LECTURE 16: UI PROGRAMMING PART 3

BY LINA HAMMAD & AHMAD BARGHASH

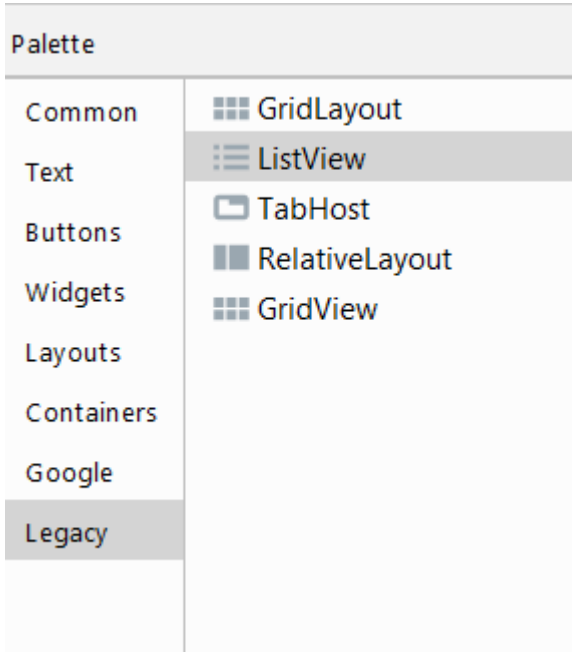
In this tutorial, we shall learn how to display elements of an array using Android ListView with the help of Kotlin Android Application. Then we will proceed further by adding ListView Item Click Listener so that a particular action would be taken when a click is made on an item in ListView

LEGACY – LISTVIEW

- Android **ListView** is a view which groups several items and display them in vertical scrollable list.

- key attributes:

android:clickable="***bool***"	set to false to disable the list
android:id="@+id/***theID***"	unique ID for use in Kotlin code
android:entries="@array/***array***"	set of options to appear in the list (must match an array in strings.xml)



LEGACY – LISTVIEW

- Xml code:

```
<ListView
        android:id="@+id/listView1"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>
```

- Kotlin code:

```
// use arrayadapter and define an array
val arrayAdapter: ArrayAdapter<*>
var cities = arrayOf("Melbourne", "Vienna", "Vancouver", "Toronto", "Calgary", "Adelaide",
        "Perth", "Auckland", "Helsinki", "Hamburg", "Munich", "New York", "Sydney",
        "Paris", "Cape Town", "Barcelona", "London", "Bangkok")

// access the listView from xml file
var myListView = findViewById<ListView>(R.id.listView1)
arrayAdapter = ArrayAdapter(this,
        android.R.layout.simple_list_item_1, cities)
myListView.adapter = arrayAdapter
```

Remember to add the Koltin code inside the onCreate function.

LEGACY – LISTVIEW

- On click item, kotlin code:

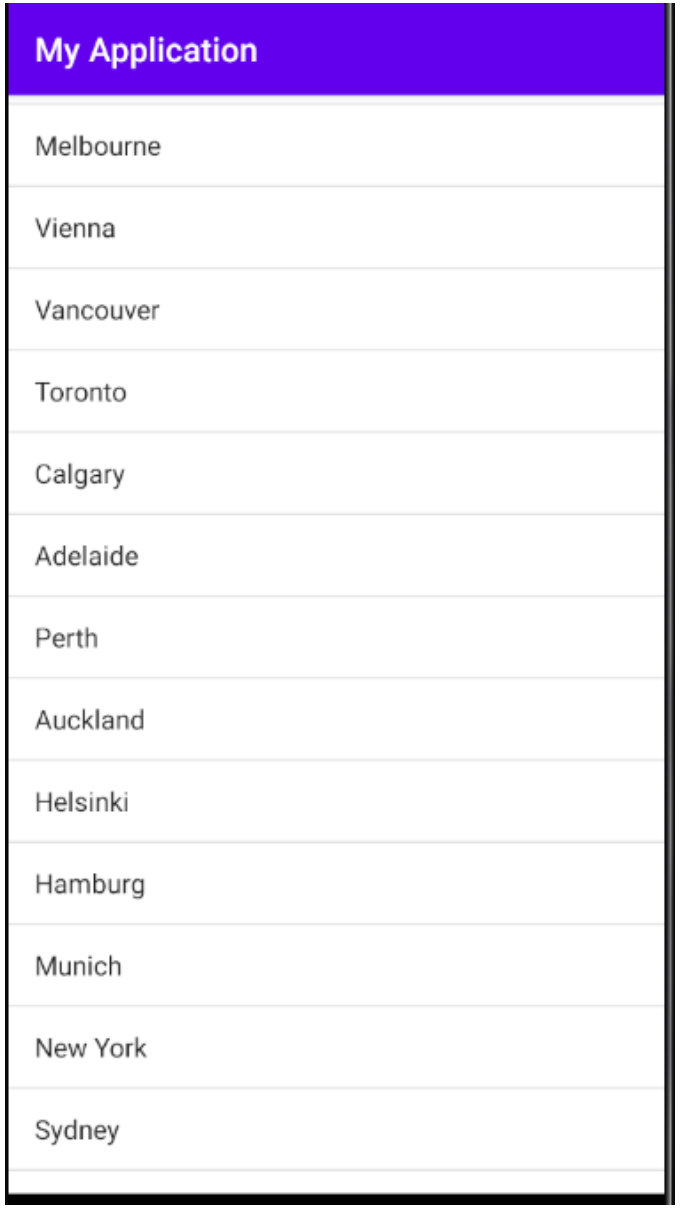
```
myListView.onItemClickListener = object : OnItemClickListener {
    override fun onItemClick(parent: AdapterView<*>, view: View,
                             position: Int, id: Long) {

        // value of item that is clicked
        val itemValue = myListView.getItemAtPosition(position) as String

        // Toast the values
        Toast.makeText(applicationContext,
                "Position :$position\nItem Value : $itemValue", Toast.LENGTH_LONG).show()
    }
}
```

LEGACY – LISTVIEW

After run the program



LIST ADAPTERS

- **adapter**: Helps turn list data into list view items.
- common adapters: ArrayAdapter, CursorAdapter
- Syntax for creating an adapter:

```
ArrayAdapter name = ArrayAdapter(activity, layout, array)
```

  - the ***activity*** is usually this
  - the default ***layout*** for lists is android.R.layout.simple_list_item_1
  - get the ***array*** by reading your file or data source of choice (it can be an array like String[ ], or a list like ArrayList<String>)
- Once you have an adapter, you can attach it to your list by calling the setAdapter method of the ListView object in the Kotlin code.

LIST EVENTS

- List views respond to the following events:
  1. **setOnItemClickListener**(AdapterView.OnItemClickListener)
     - Listener for when an item in the list has been clicked.
  2. **setOnItemLongClickListener**(AdapterView.OnItemLongClickListener)
     - Listener for when an item in the list has been clicked and held.
  3. **setOnItemSelectedListener**(AdapterView.OnItemSelectedListener)
     - Listener for when an item in the list has been selected.
  4. Others:
     - onDrag, onFocusChanged, onHover, onKey, onScroll, onTouch, ...