

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。

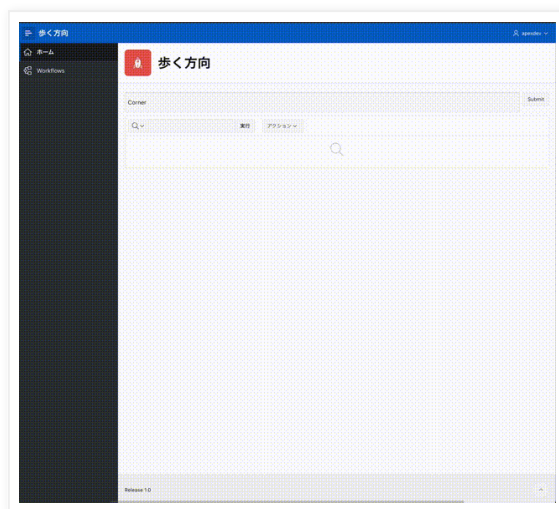
2024年2月13日 火曜日

アクション・タスクで承認と却下ではない任意の分岐を実装する

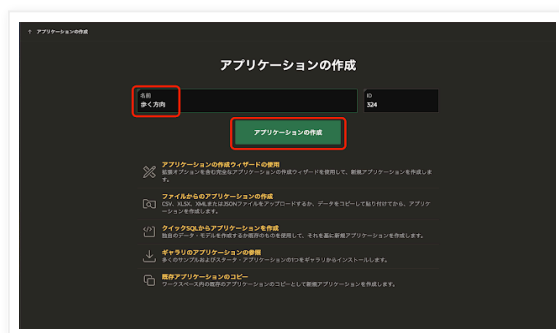
Oracle APEX 23.2のAPEXワークフローではユーザーからの入力は、**アクティビティのタイプ**の**ヒューマン・タスク**の作成、つまり**承認タスク**または**アクション・タスク**といった**タスク**より行います。ユーザーにアサインされたタスクは、**統合タスク・リスト**のページに一覧され、**承認タスク**であれば割り当たっているタスクの**承認**または**却下**、**アクション・タスク**であれば、**タスク**を完了させます。

今回の記事では、**承認**または**却下**ではない選択肢を、**アクション・タスク**を使って実装します。散歩途中の曲がり角で、**左に曲がるか**、**中心を進むか**、**右に曲がるか**、この3つの選択肢よりひとつの方向を選択します。

作成するアプリケーションは以下のように動作します。



最初に空のアプリケーションを作成します。名前は**歩く方向**とします。



アプリケーションが作成されたら、**共有コンポーネントのタスク定義**を開きます。

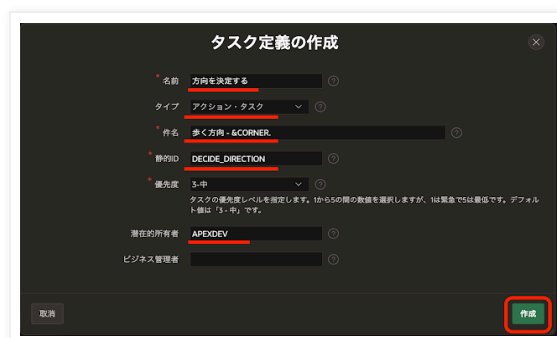


タスク定義の一覧より**作成**をクリックし、タスク定義の作成を開始します。

名前は方向を決定する、タイプは**アクション・タスク**を選択します。件名は歩く方向 - &CORNER. とします。静的IDはDECIDE_DIRECTION、潜在的所有者は以下のスクリーンショットではAPEXDEV となっていますが、現在開発作業を行なっているユーザー名を指定します。

作成したアプリケーションにサインインして動作を確認する作業も、開発しているユーザーで実施します。

作成をクリックします。



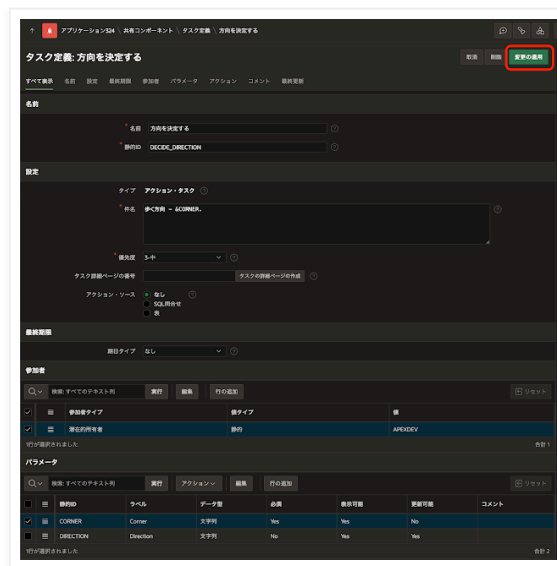
アクション・タスクが作成されたら、パラメータを2つ追加します。

最初に**件名**に含まれるパラメータとして、**CORNER**を追加します。静的IDはCORNER、ラベルはCorner、データ型は文字列、必須はYes、表示可能はYes、更新可能はNoを選択します。

次にアクション・タスクとして**選択した選択肢**を保存するパラメータとして、**DIRECTION**を追加します。静的IDはDIRECTION、ラベルはDirection、データ型は文字列、必須はNo、表示可能はYes (Noで良いかもしれませんが)、**更新可能はYesを選択します。**

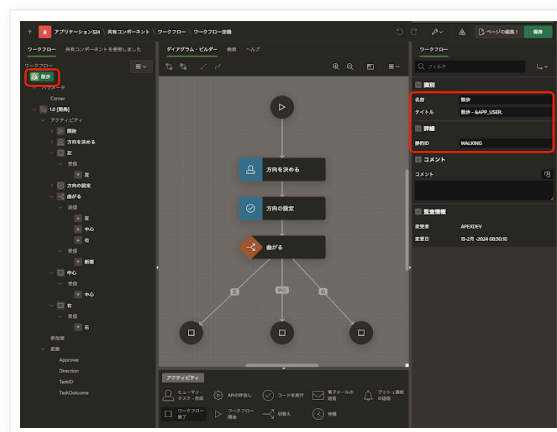


以上でタスク定義は完成です。

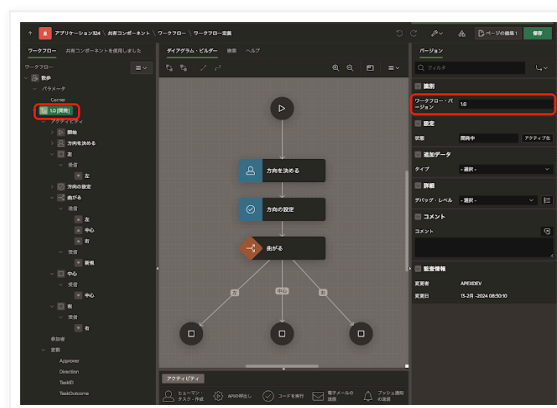


共有コンポーネントのワークフローを開き、以下のワークフローを作成します。

識別の名前は散歩、タイトルは散歩 - &APP_USER.、静的IDはWALKINGとします。



ワークフロー・バージョンは1.0とします。



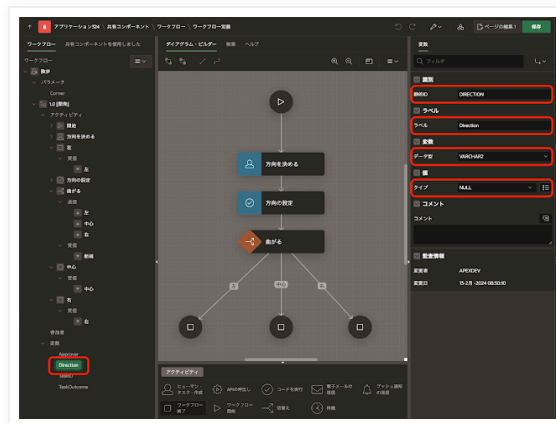
ワークフロー・パラメータとしてCornerを作成します。

識別の静的IDはCORNER、ラベルはCorner、変数のデータ型はVARCHAR2、値の必須はオンでデフォルト値は空白にします。

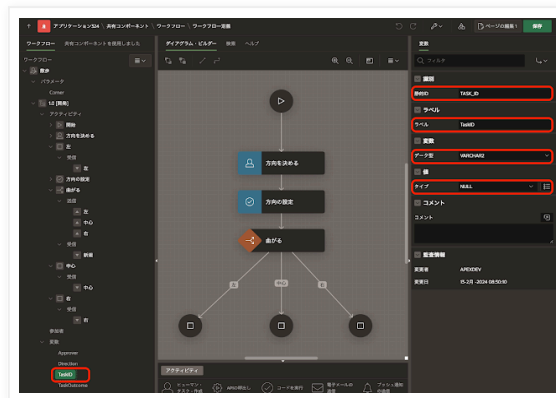


ワークフロー・バージョン変数としてDirectionとTaskIDを作成します。

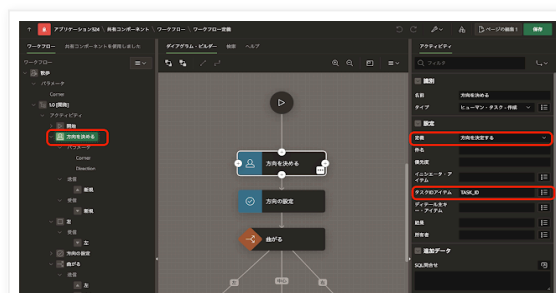
Directionは静的IDがDIRECTION、ラベルはDirection、変数のデータ型はVARCHAR2、値のタイプはNULLです。DirectionとTaskIDともにアクティビティによって値が設定されます。



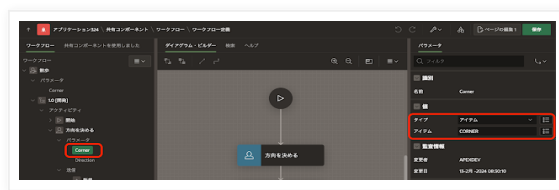
TaskIDの静的IDはTASK_IDとします。それ以外はDirectionと同じ設定です。



最初のアクティビティ方向を決めるのタイプにヒューマン・タスク - 作成を選び、設定の定義として先ほど作成したタスク定義方向を決定するを選択します。このアクティビティにより作成されたタスク（タスク定義方向を決定するを基に作成されたタスク）のタスクIDの保存先（タスクIDアイテム）として、ワークフロー・バージョン変数TASK_IDを設定します。



タスク・パラメータのCornerに、ワークフロー・パラメータのCORNERを設定します。



タスク・パラメータDirectionは入力不要なので、値のタイプにNULLを設定します。



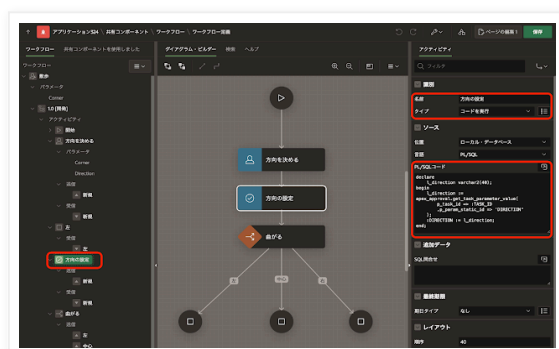
続くアクティビティ方向の設定のタイプはコードの実行です。タスク・パラメータDirectionの値を、ワークフロー・バージョン変数DIRECTIONに代入します。

ソースのPL/SQLコードに以下を記述します。

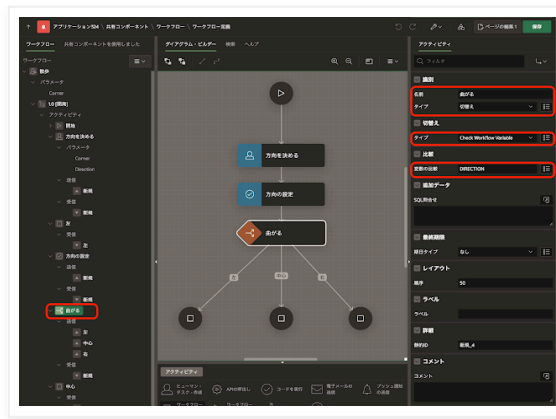
```
declare
    l_direction varchar2(40);
begin
    l_direction := apex_approval.get_task_parameter_value(
        p_task_id => :TASK_ID
        ,p_param_static_id => 'DIRECTION'
    );
    :DIRECTION := l_direction;
end;
```

set-direction.sql hosted with ❤ by GitHub

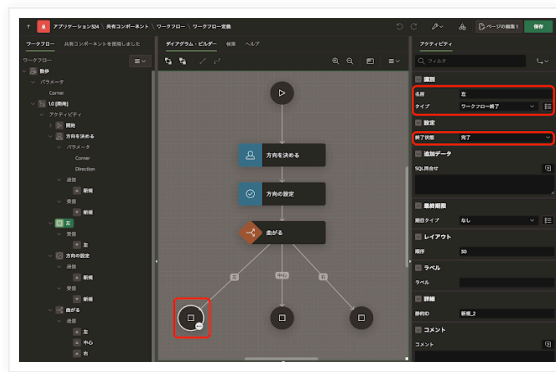
[view raw](#)



続く切替えのアクティビティ曲がるでは、切替えのタイプにCheck Workflow Variableを選択し、タスク・パラメータのDirectionから値を代入したワークフロー・バージョン変数のDIRECTIONを、比較の変数の比較に選択します。

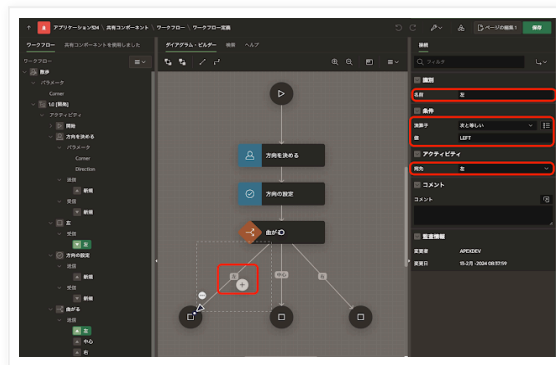


ワークフロー終了として左、中心、右を作成します。設定の終了状態は完了とします。



切替え曲がるとそれぞれの終了を接続します。

接続の名前はそれぞれ、左、中心、右、条件の演算子に次と等しいを選択し、値にLEFT、CENTER、RIGHTを設定します。アクティビティの宛先は左、中心、右になります。



以上でワークフローは完成です。

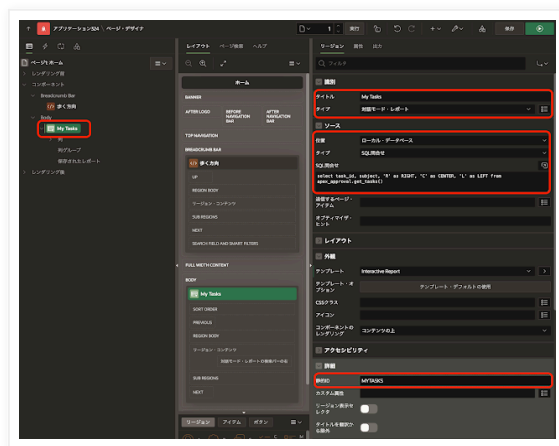
ホーム・ページに対話モード・レポートを作成し、方向を決める処理を実装します。

識別のタイトルはMy Tasks、タイプは対話モード・レポートです。

ソースのタイプにSQL問合せを選択し、SQL問合せとして以下を記述します。

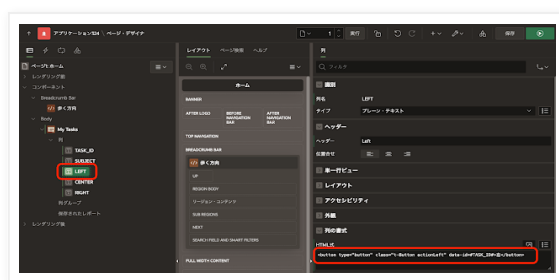
```
select task_id, subject, 'L' as LEFT, 'C' as CENTER, 'R' as RIGHT from apex_approval.get_tasks()
```

列RIGHT、CENTER、LEFTに左に曲がる、中心へ進む、右へ曲がる処理を実装します。これらの処理は動的アクションで実装するため、対話モード・レポートの静的IDとしてMYTASKSを設定します。



列LEFTの列の書式のHTML式に以下を記述し、列にボタンを表示します。CSSクラスのactionLeftはボタンの見かけの修飾のためではなく、クリックのイベントを認識させるために使用します。

<button type="button" class="t-Button actionLeft" data-id=#TASK_ID#>左</button>



列CENTERのHTML式は以下です。

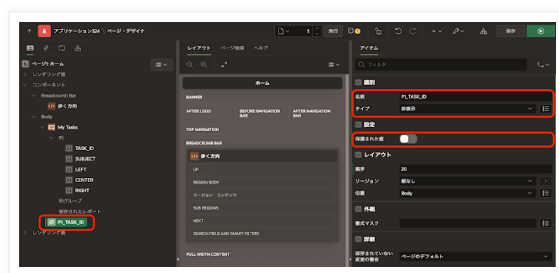
<button type="button" class="t-Button actionCenter" data-id=#TASK_ID#>中心</button>

列RIGHTのHTML式は以下です。

<button type="button" class="t-Button actionRight" data-id=#TASK_ID#>右</button>

カスタム属性のdata-idとして受け取ったタスクIDを、サーバー側の処理に渡すために使用するページ・アイテムP1_TASK_IDを作成します。

タイプは非表示、JavaScriptのコードより値を設定するため、設定の保護された値はオフにします。

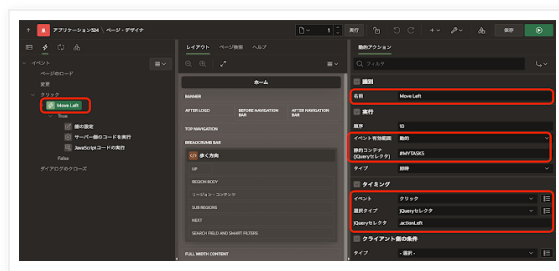


動的アクション**Move Left**を作成します。

実行の**イベント有効範囲**に**動的**を選択し、**静的コンテナ(jQueryセクタ)**として対話モード・レポートの**静的ID**を指定します。idを指定するjQueryセクタなので#を付加し、**#MYTASKS**になります。

タイミングは**クリック**、**選択タイプ**に**jQueryセクタ**を選択し、**jQueryセクタ**としてクラスを指定するため先頭に.(ピリオド)を付加し、**.actionLeft**を設定します。

以上で列**LEFT**のボタンを**クリック**したときに、この動的アクション**Move Left**が呼び出されます。



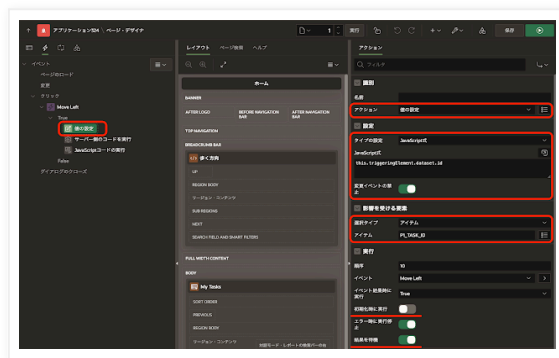
最初に実行する**TRUEアクション**として、**値の設定**を実行します。

設定の**タイプの設定**として**JavaScript式**を選択し、**JavaScript式**として以下を記述します。**変更イベントの禁止**は**オン**にします。カスタム属性**data-id**の値を取り出しています。

`this.triggeringElement.dataset.id`

影響を受ける要素の選択タイプに**アイテム**、**アイテム**として**P1_TASK_ID**を選択します。上記の**data-id**の値がページ・アイテム**P1_TASK_ID**に設定されます。

実行の**初期化時に実行**は**オフ**、**結果を待機**は**オン**にします。



続く**TRUEアクション**で**サーバー側のコード**を実行します。実行するコードは以下になります。タスク・パラメータの**DIRECTION**に値として**LEFT**を設定し、**タスクを完了**しています。

```
begin
  apex_approval.set_task_parameter_values(
    p_task_id      => :P1_TASK_ID
    ,p_parameters => apex_approval.t_task_parameters(
      1 => apex_approval.t_task_parameter(
        static_id => 'DIRECTION'
```



```

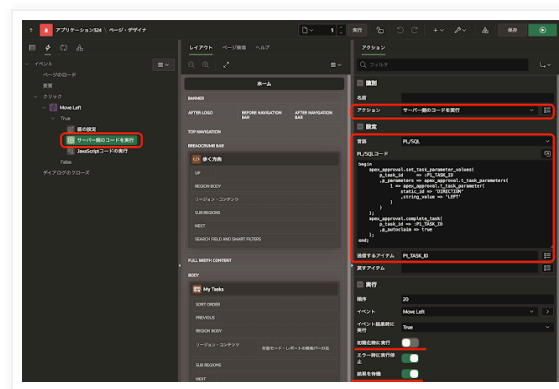
        ,string_value => 'LEFT'
    )
)
);
apex_approval.complete_task(
    p_task_id => :P1_TASK_ID
    ,p_autoclaim => true
);
end;

```

direction-left.sql hosted with ❤ by GitHub

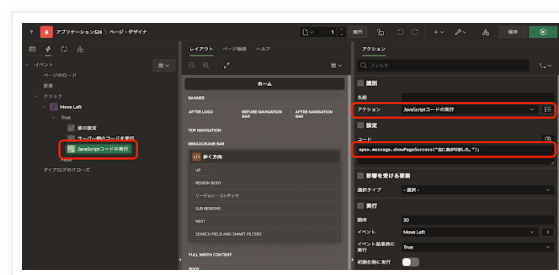
[view raw](#)

送信するアイテムとしてP1_TASK_IDを指定します。

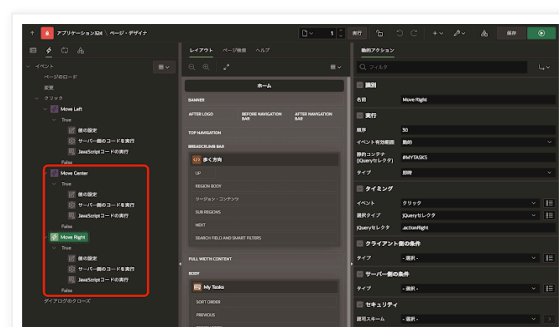


最後のTRUEアクションとしてJavaScriptコードの実行を選択し、設定のコードに以下を記述して、画面にメッセージを表示します。

```
apex.message.showPageSuccess("左に曲がりました。");
```

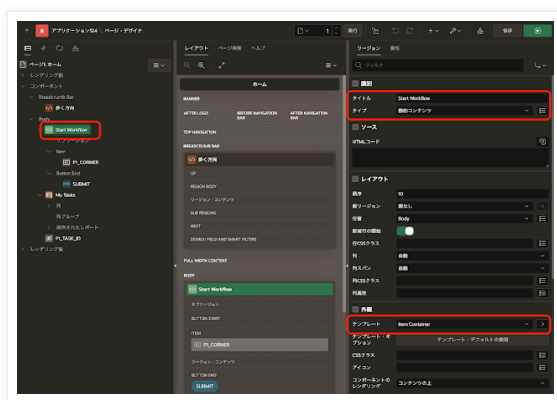


作成した動的アクションMove Leftをコピー（重複）し、動的アクションMove CenterとMove Rightを作成します。コピーした動的アクションのLeft（LEFT）となっている部分をそれぞれCenter（CENTER）やRight（RIGHT）に置き換えます。



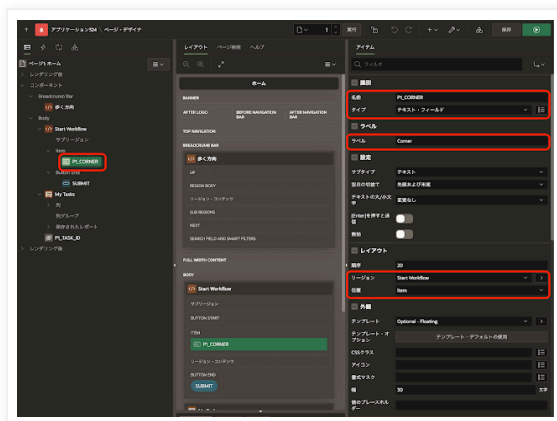
ワークフローを開始する機能を加えます。

静的コンテンツのリージョン**Start Workflow**を作成します。外観のテンプレートに**Item Container**を選択します。



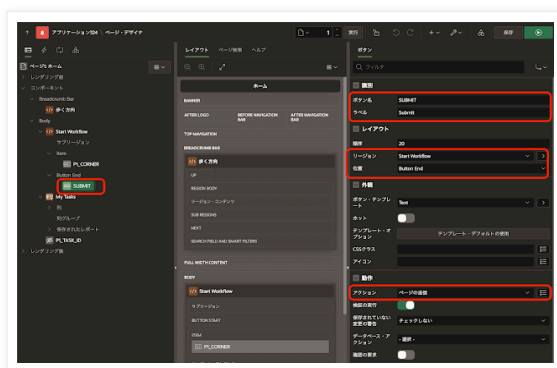
曲がり角の名前（タスクの件名に含める文字列）を設定するページ・アイテム**P1_CORNER**を作成します。**タイプ**は**テキスト・フィールド**とします。

レイアウトのリージョンに**Item Container**の**Start Workflow**を選択します。**位置**は**Item**になります。



ボタン**SUBMIT**を作成します。レイアウトのリージョンは**Start Workflow**、位置は**Button End**を選択します。

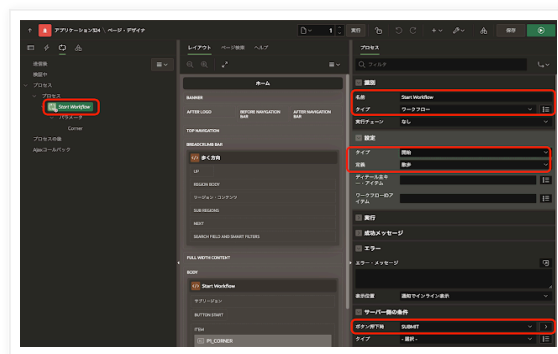
動作の**アクション**はデフォルトの**ページの送信**です。



ボタン**SUBMIT**をクリックしたときに実行されるプロセス**Start Workflow**を作成します。

設定の**タイプ**は**開始**、**定義**としてワークフロー**散歩**を選択します。

サーバー側の条件のボタン押下時に**SUBMIT**を指定します。



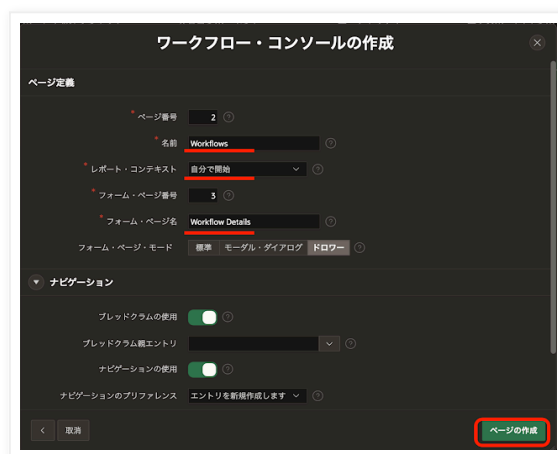
以上でアプリケーションは完成ですが、ワークフローの進捗を確認するためにワークフロー・コンソールのページを作成します。

ページの作成を実行し、**ワークフロー・コンソール**を選択します。



レポート・ページの名前は**Workflows**、レポート・コンテキストとして**自分で開始**を選択します。
フォーム・ページ名は**Workflow Details**とします。

ページの作成をクリックします。



以上でアプリケーションは完成です。アプリケーションを実行すると、記事の最初のGIF動画のように動作します。

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/sample-workflow-walk-direction.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: 22:45

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.
