**Lab 29: JSON Parsing using Kotlin**

# **Introduction**

# JSON stands for JavaScript Object Notation.It is an independent data exchange format and is the best alternative for XML

Android provides four different classes to manipulate JSON data. These classes are.

* **JSONArray**
* **JSONObject**
* **JSONStringer**
* **JSONTokenizer.**

**Let’s get Started**

In this experiment we will develop an Android App to demonstrate the use of Android JSON Parsing.

**Step 1: Create a New Project in Android Studio as shown below**

Graphical user interface, text, application

Description automatically generated

**Step 2: Select Empty Activity as shown below**

Graphical user interface, application, shape

Description automatically generated

**Step 3: Update MainActivity.kt as per the code given below**

**package** com.example.demojson  
**import** android.os.Bundle  
**import** android.util.Log  
**import** android.widget.ListAdapter  
**import** android.widget.ListView  
**import** android.widget.SimpleAdapter  
**import** androidx.appcompat.app.AppCompatActivity  
**import** org.json.JSONException  
**import** org.json.JSONObject  
**import** java.util.\*  
  
**class** MainActivity : AppCompatActivity() {  
 **override fun** onCreate(savedInstanceState: Bundle?) {  
 **super**.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
 *// private string declare in the latter section of the program* **val** jsonStr = **listData  
 try** {  
  
 *// Create a userList string hashmap arraylist* **val** userList = ArrayList<HashMap<String, String?>>()  
  
 *// Declaring the listView from the layout file* **val** lv = findViewById<ListView>(R.id.*user\_list*)  
  
 *// Initializing the JSON object and extracting the information* **val** jObj = JSONObject(jsonStr)  
 **val** jsonArry = jObj.getJSONArray(**"users"**)  
 **for** (i **in** 0 *until* jsonArry.length()) {  
 **val** user = HashMap<String, String?>()  
 **val** obj = jsonArry.getJSONObject(i)  
 user[**"name"**] = obj.getString(**"name"**)  
 user[**"designation"**] = obj.getString(**"designation"**)  
 user[**"location"**] = obj.getString(**"location"**)  
 userList.add(user)  
 }  
  
 *// ListAdapter to broadcast the information to the list elements* **val** adapter: ListAdapter = SimpleAdapter(  
 **this**, userList, R.layout.*list\_row*,  
 *arrayOf*(**"name"**, **"designation"**, **"location"**), *intArrayOf*(  
 R.id.*name*,  
 R.id.*designation*, R.id.*location* )  
 )  
 lv.*adapter* = adapter  
 } **catch** (ex: JSONException) {  
 Log.e(**"JsonParser Example"**, **"unexpected JSON exception"**, ex)  
 }  
 }  
  
  
 *// JSON object in the form of input stream* **private val listData**: String  
 **get**() = (**"{ \"users\" :["** +  
 **"{\"name\":\"Ace\",\"designation\":\"Engineer\",\"location\":\"New York\"}"** +  
 **",{\"name\":\"Tom\",\"designation\":\"Director\",\"location\":\"Chicago\"}"** +  
 **",{\"name\":\"Tim\",\"designation\":\"Charted Accountant\",\"location\":\"Sunnyvale\"}] }"**)  
}

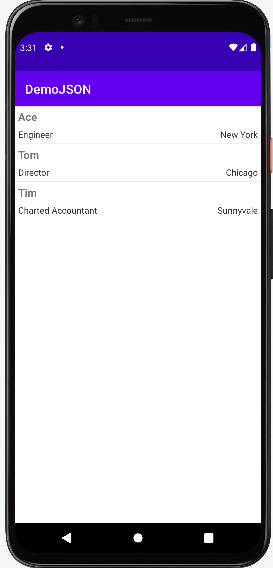
**Step 4: Update activity\_main.xml as per the code given below**

*<?***xml version="1.0" encoding="utf-8"***?>*<**LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:orientation="vertical"** >  
  
 *<!--This listView will display the list items-->* <**ListView  
 android:id="@+id/user\_list"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:dividerHeight="1dp"** />  
  
</**LinearLayout**>

**Step 6: Create list\_row.xml**

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal"  
 android:padding="5dip"**>  
  
 *<!--TextView to display the name-->* <**TextView  
 android:id="@+id/name"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textSize="17dp"  
 android:textStyle="bold"** />  
  
 *<!--TextView to display the designation-->* <**TextView  
 android:id="@+id/designation"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/name"  
 android:layout\_marginTop="7dp"  
 android:textColor="#343434"  
 android:textSize="14dp"** />  
  
 *<!--TextView to display the location-->* <**TextView  
 android:id="@+id/location"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignBaseline="@+id/designation"  
 android:layout\_alignBottom="@+id/designation"  
 android:layout\_alignParentRight="true"  
 android:textColor="#343434"  
 android:textSize="14dp"** />  
</**RelativeLayout**>

**Step 7: Check Output on Android Emulator.**

****