Android UI Control [Spinner]

Android – UI Control

Spinner (Drop down List)

- **Spinner** is a view that allows a user to select one value from the list of values.

- The spinner in android will behave same as a dropdown list in other programming languages.

- The android spinners will provide a quick way to select one item from the list of values and it will show a dropdown menu with a list of all values when we click or tap on it.

- By default, the android spinner will show its currently selected value and by using **Adapter** we can bind the items to spinner objects.

Android – UI Control

Spinner (Drop down List)

- Following is the pictorial representation of using **spinner** in android applications.

Android – UI Control

Spinner

- We can populate our **Spinner** control with list of choices by defining an **ArrayAdapter** in our Activity file.

- The **Adapter** pulls data from sources such as an array or database and converts each item into a result view and that's placed into the list.

Android – UI Control

Android Adapter

- **Adapter** will act as an intermediate between the data sources and adapter views such as **ListView**, **Gridview** to fill the data into adapter views.

- The adapter will hold the data and iterates through an items in data set and generate the views for each item in the list.

- In android we have a different types of adapters available to fetch the data from different data sources to fill the data into adapter views, those are

Adapter	Description
ArrayAdapter	It will expect an Array or List as input.
CurosrAdapter	It will accepts an instance of a cursor as an input.
SimpleAdapter	It will accept a static data defined in the resources.
BaseAdapter	It is a generic implementation for all three adapter types and it can be used for ListView, Gridview or Spinners based on our requirements

Android – UI Control

Create Android Spinner in XML Layout File

- We can create **Spinner** in XML layout file using **\<Spinner>** element with different attributes like as shown below.

```
<Spinner android:id="@+id/spinner1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
```

Populate Android Spinner with Values

- To populate spinner with list of values, we need to specify spinner adapter, such as an **ArrayAdapter** in activity file like as shown below.

```
String[] users = { "Suresh Dasari", "Trishika Dasari", "Rohini Alavala", "Praveen Kumar", "Madhav Sai" };
Spinner spin = (Spinner) findViewById(R.id.spinner1);
ArrayAdapter<String> adapter = new ArrayAdapter<String>(this, android.R.layout.simple_spinner_item, users);
adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
spin.setAdapter(adapter);
```

- This is how we can define and bind data to Spinner control in android applications.

Android Spinner Example

- In this example we define a one **Spinner** control, one TextView control in RelativeLayout to show the list of user details in android application.

- Create a new android application using android studio and give names as **SpinnerExample**.

- Now open an **activity_main.xml** file from \res\layout path and write the code like as shown below

Android Spinner Example

```
activity_main.xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com
/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent">
   <TextView
      android:id="@+id/txtVw"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_marginLeft="50dp"
      android:layout_marginTop="150dp"
      android:text="Select User:"
      android:textStyle="bold"
      android:textSize="15dp" />
```

```
<Spinner
      android:id="@+id/spinner1"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_alignBottom="@+id/txtVw"
      android:layout_toRightOf="@+id/txtVw" />
</RelativeLayout>
```

Android Spinner Example

```
MainActivity.java
package com.tutlane.spinnerexample;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Spinner;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements
AdapterView.OnItemSelectedListener {
String[] users = { "Suresh Dasari", "Trishika Dasari", "Rohini Alavala", "Praveen Kumar",
"Madhav Sai" };
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_main);
```

Android Spinner Example

```
Spinner spin = (Spinner) findViewById(R.id.spinner1);
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
    android.R.layout.simple_spinner_item, users);
    adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_
    item);
    spin.setAdapter(adapter);
    spin.setOnItemSelectedListener(this);
}
@Override
public void onItemSelected(AdapterView<?> arg0, View arg1, int position,long id)
```
//**parent :** AdapterView: The AdapterView where the selection happened,
View : The //view within the AdapterView that was clicked, **position:**int: The position of
the view //in the adapter, **id:** ong: The row id of the item that is selected
```
{
    Toast.makeText(getApplicationContext(), "Selected User: "+users[position]
    ,Toast.LENGTH_SHORT).show();
}
```

Android Spinner Example

```
@Override
public void onNothingSelected(AdapterView<?> arg0) {
    // TODO - Custom Code
    //Callback method to be invoked when the selection disappears from this view.
    //The selection can disappear for instance when touch is activated or when the
    //adapter becomes empty.
}
}
```

Output of Android Spinner Example

13

14