

\*\*\*\*\*XML and JSON Data\*\*\*\*\*

```
<?xml version="1.0" encoding="utf-8"?>
<records>
  <place>
    <City_Name>Mysore</City_Name>
    <Latitude>12.295</Latitude>
    <Longitude>76.639</Longitude>
    <Temperature>22</Temperature>
    <Humidity>90%</Humidity>
  </place>
  <place>
    <City_Name>Mandya</City_Name>
    <Latitude>12.298</Latitude>
    <Longitude>76.839</Longitude>
    <Temperature>23</Temperature>
    <Humidity>95%</Humidity>
  </place>
</records>
```

#### JSON DATA

```
[
  {
    "City_Name": "Mysore",
    "Latitude": "12.295",
    "Longitude": "76.639",
    "Temperature": "22",
    "Humidity": "90%"
  },
  {
    "City_Name": "Mysore",
    "Latitude": "12.298",
    "Longitude": "76.739",
    "Temperature": "23.0",
    "Humidity": "90%"
  }
]
```

\*\*\*\*\*START HERE\*\*\*\*\*

```
<?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="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="XML and JSON PARSER!"
        android:textAlignment="center"
        android:textColor="@color/black"
        android:textSize="31sp"
        android:textStyle="bold" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="XMLParser"
        android:text="Parse XML Data"
        android:textSize="18sp"
        android:textAlignment="center"/>

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:onClick="JSONParser"
        android:text="Parse JSON Data"
        android:textSize="18sp" />

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

        <TextView
            android:id="@+id/resxml"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Dummy XML Data"
            android:textAlignment="center"
            android:textColor="@color/black"
            android:textSize="15sp"
            android:paddingLeft="10sp"
            android:paddingRight="40sp"/>
```

```

<TextView
    android:id="@+id/resjson"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Dummy JSON Data"
    android:textAlignment="center"
    android:textColor="@color/black"
    android:textSize="15sp"
    android:paddingLeft="40sp"/>
</LinearLayout>

</LinearLayout>

```

## LAB 6: JAVA FILE

```

package com.example.lab6;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.TextView;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Node;
import org.w3c.dom.NodeList;
import org.xml.sax.SAXException;

import java.io.IOException;
import java.io.InputStream;

import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.parsers.ParserConfigurationException;

public class MainActivity extends AppCompatActivity {
    TextView resxml, resjson; //Create textView objects

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //Create Reference to both text views
        resxml = findViewById(R.id.resxml);
        resjson = findViewById(R.id.resjson);
    }
}

```

```

public void XMLParser(View view) {
    //Parse/Read the XML data using InputStream,
    // ..remember to surround with TRY & CATCH
    try {
        InputStream is = getAssets().open("city.xml");
        //Next Create a document builder factory...from its class
        DocumentBuilderFactory dbFactory =
DocumentBuilderFactory.newInstance();
        //Followed by DocumentBuilder...remember to add
ParserConfigurationException
        DocumentBuilder dBuilder = dbFactory.newDocumentBuilder();
        //Step 3: Create another object of type Document, add
SAXException
        Document doc = dBuilder.parse(is);
        //Create another Document Element
        Element element = doc.getDocumentElement(); //Reads the
element from XML and you must Normalize them
        element.normalize();
        //VIP: Specity the Root Note from which my data is to be taken
and populated
        //...in our case(xml), it is <place> under <records>
        //CREATE the NodeList
        NodeList nList = doc.getElementsByTagName("place");//This will
find two such "places" and assign it to nList
        resxml.setText("XML DATA");//Sets this in the TextView instead
of Manually assigning them as we did before
        //We need to LOOP through all the places and display the
records along with the tag name
        for(int i = 0; i<nList.getLength();i++)
        { //Same as i<2, i++
            Node node = nList.item(i);
            if (node.getNodeType() == Node.ELEMENT_NODE)
            {
                Element element1=(Element) node;
                //Display the data
                resxml.setText(resxml.getText()+"\n City
Name:"+getValue("City_Name", element1)+"\n");//This helps to append the
data
                //NOTE: cityname is the tagname and getValue() is user
define function
                resxml.setText(resxml.getText()+"\n
Latitude:"+getValue("Latitude", element1)+"\n");
                resxml.setText(resxml.getText()+"\n
Longitude:"+getValue("Longitude", element1)+"\n");
                resxml.setText(resxml.getText()+"\n
Temperature:"+getValue("Temperature", element1)+"\n");
                resxml.setText(resxml.getText()+"\n City
Name:"+getValue("Humidity", element1)+"\n");
                resxml.setText(resxml.getText()+"\n");
            }
        }
    }
}

```

```

    } catch (IOException | ParserConfigurationException | SAXException
e) {
        e.printStackTrace();
    }
}

private static String getValue(String tag, Element element) {
    NodeList nodeList =
element.getElementsByTagName(tag).item(0).getChildNodes();//initially if
theres's no childNode, 0 will be returned
    Node node = nodeList.item(0);
    return node.getNodeValue(); //returns the value associated with
Mysore City
}

public void JSONParser(View view) {
    String json;//Need this later for UTF-8 Encoding
    try {
        InputStream is = getAssets().open("city1.json");
        int size = is.available();//checks the size of the Json file
        byte[] buffer = new byte[size];//create a buffer to hold the
size of the array and finally read it
        is.read(buffer);
        is.close();//closes the input stream

        json=new String(buffer,"UTF-8");//UTF-8 is an encoding system
for Unicode. It can translate any Unicode character to a matching unique
binary string, and can also translate the binary string back to a Unicode
character. This is the meaning of "UTF", or "Unicode Transformation Format
        JSONArray jsonArray = new JSONArray(json);
        resjson.setText("JSON DATA");
        //Use a FOR-LOOP to iterate through the indices and returns
the JSON array objects
        for(int i=0; i<jsonArray.length();i++){
            JSONObject obj = jsonArray.getJSONObject(i);
            resjson.setText(resjson.getText()+"\n City Name:
"+obj.getString("City_Name")+"\n");//So, since JSON mimics a dictionary,
using the City_Name, we can get the value
            resjson.setText(resjson.getText()+"\n Longitude:
"+obj.getString("Longitude")+"\n");
            resjson.setText(resjson.getText()+"\n Latitude:
"+obj.getString("Latitude")+"\n");
            resjson.setText(resjson.getText()+"\n Temperature:
"+obj.getString("Temperature")+"\n");
            resjson.setText(resjson.getText()+"\n Humidity:
"+obj.getString("Humidity")+"\n");
            resjson.setText(resjson.getText()+"\n");
        }

    } catch (IOException | JSONException e) {
        e.printStackTrace();
    }
}

```

}  
}  
}