Salesforce Integration Mini Project - 1

Weather Component

learnsalesforce@madhu

Weather API

https://openweathermap.org/api

https://api.openweathermap.org/data/2.5/weather?q={CityName}&appid={APIKey}

Output: {hyderabad}

{"coord":{"lon":78.4744,"lat":17.3753},"weather":[{"id":804,"main":"Clouds","description":"
overcast
clouds","icon":"04d"}],"base":"stations","main":{"temp":304.38,"feels_like":303.03,"temp_
min":304.38,"temp_max":305.88,"pressure":1014,"humidity":29},"visibility":6000,"wind":{
"speed":3.09,"deg":30},"clouds":{"all":98},"dt":1675674055,"sys":{"type":1,"id":9214,"cou
ntry":"IN","sunrise":1675646208,"sunset":1675687409},"timezone":19800,"id":1269843,"
name":"Hyderabad","cod":200}

Final Output:

Current Temperature

31°C

Wherever you go, no matter what the weather, always bring your own sunshine. ...

View More Info

Coordinates	Weather
longitude: 78.4744 latitude: 17.3753	Id: 804 main: Clouds description: overcast clouds icon: 04d

APEX CLASS:

weatherController

```
public with sharing class weatherController {
   @AuraEnabled
   public static Map<string,Object> retriveWeather(){
    HttpRequest httpRequest= new HttpRequest();
    HttpResponse httpResponse=new HttpResponse();

httpRequest.setEndpoint('https://api.openweathermap.org/data/2.5
/weather?q={CityName}&appid={APIKey}');
    httpRequest.setMethod('GET');

    Map<string,Object> newsJsonData= new Map<string,Object>();
    String strResponse = null;
      try {
        Http http=new Http();
        httpResponse= http.send(httpRequest);

        if(httpResponse.getStatusCode() == 200){
            strResponse =  httpResponse.getBody();
        }else{
            throw new calloutException(httpResponse.getBody());
        }

    } catch (Exception e) {
        throw e;
    }
    if(!String.isBlank(strResponse)){
        newsJsonData = (Map<String,
Object>)JSON.deserializeUntyped(strResponse);
    }
    if(!newsJsonData.isEmpty()){
        return newsJsonData;
```

```
    } else {
        return null;
    }
}
}
```

WeatherApex.cls

```
public class WeatherApex {


        public class Weather {
                public Integer id;
                public String main;
                public String description;
                public String icon;
        }

        public class Coord {
                public Double lon;
                public Double lat;
        }

        public class Wind {
                public Double speed;
                public Integer deg;
        }

        public class Clouds {
                public Integer all;
        }

        public Coord coord;
        public List<Weather> weather;
        public String base;
```

```
        public Main main;
        public Integer visibility;
        public Wind wind;
        public Clouds clouds;
        public Integer dt;
        public Sys sys;
        public Integer timezone;
        public Integer id;
        public String name;
        public Integer cod;

        public class Sys {
                public Integer type;
                public Integer id;
                public String country;
                public Integer sunrise;
                public Integer sunset;
        }

        public class Main {
                public Double temp;
                public Double feels_like;
                public Double temp_min;
                public Double temp_max;
                public Integer pressure;
                public Integer humidity;
        }


        public static WeatherApex parse(String json) {
                return (WeatherApex)
System.JSON.deserialize(json, WeatherApex.class);
        }
}
```

WeatherData.html

```
<template>
    <lightning-card  style="background-color: antiquewhite;" >
        <h1 class="titleStyle" align="center">Current
Temperature</h1>


                <img class="imegeStyle" style="float: left"
src="https://i.pinimg.com/originals/6f/8f/0a/6f8f0a1379f61a433b9
0c34e95789927.jpg" alt="award" height="200" width="200" >
                <h1 class="tempStyle"  > {celsius}°C</h1>




        <div title="Hyderabad">
        <h1 style="font-size: 30px;">Wherever you go, no matter
what the weather, always bring your own sunshine. ...</h1>
        </div>
        <div class="infoStyle">



        </div>
        <lightning-button variant="base" label="View More Info"
title="Looks like a link" onclick={handleClick}
class="slds-m-left_x-small"></lightning-button>

        <template if:true={openFullInfo}>
          <div class="slds-m-left_x-small">
          <div class="slds-grid slds-grid_pull-padded-medium">
```

```
                    <div class="slds-col
slds-p-horizontal_medium">
                      <span>

                      </span>
                    </div>
                    <div class="slds-col
slds-p-horizontal_medium">
                      <span>

                      </span>
                    </div>
                  </div>


                  <div class="slds-grid slds-gutters">
                    <div class="slds-box slds-box_x-small">


                    <div class="slds-col">
                      <span>
                              <div title="Coordinates">
                                        <h1 style="font-size:
20px;">Coordinates</h1>
                                        <h1 style="font-size:
15px;">longitude: {lon} </h1>
                                        <h1 style="font-size:
15px;">latitude: {lat} </h1>

                                    </div>
                      </span>
                    </div>
                    </div>
                    <div class="slds-box slds-box_x-small">
                    <div class="slds-col">
```

```
                    <span>
                            <div title="Weather">
                                    <h1 style="font-size:
20px;">Weather</h1>
                                    <h1 style="font-size:
15px;">Id: {id} </h1>
                                    <h1 style="font-size:
15px;">main: {main} </h1>
                                    <h1 style="font-size:
15px;">description: {description} </h1>
                                    <h1 style="font-size:
15px;">icon: {icon} </h1>
                            </div>
                    </span>
                  </div>
                  </div>
                  <div class="slds-col">
                    <span>
                            <img
src="https://images.unsplash.com/photo-1551161242-b5af797b7233?i
xlib=rb-4.0.3&ixid=MnwxMjA3fDB8MHxzZWFyY2h8MXx8aHlkZXJhYmFkfGVuf
DB8fDB8fA%3D%3D&w=1000&q=80" alt="Snow"  width="300px"
height="250px" >

                    </span>
                  </div>
                </div>


              <div class="container" title="Place">

                  <div class="centered"></div>
          </div>

        </div>
```

```
        </template>


        <!-- <template for:each={formateResponse}
for:item="fort">

        {fort}

        </template> -->
    </lightning-card></template>
```

weatherData.js

```
import { LightningElement , track} from 'lwc';
import weatherMap from
"@salesforce/apex/weatherController.retriveWeather";
export default class WeatherData extends LightningElement {

    @track results=[];
    formateResponse =[];
    name ;
    temp ;
    lon;
    lat;
     celsius ;
     id ;
     main;
     description;
     icon;

    openFullInfo = false;
    connectedCallback()
    {
        this.fetchWeather();
```

```
    }
    fetchWeather(){
        weatherMap().then(response=>{
            //console.log(response);
            console.log("JSON.stringify===> "+
JSON.stringify(response));
            //this.storethedata=response;



            this.name=response.name;
            this
            for(const[key,values] of Object.entries(response)){
               // console.log(key);
               // console.log(values);
                if(key==='main'){
                  /*  for(const[key,value] of
Object.entries(values)){
                            console.log(key,value);
                } */
                //console.log(values.temp);

                this.temp=values.temp;
                this.celsius = Math.round(this.temp -273.15);
                console.log(temp);

              }
              else if(key==='weather'){
                for(const[key,value] of Object.entries(values)){
                    //console.log(key,value);
                    console.log(value.id);
                //console.log(values.id);
                this.id=value.id;
                this.main=value.main;
                this.description=value.description;
                this.icon=value.icon;

              }
```

```
            }
              else if(key==='coord'){
                this.lon=values.lon;
                this.lat=values.lat;

                    console.log(values.lon);
            }




            }
            //this.temp=response.temp;
            console.log(this.name);
            //console.log(this.temp);

             /* console.log('Hi im from connected
callback',this.formateResponse);
            for(let i in this.formateResponse){
                console.log('i am back',this.formateResponse[i])
                this.formate = this.formateResponse[i]
            } */


        }).catch(error=>{
            console.error(error);
        })
           // console.log('storethe data'+storethedata);
    }
    /* formateResponse(res){
        this.results=res.map(()=>{
            let animals= animals;
            console.log(resuts);
            return { animals:animals}
        })

    } */
```

```
    handleClick(){
        console.log('clicked');
        this.openFullInfo=!this.openFullInfo;
    }
}
```

weatherData.js-meta.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<LightningComponentBundle
xmlns="http://soap.sforce.com/2006/04/metadata">
    <apiVersion>55.0</apiVersion>
    <isExposed>true</isExposed>
    <targets>
        <target>lightning__RecordPage</target>
        <target>lightning__AppPage</target>
        <target>lightning__HomePage</target>
    </targets>
</LightningComponentBundle>
```

weatherData.css

```
.titleStyle{

    font-size: 30px;

    font-weight: bold;
}
.infoStyle{


    color: black;
```

```
    font-size: 20px;
}


.imegeStyle{
    margin: 5px;
}
.tempStyle{
    top: 10px;
    left: 1000px;
    color: red;
    font-size: 100px;
    font-style: italic;
}
.container{

  text-align: center;
  color: black;




}
.centered {

    position: absolute;
    top: 50%;
    left: 50%;
```

```
    transform: translate(-50%, -50%);
    font-size: 100px;
    font-style: oblique;


}
```

THANK YOU