

ICP – 10
Developing earthquake
information application for Android

ICP GROUP: 38

ICP - 10

Name: Anil Kumar Reddy Nandikonda

Email: anggp@umsystem.edu

ICP10 Repository: <https://github.com/UMKC-APL-WebMobileProgramming/ICP10-AnilkumarreddyNandikonda>

ICP10 source code link: <https://github.com/UMKC-APL-WebMobileProgramming/ICP10-AnilkumarreddyNandikonda/tree/main/EarthquakeApp>

My Partner :

Partner name: Abhinay Yadav

Partner Email: ayr6y@umsystem.edu

Partner Repository: <https://github.com/UMKC-APL-WebMobileProgramming/ICP10-YAbhinay>

Source Code Link: <https://github.com/UMKC-APL-WebMobileProgramming/ICP10-YAbhinay/tree/main/Source/EarthquakeApp/app/src>

ICP10 video:

<https://umsystem.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=7cd45e14-fae4-400b-8c50-adda007f0f97>

Lesson Overview: This lesson helps understand some crucial aspects of Android, such as fetching JSON data from APIs, parsing the JSON data, handling errors, using the Async Task Class, and some Java elements and usage of ListView.

Programming elements: RESTful Services, ListView, Adapter, Recycling, Multi-Threading, and Async Task

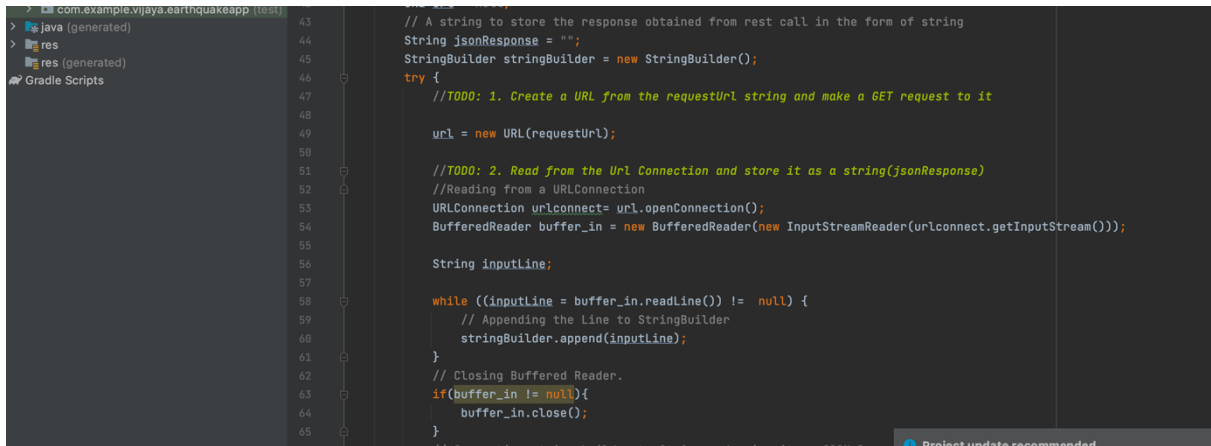
In Class Programming (ICP):

Earthquake Info App: Create a mobile application with the following requirements.

1. The main activity should be a list of earthquakes with information as shown below.
2. On clicking any item Earthquake, the user must be directed to the USGS web page, which contains further information about the selected earthquake.
3. Follow all the three TODO instructions in the QueryUtils.java file and two more TODO instructions in the EarthquakeActivity.java file and AndroidManifest.xml to complete the functionality

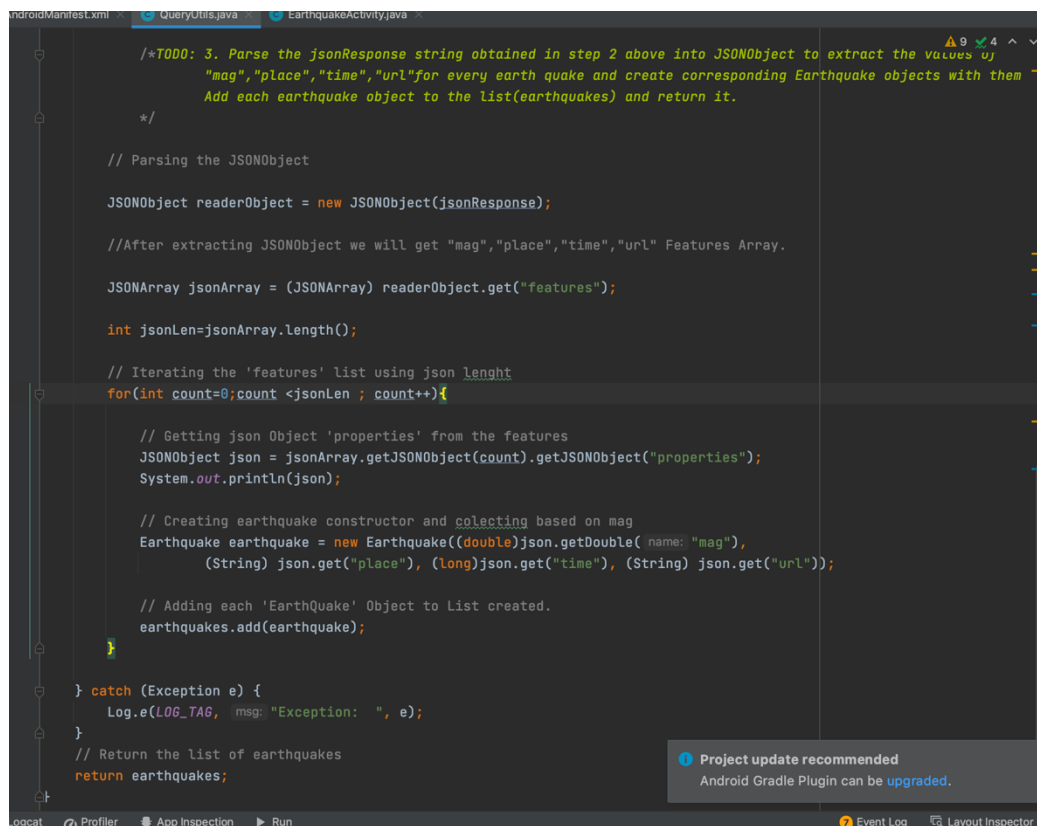
To get the ListView, the following code was added, which also finished all of the TODO's in the source code.

- URL is created, where we get the data
- We get JSON response is stored in string format.



```
43 // A string to store the response obtained from rest call in the form of string
44 String jsonResponse = "";
45 StringBuilder stringBuilder = new StringBuilder();
46 try {
47     //TODO: 1. Create a URL from the requestUrl string and make a GET request to it
48
49     url = new URL(requestUrl);
50
51     //TODO: 2. Read from the Url Connection and store it as a string(jsonResponse)
52     //Reading from a URLConnection
53     URLConnection urlConnect= url.openConnection();
54     BufferedReader buffer_in = new BufferedReader(new InputStreamReader(urlConnect.getInputStream()));
55
56     String inputLine;
57
58     while ((inputLine = buffer_in.readLine()) != null) {
59         // Appending the Line to StringBuilder
60         stringBuilder.append(inputLine);
61     }
62     // Closing Buffered Reader.
63     if(buffer_in != null){
64         buffer_in.close();
65     }
66 }
```

- In before step which we got JSON response, now it will be parse into JSON object. The JSON object will be in clearly data i.e Time, Date, Latitude, Longitude



```
/*TODO: 3. Parse the jsonResponse string obtained in step 2 above into JSONObject to extract the values of
"mag","place","time","url"for every earth quake and create corresponding Earthquake objects with them
Add each earthquake object to the list(earthquakes) and return it.
*/
// Parsing the JSONObject
JSONObject readerObject = new JSONObject(jsonResponse);
//After extracting JSONObject we will get "mag","place","time","url" Features Array.
JSONArray jsonArray = (JSONArray) readerObject.get("features");
int jsonLen=jsonArray.length();
// Iterating the 'features' list using json lenght
for(int count=0;count < jsonLen ; count++){
    // Getting json Object 'properties' from the features
    JSONObject json = jsonArray.getJSONObject(count).getJSONObject("properties");
    System.out.println(json);
    // Creating earthquake constructor and collecting based on mag
    Earthquake earthquake = new Earthquake((double)json.getDouble( name: "mag"),
        (String) json.get("place"), (Long)json.get("time"), (String) json.get("url"));
    // Adding each 'Earthquake' Object to List created.
    earthquakes.add(earthquake);
}
} catch (Exception e) {
    Log.e(LOG_TAG, "Exception: ", e);
}
// Return the list of earthquakes
return earthquakes;
```

- We can see different activities i.e. Earthquakes location countries, cities.

```

49 // to open a website with more information about the selected earthquake.
50 earthquakeListView.setOnItemClickListener((adapterView, view, position, l) -> {
51     // Convert the String URL into a URI object (to pass into the Intent constructor)
52     Uri earthquakeUri = Uri.parse(currentEarthquake.getUrl());
53
54     //TODO: 4. Create a new intent to view the earthquake URI.Send the intent to launch a new activity
55
56     Intent eqURIIntent = new Intent(Intent.ACTION_VIEW, earthquakeUri);
57     startActivity(eqURIIntent);
58 });
59
60 // Start the AsyncTask to fetch the earthquake data
61 EarthquakeAsyncTask task = new EarthquakeAsyncTask();
62 task.execute(USGS_REQUEST_URL);
63 }

```

- Giving permissions to user to use the services of the application.

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.example.vijaya.earthquakeapp">
4
5     <!--TODO 5: Add internet permission-->
6     <uses-permission android:name="android.permission.INTERNET" />
7
8     <application
9         android:allowBackup="true"
10        android:icon="@mipmap/ic_launcher"
11        android:label="EarthQuake App"
12        android:supportRtl="true"
13        android:theme="@style/AppTheme">
14        <activity android:name="com.example.vijaya.earthquakeapp.EarthquakeActivity">
15            <intent-filter>
16                <action android:name="android.intent.action.MAIN" />
17                <category android:name="android.intent.category.LAUNCHER" />
18            </intent-filter>
19        </activity>
20    </application>
21 </manifest>

```

OUTPUT :

- The activities are listed here, where we can access the data for each and every activity.

Android Emulator - Pixel_4_API_29:5554

12:51

EarthQuake App

4.6	OFF THE COAST OF Valparaiso, Chile	Nov 08, 2021 12:16 AM
5.0	96 KM ENE OF Luwuk, Indonesia	Nov 07, 2021 11:05 PM
4.8	75 KM SSW OF Vallenar, Chile	Nov 07, 2021 10:52 PM
4.8	23 KM ESE OF Santa Maria, Philippines	Nov 07, 2021 10:16 PM
4.5	119 KM E OF W?ngjing, India	Nov 07, 2021 8:18 PM
4.5	NEAR THE Pulau-Pulau Sula, Indonesia	Nov 07, 2021 1:43 PM
4.5	61 KM NE OF Misawa, Japan	Nov 07, 2021 1:22 PM
5.1	NEAR THE South Sandwich Islands region	Nov 07, 2021 12:50 PM
4.4	7 KM WNW OF Tsukuba, Japan	Nov 07, 2021 12:08 PM
4.7	67 KM NE OF Amahai, Indonesia	Nov 07, 2021 10:28 AM

- After selecting one activity we can observe the earthquake date, time latitude, longitude, URL's to view exact point of location.

