EXC1081

GDG 2 Credit Course

Project

Weather Application on Android Studio

Using Accuweather API

16BCE0752

R.Anuraag

Index

Abstract

The aim of the project is to make an application on Android Studio which displays the forecast of the day on one page and the forecast for upcoming days on the next.

This can be done with the use of API provided by Accuweather.

Android Studio is a simple software which uses Java and Xml to make an application for Android complete with its layouts.

Introduction

The project can be divided into three components

1. **Android Studio**
    a. **Java**
    b. **Xml**
2. **The Accuweather API**
3. **UI of the App and the objective**

API (Application Program Interface)

These can be defined a set of functions and procedures that allow the creation of applications which access the features or data of an operating system, application, or other service

UI (User Interface)

The user interface, in the industrial design field of human–computer interaction, is the space where interactions between humans and machines

Android Studio

Android Studio is the official integrated development environment for Google's Android operating system, built on JetBrains' IntelliJ IDEA software and designed specifically for Android development

We need to constantly update it, so as to be in touch with the new bug fixes and changes

Methodology

Code

The code is divided into several java and xml files.

First the xml files

1. activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:padding="22dp"
```

```
    android:background="@drawable/yellow"
    tools:context="com.example.hp.weather.MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="192dp"
        android:layout_height="77dp"
        android:text="Weekly Forecast"
        android:textColor="@android:color/holo_red_light"
        android:textSize="24sp"
        android:textStyle="bold" />

    <ListView
        android:id="@+id/idListView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">

    </ListView>

</LinearLayout>
```

2. list_item.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/Date"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="18-03-18"
        android:textColor="@android:color/holo_orange_dark"
        android:textSize="18sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/tvTemperature"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Temperature"
        android:textColor="@android:color/holo_red_dark"
        android:textSize="18sp"
        android:textStyle="bold" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <TextView
            android:layout_width="91dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="High"
            android:textColor="@android:color/holo_blue_dark"
            android:textSize="18sp"
            android:textStyle="normal" />


        <TextView
            android:id="@+id/tvHighTemperature"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="32"
```

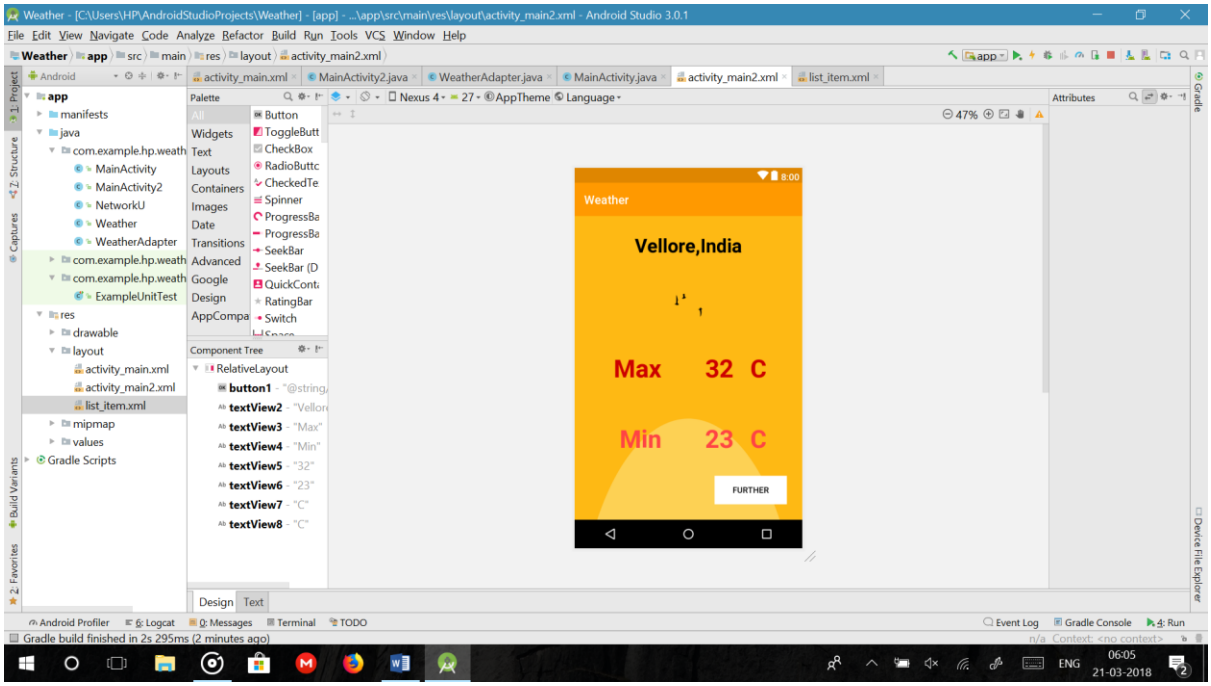
```
            android:textColor="@android:color/holo_green_dark"
            android:textSize="18sp"
            android:textStyle="normal" />

    </LinearLayout>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">

        <TextView
            android:layout_width="94dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="Low"
            android:textColor="@android:color/holo_blue_dark"
            android:textSize="18sp"
            android:textStyle="normal" />


        <TextView
            android:id="@+id/tvLowTemperature"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="23"
            android:textColor="@android:color/holo_green_dark"
            android:textSize="18sp"
            android:textStyle="normal" />

    </LinearLayout>


</LinearLayout>
```

3. activity_main2.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/back"
    tools:context="com.example.hp.weather.MainActivity2">

    <Button
        android:id="@+id/button1"
        android:layout_width="122dp"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentEnd="true"
        android:layout_marginBottom="26dp"
        android:layout_marginEnd="26dp"
        android:background="@android:color/white"
        android:text="@string/further"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.924"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.982" />
```

```
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="27dp"
    android:text="Vellore,India"
    android:textColor="@android:color/background_dark"
    android:textSize="32dp"
    android:textStyle="bold"
    tools:layout_editor_absoluteX="128dp"
    tools:layout_editor_absoluteY="32dp" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentStart="true"
    android:layout_centerVertical="true"
    android:layout_marginStart="66dp"
    android:text="Max"
    android:textColor="@android:color/holo_red_dark"
    android:textSize="42dp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/button1"
    android:layout_alignEnd="@+id/textView3"
    android:layout_marginBottom="35dp"
    android:text="Min"
    android:textColor="@android:color/holo_red_light"
    android:textSize="42dp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/textView3"
    android:layout_alignBottom="@+id/textView3"
    android:layout_alignEnd="@+id/button1"
    android:layout_marginEnd="90dp"
    android:text="32"
    android:textColor="@android:color/holo_red_dark"
    android:textSize="42dp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/textView4"
    android:layout_alignBottom="@+id/textView4"
    android:layout_alignStart="@+id/textView5"
    android:text="23"
    android:textColor="@android:color/holo_red_light"
    android:textSize="42dp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/textView6"
```

```
            android:layout_alignBottom="@+id/textView6"
            android:layout_alignStart="@+id/textView8"
            android:text="C"
            android:textColor="@android:color/holo_red_light"
            android:textSize="42dp"
            android:textStyle="bold" />

        <TextView
            android:id="@+id/textView8"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignBaseline="@+id/textView5"
            android:layout_alignBottom="@+id/textView5"
            android:layout_marginStart="14dp"
            android:layout_toEndOf="@+id/textView2"
            android:text="C"
            android:textColor="@android:color/holo_red_dark"
            android:textSize="42dp"
            android:textStyle="bold" />
    </RelativeLayout>
```



Now the java files,

Front Page

MainActivity2.java

```
package com.example.hp.weather;

import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.Button;

import static com.example.hp.weather.R.id.tvLowTemperature;


public class MainActivity2 extends AppCompatActivity {
    Button bt1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
        bt1 =(Button)findViewById(R.id.button1);
        bt1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openActivity2();
            }
        });
    }
    public void openActivity2(){
        Intent intent=new Intent(this,MainActivity.class);
        startActivity(intent);
```

MainActivity.java

```
package com.example.hp.weather;

import android.os.AsyncTask;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.widget.ListView;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.IOException;
import java.net.URL;
import java.util.ArrayList;
import java.util.Iterator;

public class MainActivity extends AppCompatActivity {

    private static final String TAG = MainActivity.class.getSimpleName();
    private ArrayList<Weather>weatherArrayList= new ArrayList<>();
    private ListView listView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);

        listView= findViewById(R.id.idListView);

        URL weatherUrl = NetworkU.buildUrlForWeather();
        new FetchWeatherDetails().execute(weatherUrl);
        Log.i(TAG, "onCreate: weatherUrl:"+ weatherUrl);
        }

        class FetchWeatherDetails extends AsyncTask<URL, Void, String> {

        @Override
        protected void onPreExecute() {
            super.onPreExecute();
        }
        @Override
        protected String doInBackground(URL...urls) {
            URL weatherUrl = urls[0];
            String weatherSearchResults = null;

            try {
                weatherSearchResults = NetworkU.getResponseFromHttpUrl(weatherUrl);
            } catch(IOException e){
                e.printStackTrace();
            }
            Log.i(TAG,"doInBackground: weatherSearchResults:"
+weatherSearchResults);
            return weatherSearchResults;
        }

        @Override
            protected void onPostExecute(String weatherSearchResults) {
            if(weatherSearchResults != null && !weatherSearchResults.equals("")){
                weatherArrayList = parseJSON(weatherSearchResults);
                //Just for Testing
                Iterator itr= weatherArrayList.iterator();
                while(itr.hasNext()) {
                    Weather weatherInIterator = (Weather) itr.next();
                    Log.i(TAG, "onPostExecute" + weatherInIterator.getDate() +
                            "Min:" + weatherInIterator.getMinTemp() +
                            "Max:" + weatherInIterator.getMaxTemp() +
                            "Link: " + weatherInIterator.getLink());

                }
        }
        super.onPostExecute(weatherSearchResults);

            }
        }

    private ArrayList<Weather> parseJSON(String weatherSearchResults){
     if(weatherArrayList !=null){
         weatherArrayList.clear();
    }

    if(weatherSearchResults !=null){
         try{

             JSONObject rootObject = new JSONObject(weatherSearchResults);
             JSONArray results =rootObject.getJSONArray("DailyForecasts");

             for(int i=0;i<results.length();i++){
                 Weather weather =new Weather();

                 JSONObject resultsObj = results.getJSONObject(i);

                 String date = resultsObj.getString("Date");
                 weather.setDate(date);
```

```
                Log.i(TAG,"parseJSON:date:"+ date);

                JSONObject temperatureObj =
resultsObj.getJSONObject("Temperature");
                String minTemperature =
temperatureObj.getJSONObject("Minimum").getString("Value");
                weather.setMinTemp(minTemperature);

                Log.i(TAG,"parseJSON:minTemperature:"+ minTemperature);

                String maxTemperature =
temperatureObj.getJSONObject("Maximum").getString("Value");
                weather.setMaxTemp(maxTemperature);

                Log.i(TAG,"parseJSON:maxTemperature:"+ maxTemperature);

                String link= resultsObj.getString("Link");
                weather.setLink(link);

                /*Log.i(TAG,"parseJSON:date:" + date +" "+
                        "Min:"+ minTemperature+" "+
                        "Max:"+maxTemperature+" "+
                        "Link:"+ link);*/

                weatherArrayList.add(weather);
            }
            if(weatherArrayList !=null){
                WeatherAdapter weatherAdapter = new WeatherAdapter(this,
weatherArrayList);
                listView.setAdapter(weatherAdapter);
            }
             return weatherArrayList;
         }catch (JSONException e){
             e.printStackTrace();
         }
    }
    return null;
}
}
```

WeatherAdapter.java

```
package com.example.hp.weather;

import android.content.Context;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.TextView;

import java.util.ArrayList;

import static com.example.hp.weather.R.id.tvLowTemperature;

/**
 * Created by HP on 18-03-2018.
 */

public class WeatherAdapter extends ArrayAdapter<Weather> {

    public WeatherAdapter(@NonNull Context context, ArrayList<Weather>
weatherArrayList) {
        super(context,0,weatherArrayList);
    }
```

```
    @NonNull
    @Override
    public View getView(int position, @Nullable View convertView, @NonNull
ViewGroup parent) {
        Weather weather = getItem(position);

        if(convertView == null){
            convertView =
LayoutInflater.from(getContext()).inflate(R.layout.list_item,parent,false);
        }

        TextView dateTextView=convertView.findViewById(R.id.Date);
        TextView minTextView = convertView.findViewById(tvLowTemperature);
        TextView maxTextView = convertView.findViewById(R.id.tvHighTemperature);

        dateTextView.setText(weather.getDate());
        minTextView.setText(weather.getMinTemp());
        maxTextView.setText(weather.getMaxTemp());

        return convertView;
    }
}
```

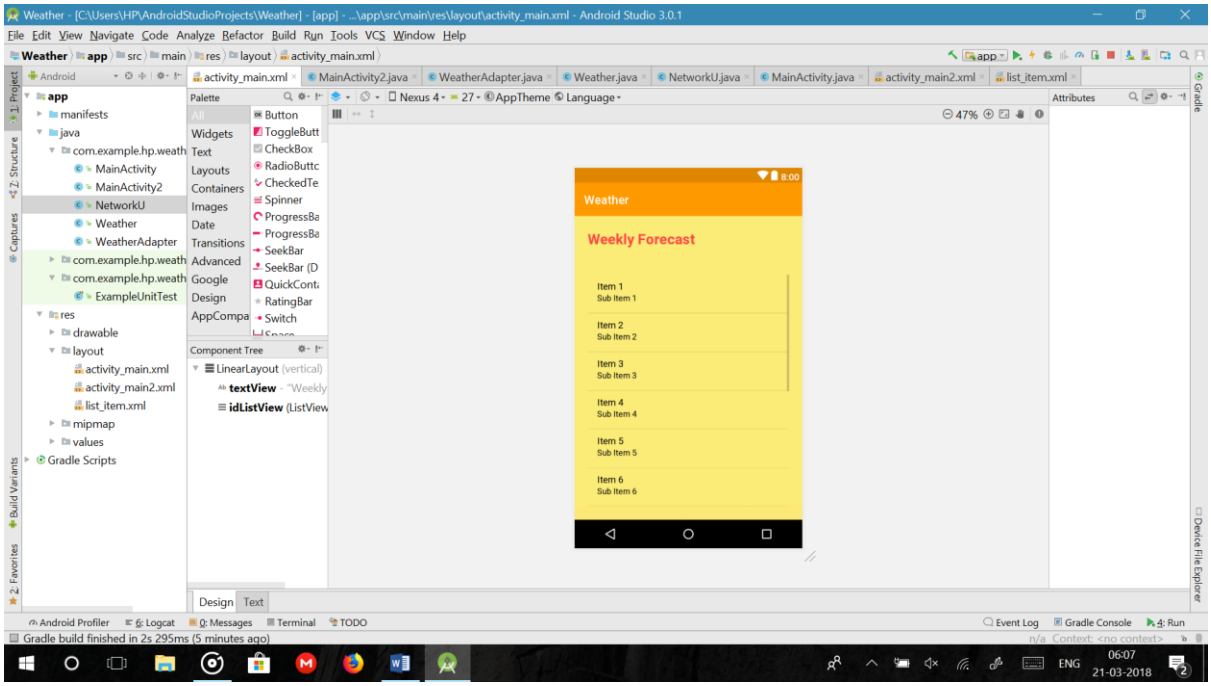
Weather.java

```
package com.example.hp.weather;

/**
 * Created by HP on 18-03-2018.
 */

public class Weather {
    String date;
    String minTemp;
    String maxTemp;
    String link;

    public String getDate() {
        return date;
    }

    public void setDate(String date) {
        this.date = date;
    }

    public String getMinTemp() {
        return minTemp;
    }

    public void setMinTemp(String minTemp) {
        this.minTemp = minTemp;
    }

    public String getMaxTemp() {
        return maxTemp;
    }

    public void setMaxTemp(String maxTemp) {
        this.maxTemp = maxTemp;
    }

    public String getLink() {
        return link;
    }

    public void setLink(String link) {
        this.link = link;
```

```
    }
}
```

Network.java

```
package com.example.hp.weather;

import android.net.Uri;
import android.util.Log;

import java.io.IOException;
import java.io.InputStream;
import java.net.HttpURLConnection;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.Scanner;

/**
 * Created by HP on 18-03-2018.
 */

public class NetworkU {
    private static final String TAG = "NetworkU";
    private final static String WEATHERDB_BASE_URL =
"http://dataservice.accuweather.com/forecasts/v1/daily/5day/190795";

    private final static String API_KEY = "aQMG1AeQjo788PzhW44ajsd9XV4g7whU";
    private final static String METRIC_VALUE="true";

    private final static String PARAM_API_KEY = "apikey";

    private final static String PARAM_METRIC="metric";

    public static URL buildUrlForWeather() {
        Uri builtUri = Uri.parse(WEATHERDB_BASE_URL).buildUpon()
                .appendQueryParameter(PARAM_API_KEY, API_KEY)
                .appendQueryParameter(PARAM_METRIC,METRIC_VALUE)
                .build();

        URL url = null;
        try {
            url = new URL(builtUri.toString());
        } catch (MalformedURLException e) {
            e.printStackTrace();
        }
        Log.i(TAG, "buildUrlForWeather: url: " + url);
        return url;
    }

    public static String getResponseFromHttpUrl(URL url) throws IOException {
        HttpURLConnection urlConnecton = (HttpURLConnection) url.openConnection();
        try {
            InputStream in = urlConnecton.getInputStream();

            Scanner scanner = new Scanner(in);
            scanner.useDelimiter("\\A");

            boolean hasInput = scanner.hasNext();
            if (hasInput) {
                return scanner.next();
            } else {
                return null;
            }
        } finally {
            urlConnecton.disconnect();
        }}
```



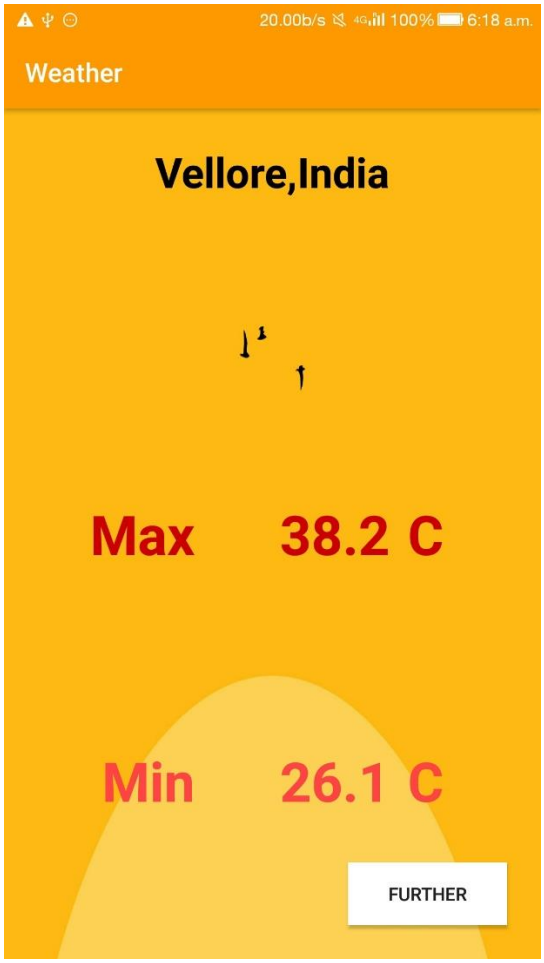
Basic Concept

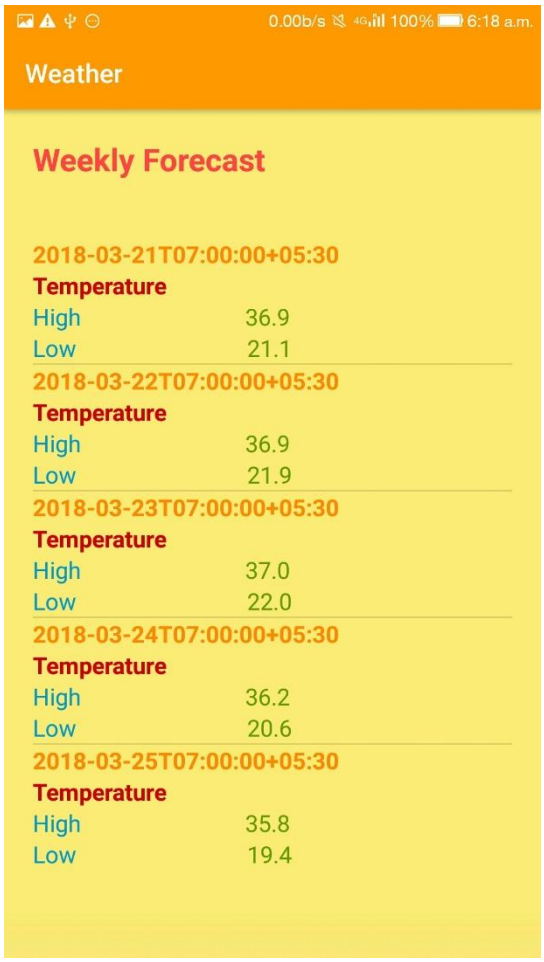
The Basic Concept is the use of API provided by Accuweather, taking the required data and information and storing them in separate variables for accessing later.

Step-by Step Execution

1. Getting or Accessing the API provided by Accuweather.
2. Formation of the Front page(actvit_main2) and the second page(activity_main).
3. Then creation of Weather,java file which consists of ‘Getter and Setter’ functions for easy array access.
4. Then creation of the Network.java file for HttpAccessRequest i.e. the setting up of connection with accuweather to access the weather data.
5. Then finally WeatherAdapter.java that is used to put the minTemp and maxTemp functions into the Text View.
6. Finally, use of MainActivity for compilation of all the above files.
7. Finally, Adding the first array of Weather to MainActivity2 for displaying the current day Forecast.

Result





Difficulties Faced

1. There was a lot of difficulty in bringing a day's forecast to the foreground, due to use of two different API calls.
2. Another issue was the 50 call limit put of Accuweather APIs

Conclusion

Hence, We can say that a weather app can be created with the help of certain simple compnents and some creativity.

This app can be later put on the App Store for public use and convenience.

Plus, the app occupies a lot less space , that being a plus.