

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。

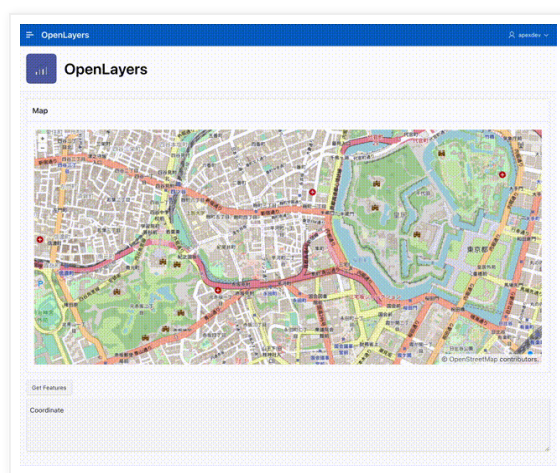
2023年2月13日月曜日

OpenLayersを使って図形を書く

折角なので[以前の記事](#)に続き、OpenLayersの部分も試してみました。

Drawing objects on maps in Oracle APEX - MapLibre vs OpenLayers

<https://pretius.com/blog/maplibre-vs-openlayers/>

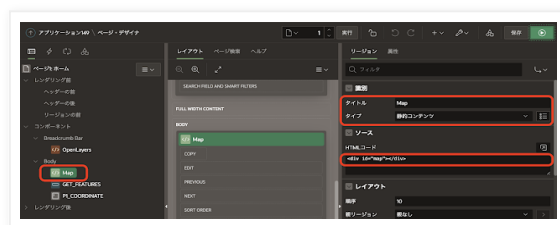


空のアプリケーションを作成したところから始めます。

静的コンテンツのリージョンを作成します。識別のタイトルはMapとします。

ソースのHTMLコードは以下です。

```
<div id="map"></div>
```



ページ・プロパティのJavaScriptとCSSを設定します。

JavaScriptのファイルURLは以下になります。OpenLayersのサイトより、最新のバージョンを確認しましょう。

<https://cdn.jsdelivr.net/npm/ol@v7.2.2/dist/ol.js>

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。

```
/*
 * This code is based on the article written by Lech Cieřlik, Pretius.
 * Drawing objects on maps in Oracle APEX – MapLibre vs OpenLayers
 * https://pretius.com/blog/maplibre-vs-openlayers/
 */

//Create Raster Source
const sourceRaster = new ol.source.OSM();
//Create new layer – Open Street Maps
const raster = new ol.layer.Tile({
  source: sourceRaster
});
//Create Vector Source
const source = new ol.source.Vector({wrapX: false});
//Create new layer – Vector Source
const vector = new ol.layer.Vector ({
  source: source,
});

/*
 * マップに書き込まれた図形のジオメトリを取得する。
 */
const GET_FEATURES = {
  name: "get-features",
  action: (event, element, args) => {
    let geojson = new ol.format.GeoJSON().writeFeatures(vector.getSource().getFeatures(), {
      dataProjection: 'EPSG:4326',
      featureProjection: 'EPSG:3857'
    });
    apex.items.P1_COORDINATE.setValue(geojson);
  }
};

/*
 * ページの準備ができてから実行する。
 */
apex.jQuery(window).on('theme42ready', () => {
  // apex.actionsにアクションを登録する。
  apex.actions.add([GET_FEATURES]);

  //Create map
  const map = new ol.Map({
    //set layer(s)
    layers: [raster, vector],
```

```

//set container
target: "map",
//set view
view: new ol.View({
  center: ol.proj.fromLonLat([12.48, 41.85]),
  zoom: 8
})
});

//Add Interaction (Draw object functionality)
var draw = new ol.interaction.Draw({
  source: source,
  type: "Polygon", //you can use 'Point' or 'LineString'
});
map.addInteraction(draw);
});

```

openlayers-init.js hosted with ❤ by GitHub

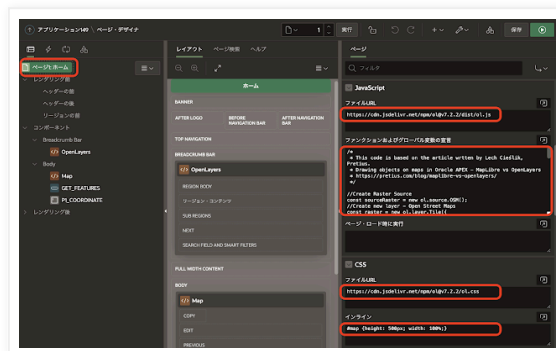
[view raw](#)

CSSのファイルURLに以下を記述します。

<https://cdn.jsdelivr.net/npm/ol@v7.2.2/ol.css>

CSSのインラインに以下を記述します。

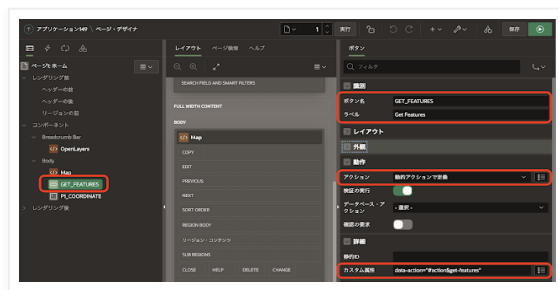
`#map {height: 500px; width: 100%;}`



マップに記述した図形のジオメトリを取得するボタンを作成します。

ボタン名はGET_FEATURES、ラベルはGet Features、動作のアクションとして動的アクションで定義を選択し、詳細のカスタム属性として以下を記述します。

`data-action="#action$get-features"`



ジオメトリを保持するページ・アイテムを作成します。

識別の名前はP1_COORDINATE、タイプとしてテキスト領域を選択します。



以上で完成です。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/openlayers-sample.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: 18:20

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.