Maryam Soleimani

In this program we will get an API key from the weather forecast website and write it down in the code in the appropriate place as a string. So it will make a request and gets some information as a Jason string. Also we already have used Gradle to use and import Jason libraries. Then here we use Jason Object to convert those Jason strings to readable and usable datum. Here there are explanations about every functions(methods) that were used to convert that datum as long as the main function:

getTemperature:

Here first we declare a double variable called temperature and assign it to 0.0 as an initial value. Then we use JASONObject. Jasonobject is a class that contains keys and their values. In here we get the "weatherJason" string and inside that there are keys and values. Then here "current" is a key and it has a number of values for example for the key "current" we have "wind_mph" ,"wind_kph" , "wind_dir" , "cloud" , … values. And each of these values are keys for some other values:

{ "wind_mph" : 8.1 , "wind_kph" : 13.0 , "wind_dir " : "NNE"}

Here 8.1 , 13.0 and „NNE" are the values for the "wind_mph" ,"wind_kph" and "wind_dir" keys.

Keys must be strings, and values must be a valid JSON data type. For example string, number , object , array , Boolean and null.

Here we declare a Jason object called "obj" and its key is "weatherJason" (in the main function "weatherJason" is "weatherData(City)" which is assigned to string variable "Response")

Then again we declare a new Jason object called "time". Here we access to the values for the key "current" (JASONObject time = obj.getJASONObject("current");)). Then in the next line we want to access to the value of "temp_c" so "temp_c" is a key but it is also a value for "current" which we wanted to access to. And we assign it to our double variable "temperature". (temperature = time.getDouble("temp_c").

And at the end it returns the value of "temperature".

Response Body


```
{
    "location": {
        "name": "London",
        "region": "City of London, Greater London",
        "country": "United Kingdom",
        "lat": 51.52,
        "lon": -0.11,
        "tz_id": "Europe/London",
        "localtime_epoch": 1677677808,
        "localtime": "2023-03-01 13:36"
    },
    "current": {
        "last_updated_epoch": 1677677400,
        "last_updated": "2023-03-01 13:30",
        "temp_c": 7.0,
        "temp_f": 44.6,
```

```
        "is_day": 1,
        "condition": {
            "text": "Light rain",
            "icon": "//cdn.weatherapi.com/weather/64x64/day/296.png",
            "code": 1183
        },
        "wind_mph": 8.1,
        "wind_kph": 13.0,
        "wind_degree": 30,
        "wind_dir": "NNE",
        "pressure_mb": 1028.0,
        "pressure_in": 30.36,
        "precip_mm": 0.0,
        "precip_in": 0.0,
        "humidity": 70,
        "cloud": 100,
        "feelslike_c": 4.4,
        "feelslike_f": 39.9,
        "vis_km": 10.0,
        "vis_miles": 6.0,
        "uv": 2.0,
        "gust_mph": 10.7,
        "gust_kph": 17.3
    }
}
```


Above you can see the keys and values that are available in the weather forecast website.

getHumidity , WindSpeed and windDirection functions work the same way as explained.

Main function :

Here we go through a "while (true)" loop then we get the name of a city as an input.

This while loop will get inputs and shows the wanted outputs until the user enters the character "o" so the program will stop.

Then we call the "getWeatherData" function and give the "City" variable as a parameter to it and assign it to a string variable called "Response".

And then to print our outputs we need to call the getTemperature, getHumidity , WindSpeed and windDirection functions and give "Response" as a parameter to them.

But still there is a case that the user enters an invalid name for the city. To handle that, in "get WeatherData" function in the following part, it will return "null" if the city is invalid:

```
Catch ( Exception e){
 e.printStackTrace();
  return null;}
```

So in this case if "Response " is equal to "null" it means that the city name is invalid so the program will print that the city is invalid and asks the user to try again and enters a valid name for city.