

対話グリッドのリフレッシュ前後で行の選択を維持する

対話グリッドは単体で編集可能なのでフォームを付けるケースはあまりありませんが（一般的には対話モード・レポートにフォームを付ける）、フォームをつけたときにフォームを閉じると行の選択がリセットされます。この動作自体は、対話グリッドの通常のリフレッシュ処理なのですが、行の選択を維持したいという相談がありました。

実装が思ったよりも大変だったので、記事にしました。

サンプル・データセットのEMP/DEPTに含まれるEMP表をデータ・ソースとした対話グリッドと、フォームを作成しました。以下のように動作します。

EMP List

EMP List

作成

Q▼		検索: すべてのデータストリ		実行		アクション▼		🔍 検索 🔍		
🔍	🔍		Ename	Job	Mgr	Hiredate	Sal	Comm	Deptno	
✓	🔍	🔍	CLARK	MANAGER	KING	1981/06/09	2450		ACCOUNTING	
	🔍	🔍	JONES	MANAGER	KING	1981/04/02	2975		RESEARCH	
🔍	🔍	🔍	BLAKE	MANAGER	KING	1981/05/01	2850		SALES	
🔍	🔍	🔍	ALLEN2	SALESMAN	BLAKE	1981/02/20	1601	300	SALES	
	🔍	🔍	WARD	SALESMAN	BLAKE	1981/02/22	1303	500	SALES	
🔍	🔍	🔍	MARTIN	SALESMAN	BLAKE	1981/09/28	133	1400	SALES	
🔍	🔍	🔍	TURNER	SALESMAN	BLAKE	1981/09/08	1520	0	SALES	
🔍	🔍	🔍	JAMES	CLERK	BLAKE	1981/12/03	1035		SALES	
✓	🔍	🔍	MILLER	CLERK	CLARK	1982/01/23	1322		ACCOUNTING	
🔍	🔍	🔍	SCOTT	ANALYST	JONES	1982/12/09	3022		RESEARCH	
🔍	🔍	🔍	FORD	ANALYST	JONES	1981/12/03	3041		RESEARCH	
🔍	🔍	🔍	ADAMS	CLERK	SCOTT	1983/01/12	1122		RESEARCH	
🔍	🔍	🔍	SMITH	CLERK	FORD	1980/12/17	800		RESEARCH	
🔍	🔍	🔍	KING	PRESIDENT		1981/11/17	5000		ACCOUNTING	

2行が選択されました

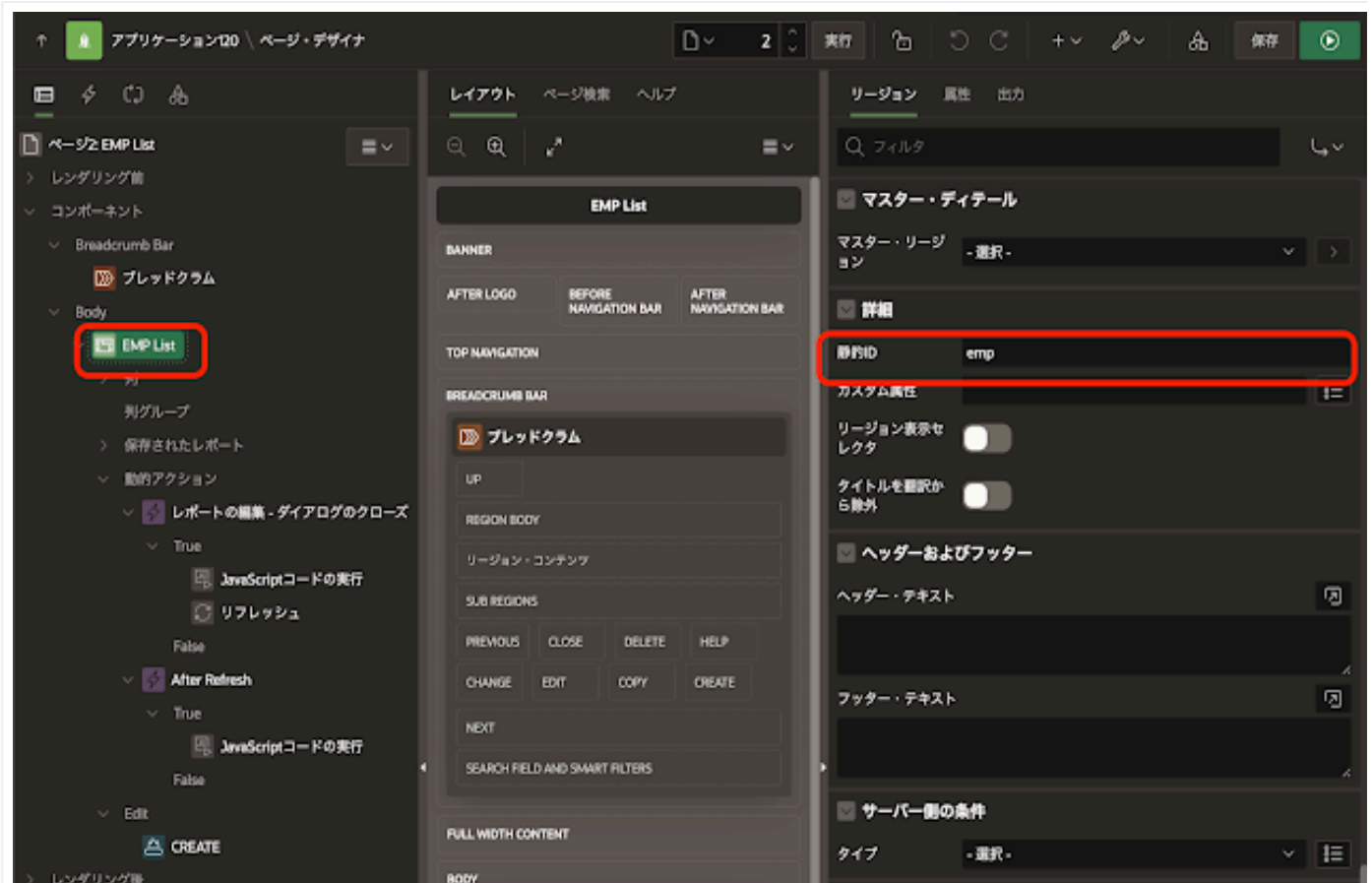
合計 14

対話グリッドを含むページのページ・プロパティに、対話グリッドで選択した行（の配列）を保持するための変数`selectedIds`を宣言します。

JavaScriptのファンクションおよびグローバル変数の宣言に以下を記述します。

```
var selectedIds;
```

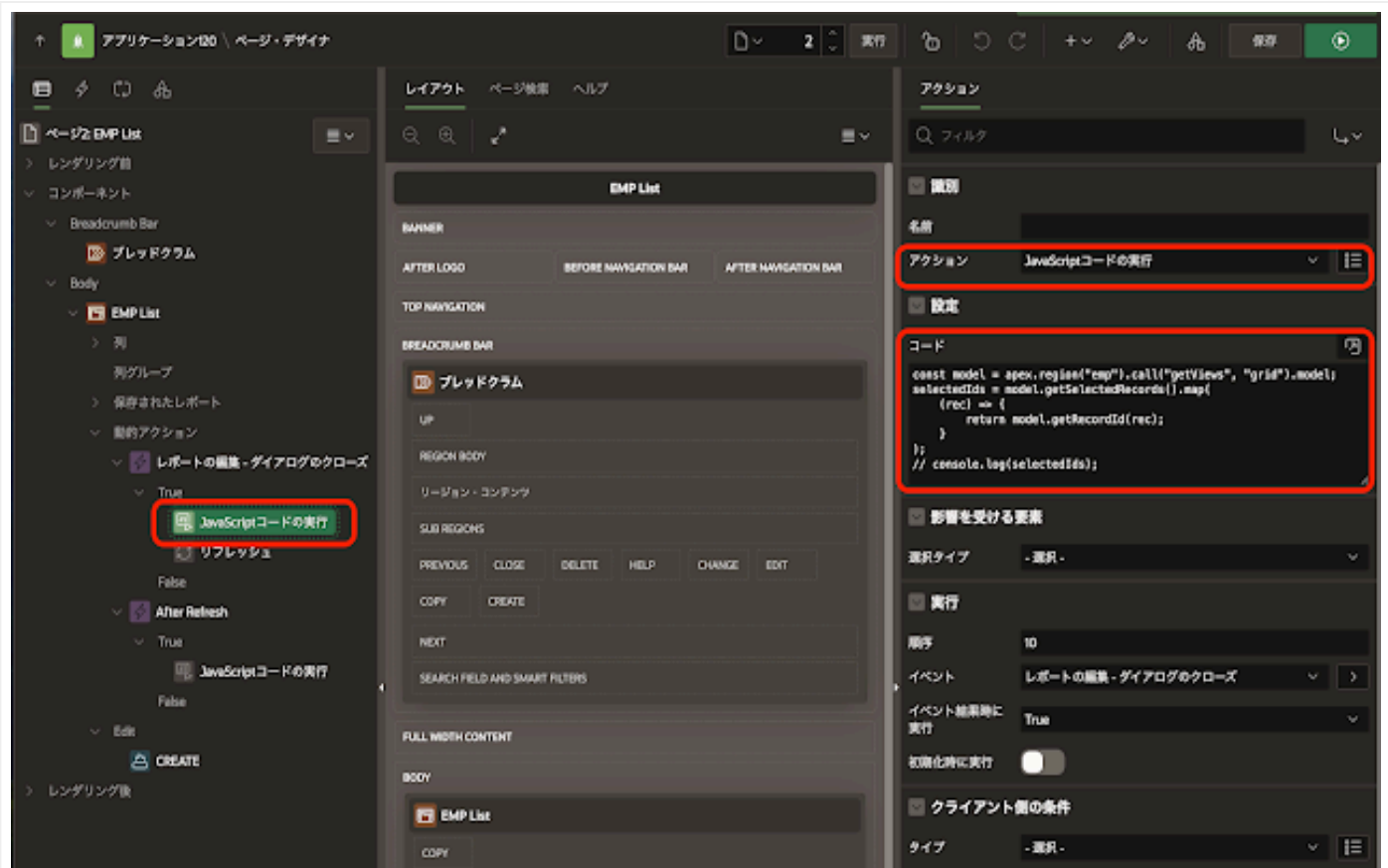
対話グリッドに静的IDとして`emp`を設定します。



イベントのダイアログのクローズを受けて、対話グリッドをリフレッシュするTRUEアクションは作成済みです。

リフレッシュの直前に、以下のJavaScriptのコードを実行するTRUEアクションを作成します。対話グリッドで選択されている行のID（主キーの値）の配列をselectedIdsに保存します。

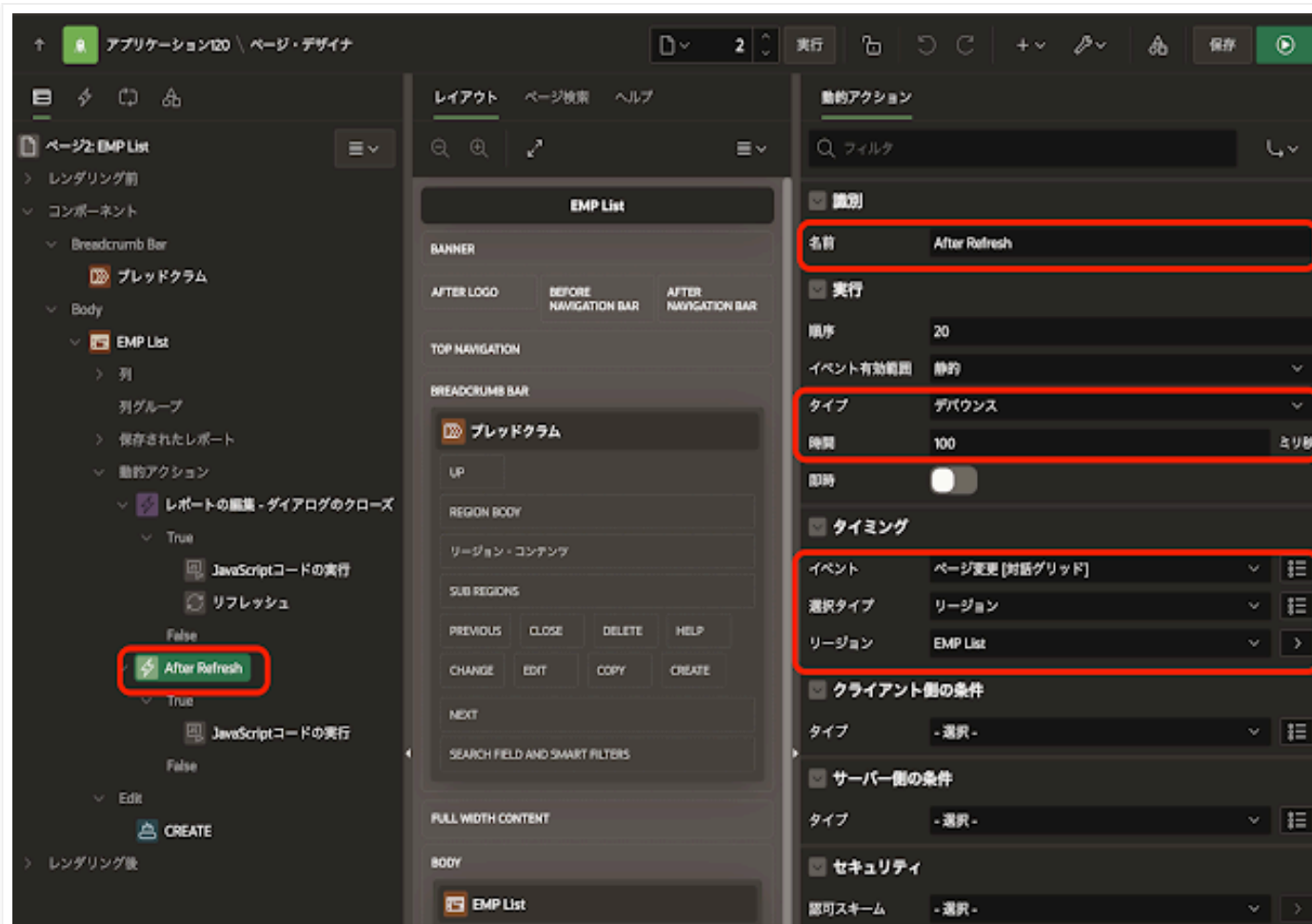
```
const model = apex.region("emp").call("getViews", "grid").model;
selectedIds = model.getSelectedRecords().map(
  (rec) => {
    return model.getRecordId(rec);
  }
);
// console.log(selectedIds);
```



対話グリッドのリフレッシュが完了した後に、**selectedIds**に保存した行を選択した状態に戻します。

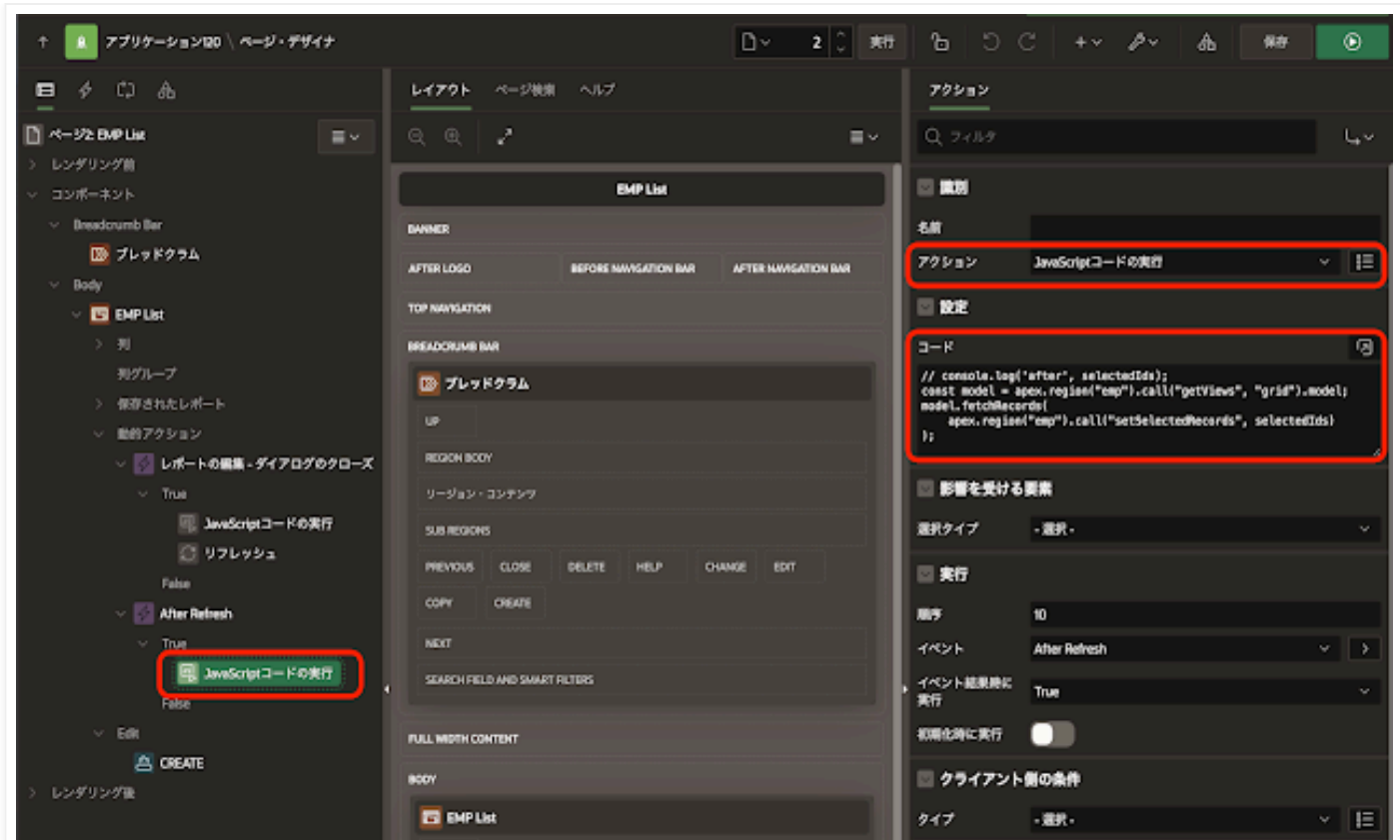
対話グリッドに動的アクションを作成します。**タイミングのイベント**はページ変更[対話グリッド]を選択します。ほとんどのリフレッシュ可能なコンポーネントは、**イベント**としてリフレッシュ後を選ぶのですが、**対話グリッドではリフレッシュ後のイベントが発生しません**。従って、それに近いイベントとして**ページ変更**を使います。

また、**実行のタイプ**として**デバウンス**を選択し、**ページ変更**のイベントが発生してから**100ミリ秒**待機したのちに処理を開始するように設定します。100ミリ秒というのは、人が気が付かないほどの短時間で、かつ、JavaScriptの処理が空振りしない程度の時間として選んでいます。



TRUEアクションとして実行するJavaScriptコードは以下になります。行の選択はfetchRecordsの処理が完了した後に実行するようにしています。

```
// console.log('after', selectedIds);
const model = apex.region("emp").call("getViews", "grid").model;
model.fetchRecords(
  apex.region("emp").call("setSelectedRecords", selectedIds)
);
```



以上で実装は完了です。

今回作成した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/preserve-selection.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: 0:15

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.