 00:30:00

Printer Friendly: YES

秒は00でPrinter FriendlyはYES

先ほどのPL/SQLの確認スクリプトのP_URLを、以下にして実行してみます。

f?p=102:2:0::::P1_DATE1,P1_DATE2:\2022-10-12 01:15:03\,\2022-10-25 00:30:02\



チェックサムの部分を除くと、以下のURLが返されます。

f?p=102:2:0::::P1_DATE1,P1_DATE2:%5C2022-10-12%2001%3A15%3A03%5C,%5C2022-10-25%2000%3A30%3A02%5C

元々は以下なので、:02の部分を含んだ形で値がエスケープされていることがわかります。

f?p=102:2:0::::P1_DATE1,P1_DATE2:2022-10-12%2001%3A15%3A03,2022-10-25%2000%3A30:02

もともとOracle APEXのURLがf?p構文のときは、コロンがセパレータとして使用されていました。簡易URLが導入されてからコロンはセパレータとして現れませんが、今回の事象は簡易URLがONでも発生します。

簡単なアプリケーションですが、今回の検証に使用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/page-item-separator.sql>

以上になります。

完

Yuji N. 時刻: 17:35

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.
