

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12月13日 火曜日

カード上のボタンをクリックしてJavaScriptを実行する

カード・リージョンでは、個々のカードにたいして**アクション**を作成できます。作成できるアクションのタイプは以下の3つで、動的アクションはありません。

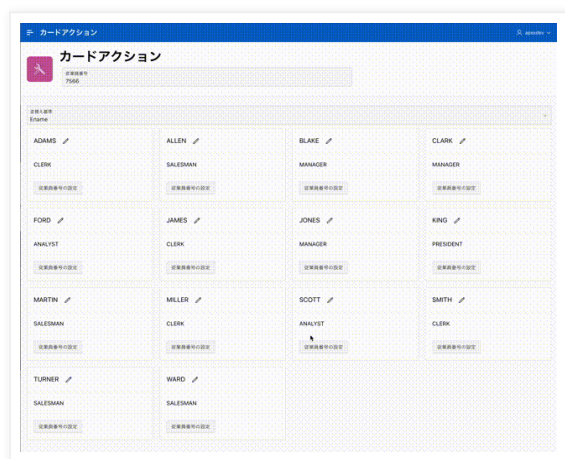
1. このアプリケーションのページにリダイレクト
2. 別のアプリケーションのページにリダイレクト
3. URLにリダイレクト

APEX 22.2よりインターフェース**actions**が追加され、ボタンやリンクのクリックからapex.actionsに登録したJavaScriptのコードが実行できるようになりました。

以下より、actionsインターフェースを使って、カード上のボタンをクリックしてJavaScriptを呼び出す実装を行なってみます。

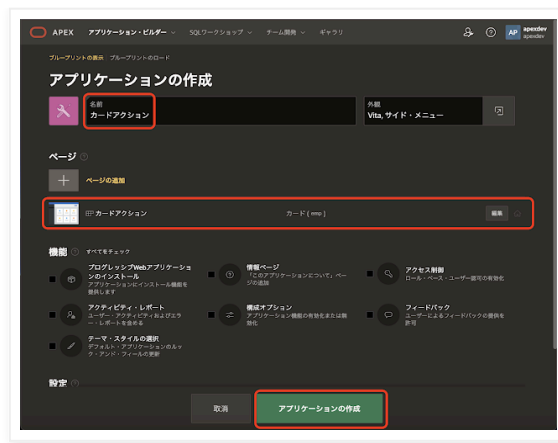
サンプル・データセットのEMP/DEPTをデータ・ソースとして使い、以下の動作を行うAPEXアプリケーションを作成します。

従業員名に隣にある**鉛筆アイコン**をクリックするか、または**ボタンの従業員番号の設定**をクリックすると、従業員番号の**テキスト・フィールド**に従業員番号を設定します。



アプリケーション作成ウィザードを起動します。アプリケーションの名前は**カードアクション**としました。

デフォルトで作成されている**ホーム・ページ**を（**編集をクリックして**）削除し、**ページの追加**をクリックして**カードのページ**を追加します。

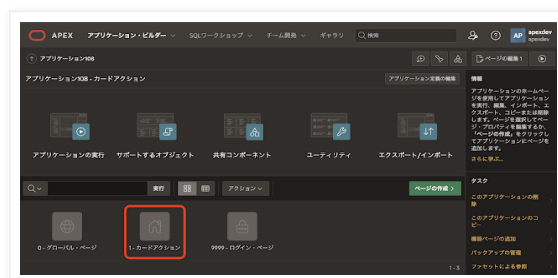


追加するカードのページ名もカードアクションとします。表またはビューとしてEMPを選択し、レイアウトはグリッドを選択します。タイトル列はENAME、本体列はJOBを指定し、ページの追加をクリックします。



以上でアプリケーションの作成を実行します。

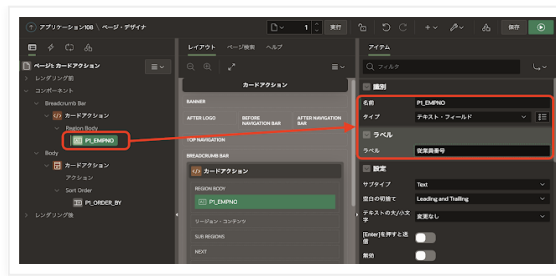
アプリケーションが作成されます。これから、カードアクションのページにアクションを実装していきます。



ページ・デザイナーにてページ番号1のカードアクションを開きます。

従業員番号を設定するページ・アイテムを、Breadcrumb Barのカードアクションのリージョンに作成します。

識別の名前はP1_EMPNOです。タイプとしてテキスト・フィールドを選択します。ラベルは従業員番号とします。



ページ・プロパティのJavaScriptのファンクションおよびグローバル変数の宣言に、ボタンのクリックで呼び出されるAPEXアクションを記述します。引数empnoに渡された従業員番号をページ・アイテムP1_EMPNOに設定します。

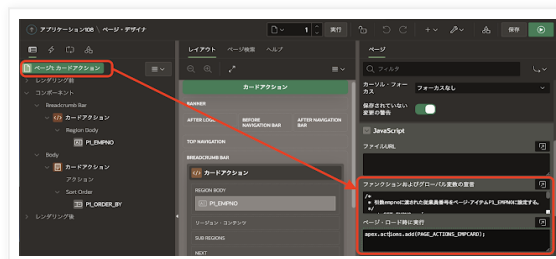
```
/*
 * 引数empnoに渡された従業員番号をページ・アイテムP1_EMPNOに設定する。
 */
const SET_EMPNO = {
  name: "setempno",
  action: function(event, element, args) {
    apex.item("P1_EMPNO").setValue(args.empno);
  }
};
// apex.actions.addへ渡す。
const PAGE_ACTIONS_EMPCARD = [ SET_EMPNO ];
```

set_empno.js hosted with ❤ by GitHub

[view raw](#)

ページ・ロード時に実行に以下の一行を記述します。

```
apex.actions.add(PAGE_ACTIONS_EMPCARD);
```

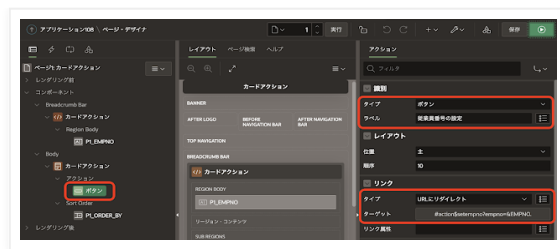


カードのアクションとしてボタン従業員番号の設定を作成します。

識別のタイプはボタン、ラベルは従業員番号の設定とします。リンクのタイプにURLにリダイレクトを選択し、ターゲットとして以下を記述します。

```
#action$setempno?empno=&EMPNO.
```

このターゲットの指定により、APEXアクションとして登録されているsetempnoが呼び出され、従業員番号がページ・アイテムP1_EMPNOに設定されます。



従業員名の横に表示されている鉛筆アイコンのボタンは、ボタン・ビルダーを使ってHTMLを生成します。

Button Builder - Universal Theme (oracle.com)

https://apex.oracle.com/pls/apex/apex_pm/r/ut/button-builder

今回の設定は以下です。Button TemplateはIcon、Button LabelはEdit (表示はされません)、Icon Classとしてfa-pencil、StyleをRemove UI Decorationsを選んでいきます。

上記の設定で生成されたEntire Markupを使用します。

生成されたマークアップをカードに設定します。

カード・リージョンの属性を開き、タイトルを拡張フォーマットに切り替えます。

HTML式として記述するマークアップには、以下のdata-actionカスタム属性を追加します。

data-action="setempno?empno=&EMPNO."

&ENAME.

```
<button type="button" title="Edit" aria-label="Edit" class="t-Button t-Button--noLabel t-Button--icon t-Button--noIcon" data-action="setempno?empno=&EMPNO."></button>
```

call-action-from-button.html hosted with ❤ by GitHub

[view raw](#)

以上でアプリケーションは完成です。

a要素では**href属性**として"**#action\$アクション名**"で呼び出すアクションを指定します。button要素ではカスタム属性**data-action="アクション名"**で呼び出すアクションを指定します。

アプリケーションを実行すると、記事の先頭のGIF動画の動作をします。

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/card-button-action.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: 9:02

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.