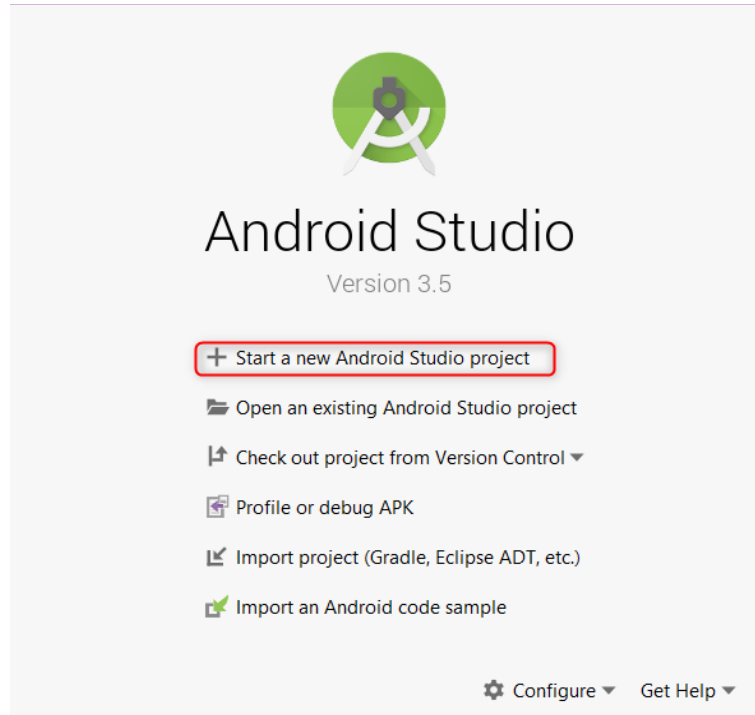
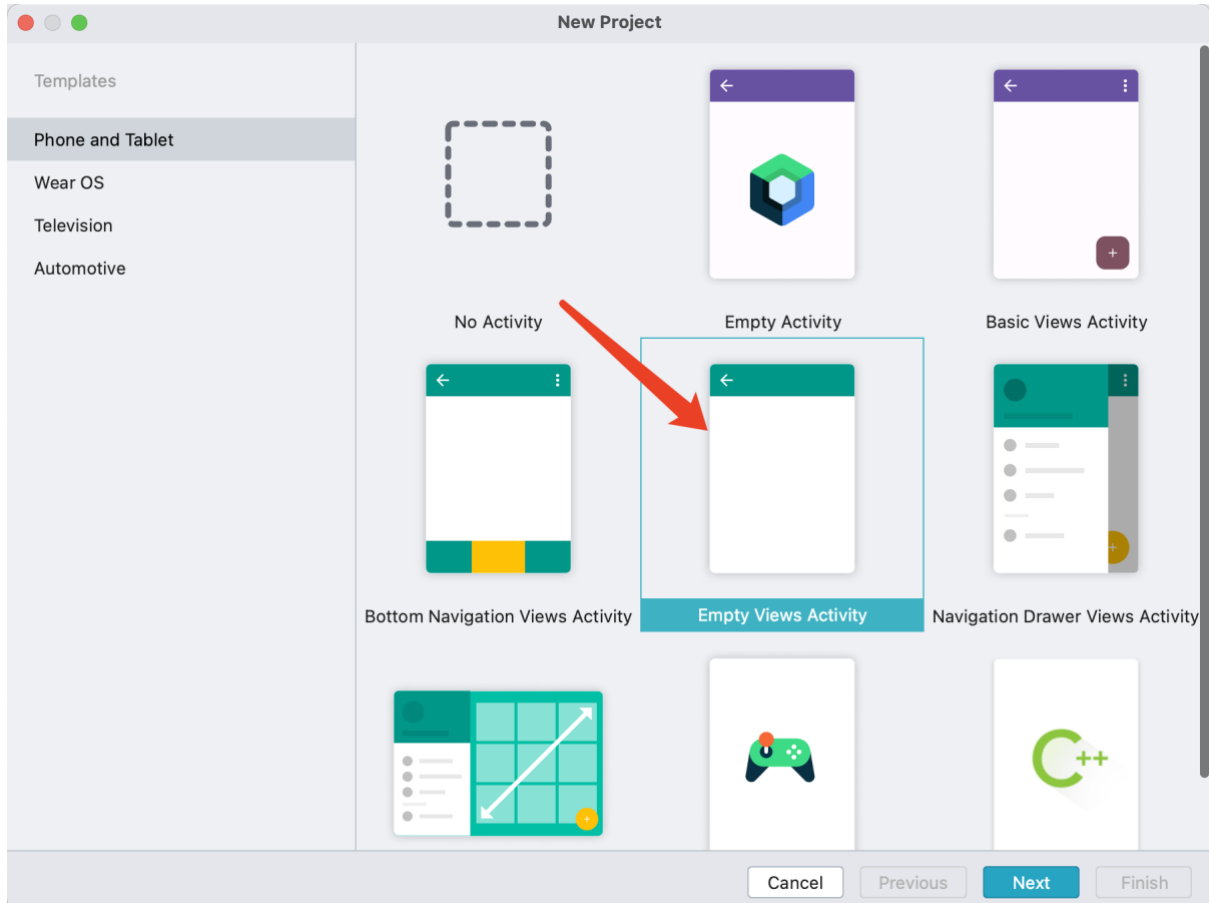


1. Create a new project in Android Studio as shown below by clicking on '**New Project**'.

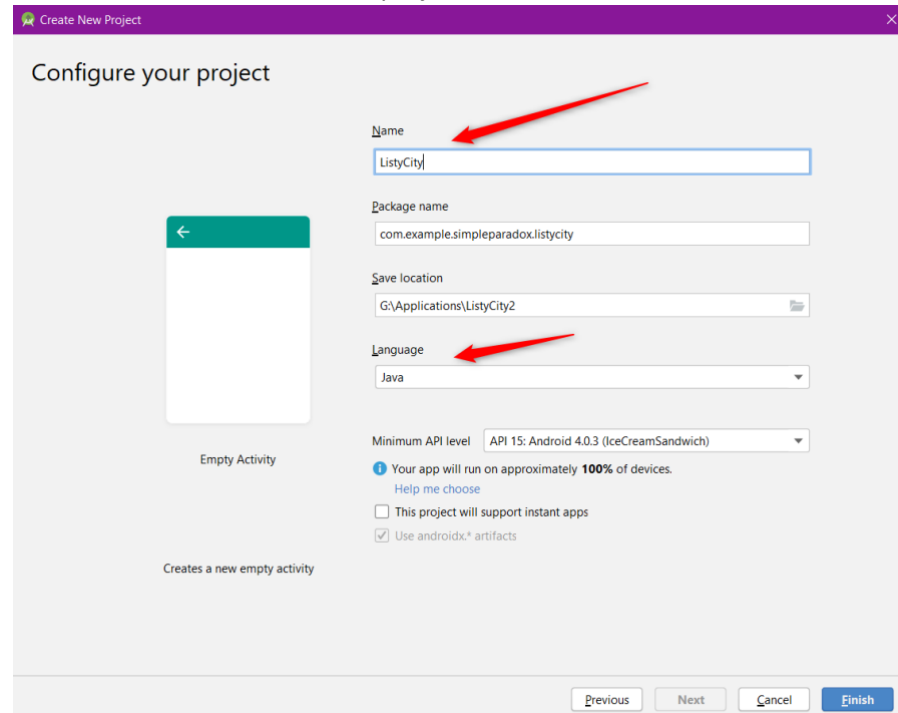


2. Select **Empty Views Activity** under the 'Phone and Tablet' tab. Click Next.

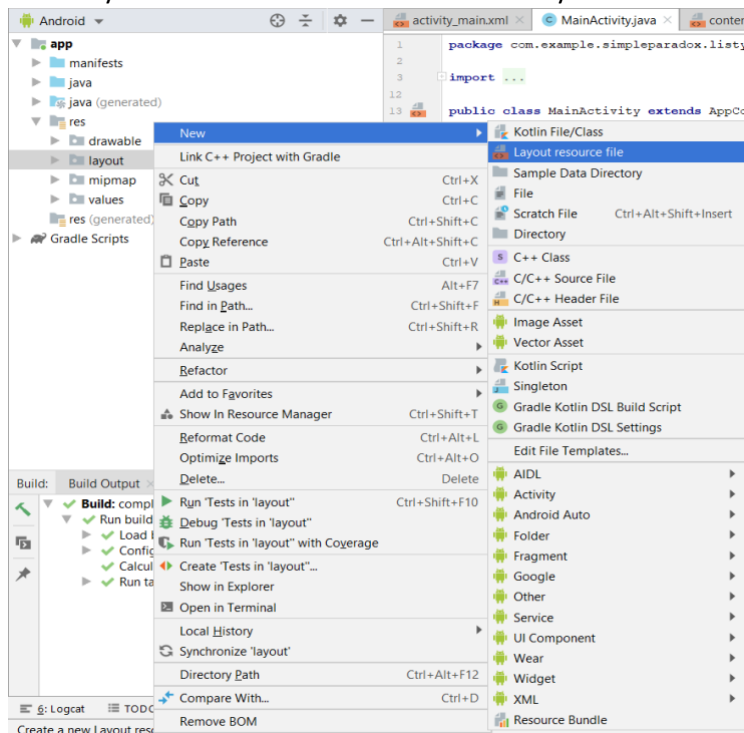


3. Configure your project.
 - a. Give the project a name.
 - b. Make sure the language is Java
 - c. The minimum API level should be enough so that the application runs on most devices.

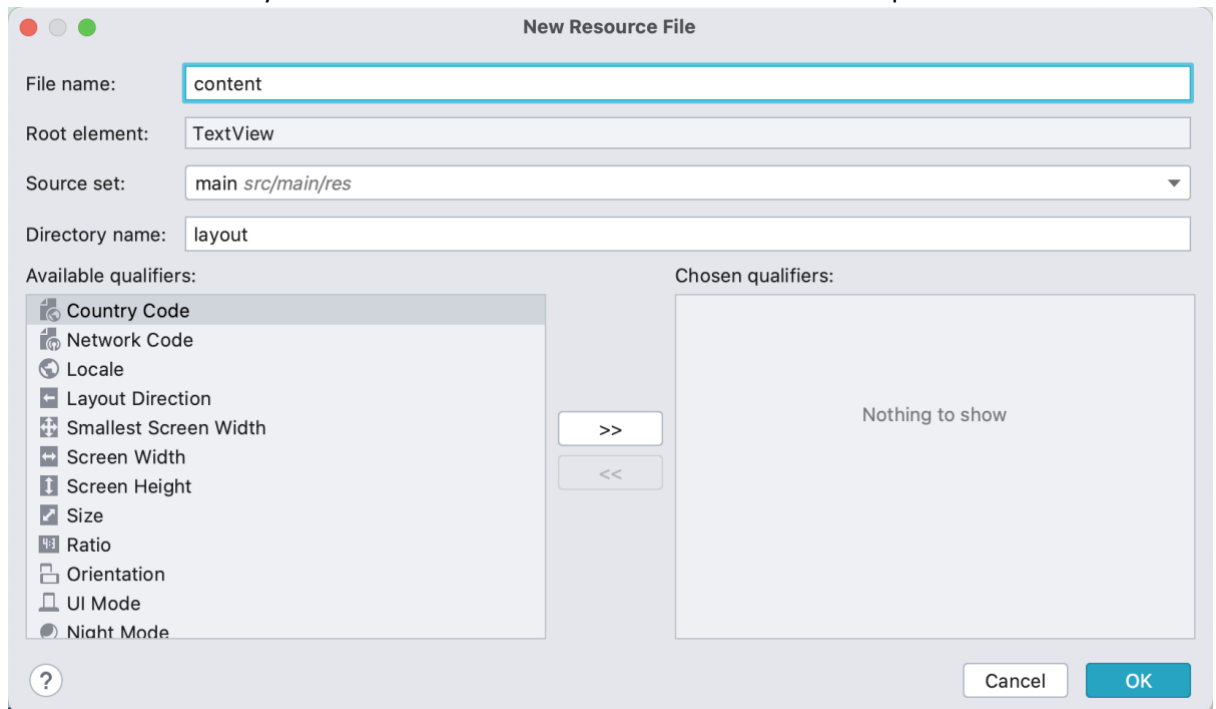
Click on finish and wait for the project to build itself.



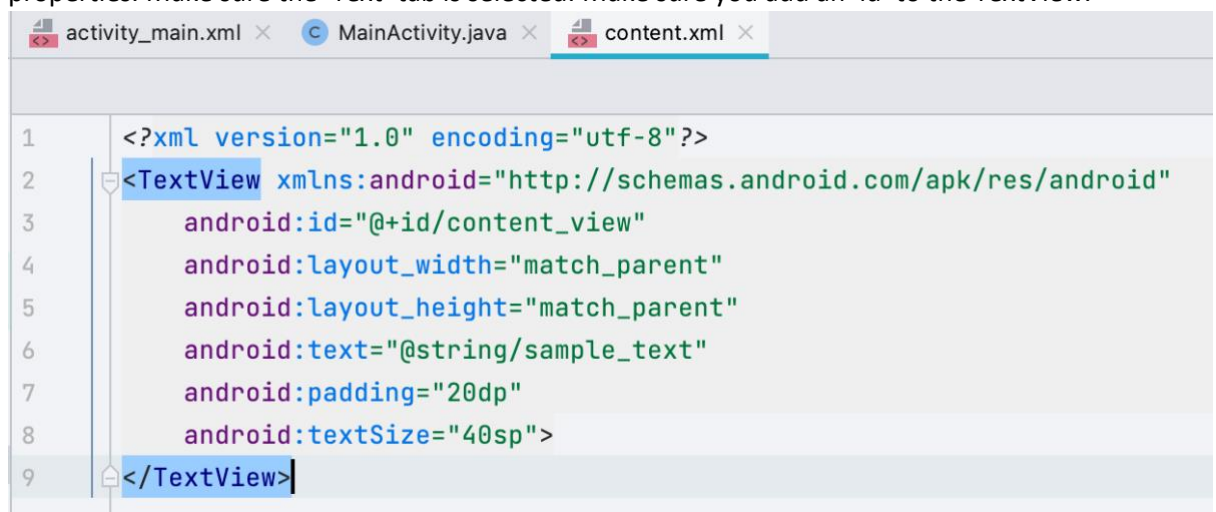
4. Navigate to app/res/layout directory in the 'Project' pane. Then right click on the 'layout' directory and then click on 'New' and then 'Layout Resource file'



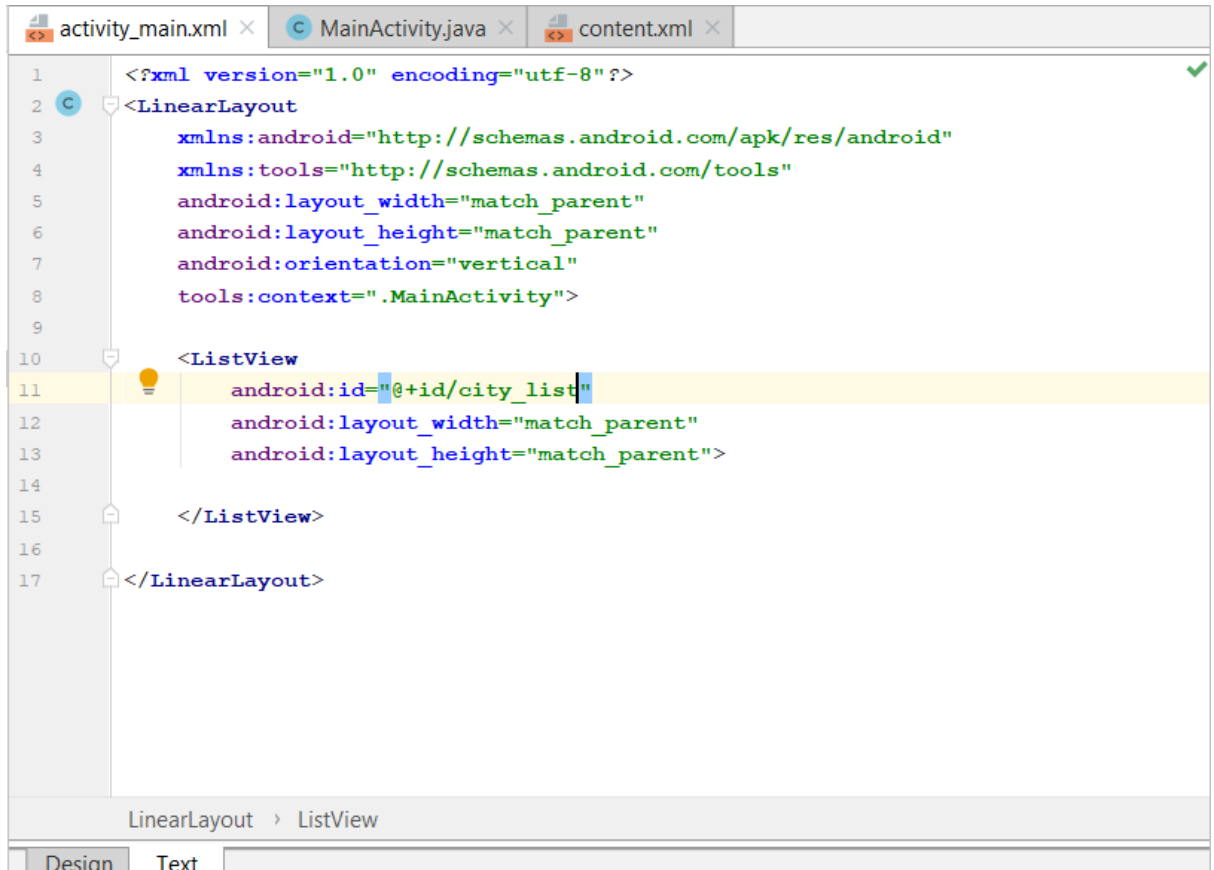
5. Give a name to the layout resource file with the extension '.xml' and then press 'OK'.



6. Now go to 'content.xml' under res/layout and then add a TextView with the following properties. Make sure the 'Text' tab is selected. Make sure you add an 'id' to the TextView.

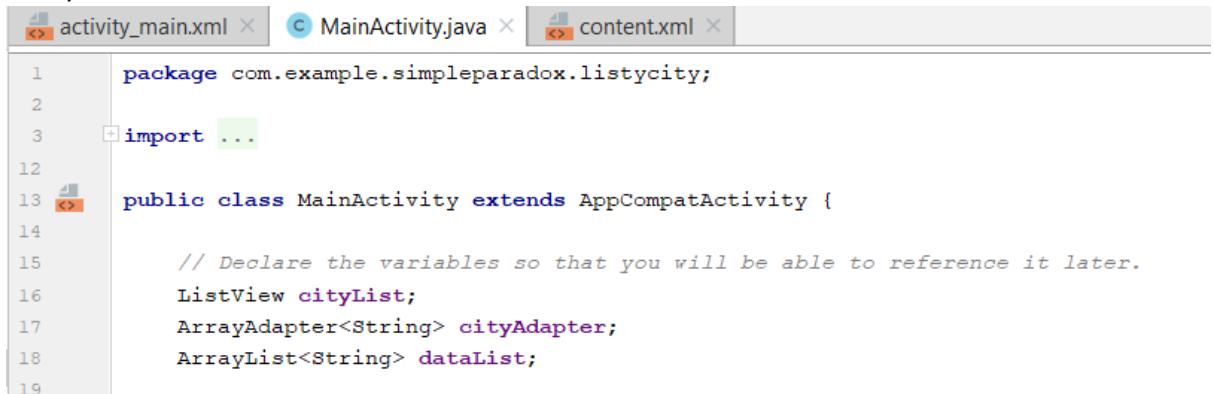


7. Go to `activity_main.xml` and add a 'ListView' inside the 'LinearLayout' or 'ConstraintLayout' viewgroup. Make sure you add an 'id' attribute to the 'ListView'.



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:orientation="vertical"
8     tools:context=".MainActivity">
9
10    <ListView
11        android:id="@+id/city_list"
12        android:layout_width="match_parent"
13        android:layout_height="match_parent">
14
15    </ListView>
16
17 </LinearLayout>
```

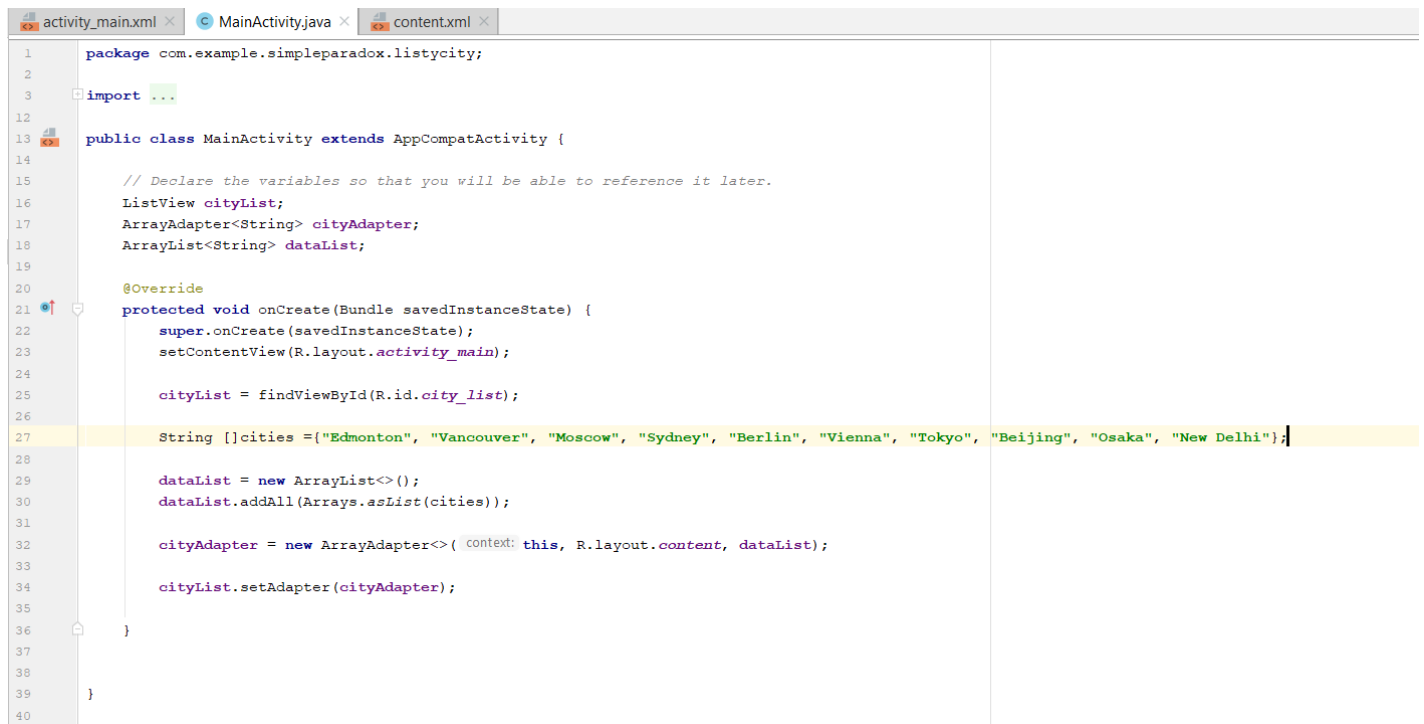
8. Now let's head over to `MainActivity.java` file which will contain the logic for mapping the data to the 'ListView' so that it can be shown as a scrolling list.
9. Create References for the 'ListView' along with a reference for 'ArrayAdapter' and an 'ArrayList'.



```
1 package com.example.simpleparadox.listycity;
2
3 import ...
4
12
13 public class MainActivity extends AppCompatActivity {
14
15     // Declare the variables so that you will be able to reference it later.
16     ListView cityList;
17     ArrayAdapter<String> cityAdapter;
18     ArrayList<String> dataList;
19 }
```

10. Now inside the onCreate() method, we write the logic that will help to bind the data to the 'ListView'.

- First find the reference to the 'ListView' using findViewById() and assign it to the reference 'cityList'.
- Then declare a string array consisting of cities which can be fed into the 'ListView' later.
- Create a new 'ArrayList' and assign it to the reference 'dataList'. This will contain the data (the string array of cities).
- Add the data(string array containing city names) to the 'dataList' as shown in the picture below.
- Now we have to link the content.xml to the 'dataList' so that each element will be displayed in a separate row in the list.
- Finally, we connect the 'ListView' to the 'ArrayAdapter'(cityAdapter) which will show each 'TextView' in the form of scrolling list.



```
1 package com.example.simpleparadox.listycity;
2
3 import ...
4
12
13 public class MainActivity extends AppCompatActivity {
14
15     // Declare the variables so that you will be able to reference it later.
16     ListView cityList;
17     ArrayAdapter<String> cityAdapter;
18     ArrayList<String> dataList;
19
20     @Override
21     protected void onCreate(Bundle savedInstanceState) {
22         super.onCreate(savedInstanceState);
23         setContentView(R.layout.activity_main);
24
25         cityList = findViewById(R.id.city_list);
26
27         String []cities ={"Edmonton", "Vancouver", "Moscow", "Sydney", "Berlin", "Vienna", "Tokyo", "Beijing", "Osaka", "New Delhi"};
28
29         dataList = new ArrayList<>();
30         dataList.addAll(Arrays.asList(cities));
31
32         cityAdapter = new ArrayAdapter<>( context: this, R.layout.content, dataList);
33
34         cityList.setAdapter(cityAdapter);
35     }
36
37
38
39 }
40
```