

日日是Oracle APEX

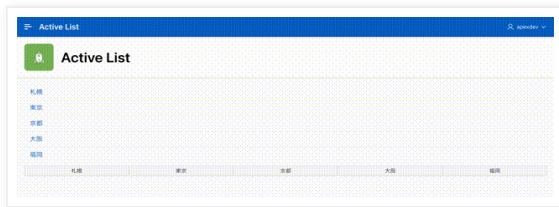
Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2024年2月8日木曜日

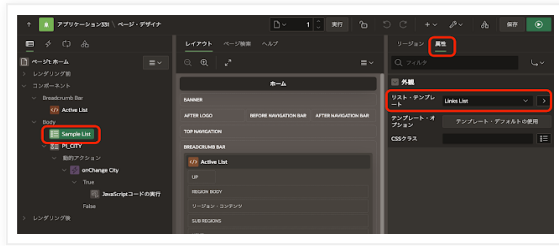
リスト・リージョン上のアイテムをクリックして選択する

リスト・リージョンに表示されているリスト・アイテムをクリックして、そのアイテムが選択されたように見せる実装をしてみました。また、タイプがラジオ・グループのページ・アイテムをボタンで表示し、双方の設定が同期するような実装も加えています。

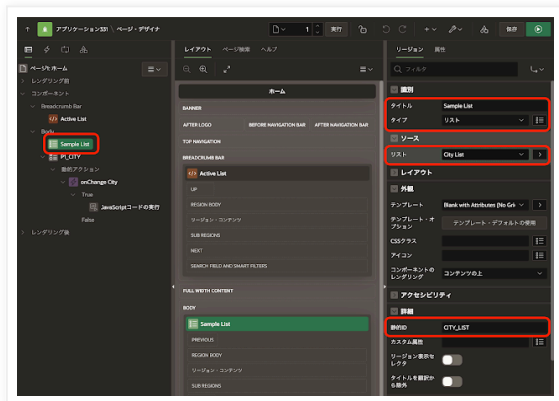
以下のように動作します。



リスト・リージョンの**外観のリスト・テンプレート**として**Links List**を選んでいきます。今回の実装はLinks Listのテンプレートを対象した実装になります。



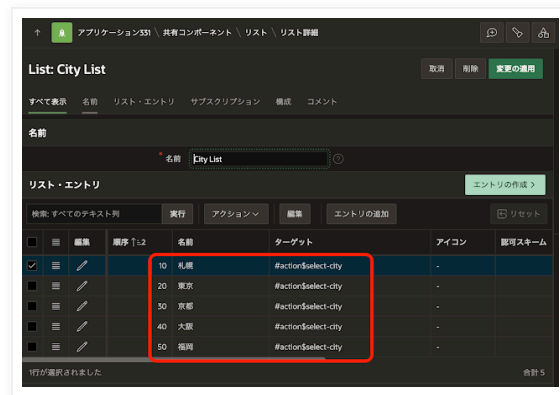
ページに**タイプ**が**リスト**のリージョンを作成しています。ソースのリストとして**City List**を選択しています。JavaScriptのコードよりリージョンを参照するため、詳細の静的IDに**CITY_LIST**を設定しています。



共有コンポーネントのリストに、以下のようにリストCity Listを作成しています。

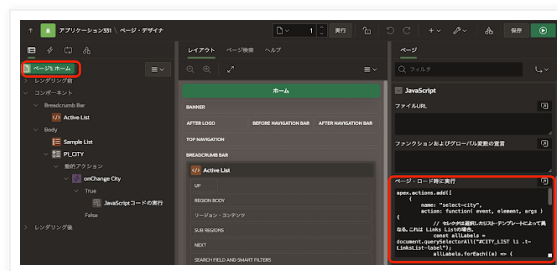
ターゲットはすべて#action\$select-cityを設定しています。この設定により、リストに表示されている要素（都市名）をクリックすると、APEXアクションのselect-cityが呼び出されます。

リストからAPEXアクションを呼び出す設定は、リスト・テンプレートに依存して変わります。今回はLinks Listですが、Menu Popupの場合は以前の記事「[メニュー・ポップアップから任意の処理を呼び出す](#)」で紹介した手順になります。



ページ・プロパティのJavaScriptのページ・ロード時に実行に、APEXアクションの定義を記述します。

```
apex.actions.add([
  {
    name: "select-city",
    action: function( event, element, args ) {
      // セレクタは選択したリスト・テンプレートによって異なる。これは Links Listの場合。
      const allLabels = document.querySelectorAll("#CITY_LIST li .t-LinksList-label");
      allLabels.forEach((e) => {
        // elementとして返されるのはli要素。
        if (element == e) {
          /*
           * クリックされた要素であれば、P1_CITYに都市名を設定し、
           * 表示を変えるためにu-dangerクラスを追加している。
           * 選択時のリンクの見せ方を変えるにはu-dangerのクラスを変える。
           */
          apex.items.P1_CITY.setValue(e.textContent, e.textContent, true);
          e.classList.add("u-danger");
        }
        else
        {
          /*
           * 一致していないli要素からはu-dangerをなく。
           */
          e.classList.remove("u-danger");
        }
      })
    }
  }
]);
```



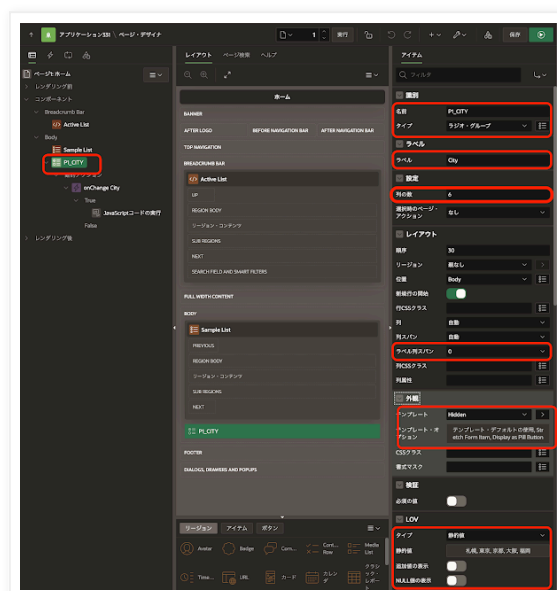
選択した都市名を保持するページ・アイテムとしてP1_CITYを作成しています。選択肢の内の1つを選択するため、**タイプ**に**ラジオ・グループ**を選びます。

ラベルには**City**と設定していますが、**外観のテンプレート**に**Hidden**を選択しているため、ラベルは表示されません。ラベルは表示されませんが、デフォルトではラベルを表示するための領域が取られるため、**レイアウトのラベル列スパン**を**0**にします。これでページ・アイテムの値が左端に寄ります。

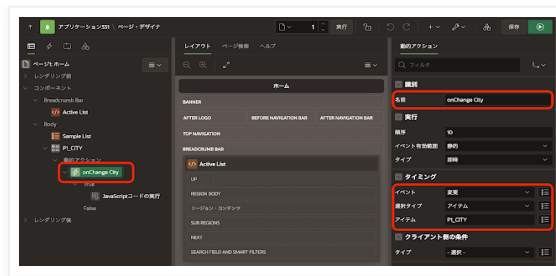
設定の列の数に**6**を設定します。都市の数が5つなので、6であれば表示が1行に収まります。

テンプレート・オプションを開き、一般の**Stretch Form Item**に**チェック**を入れ、**詳細のItem Group Display**に**Display as Pill Button**を選択します。これで、画面の左右いっぱいの幅に、かつ形状がボタンで表示されるようになります。

リストを静的値で設定しているため、ページ・アイテムの**LOV**を**静的値**で設定します。**NULL値の表示**は**オフ**にします。オンのときは、NULLの選択がラジオ・ボタンの選択肢として追加されます。



ラジオ・グループの選択を変えたときに、リストの選択を変える動的アクションを作成します。**動的アクションのタイミングのイベント**は、ページ・アイテムのデフォルトである**変更**です。

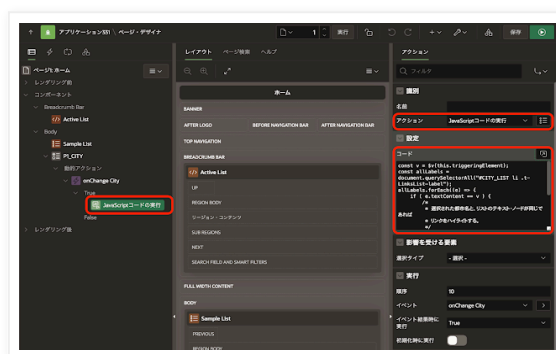


TRUEアクションとして、以下のJavaScriptのコードを実行します。

```
const v = $(this.triggeringElement);
const allLabels = document.querySelectorAll("#CITY_LIST li .t-LinksList-label");
allLabels.forEach((e) => {
    if ( e.textContent == v ) {
        /*
         * 選択された都市名と、リストのテキスト・ノードが同じであれば
         * リンクをハイライトする。
         */
        e.classList.add("u-danger");
    }
    else
    {
        /*
         * 一致していないli要素からはu-dangerを除外。
         */
        e.classList.remove("u-danger");
    }
});
```

change-selected-list-item-by-radio-button.js hosted with ❤ by GitHub

[view raw](#)



今回の実装の紹介は以上です。

今回の記事で扱ったAPEX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/select-list-item-on-click.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

[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.