



Android UI Control

[RadioButton]

Android – UI Control

RadioButton

- **Radio Button** is a two-states button that can be either checked or unchecked and it's the same as **CheckBox** control, except that it will allow only one option to select from the group of options.
- The user can press or click on the radio button to make it select. In android, checkBox control allow users to change the state of control either Checked or Unchecked but the radio button cannot be unchecked once it is checked.
- Generally, we can use **RadioButton** controls in an android application to allow users to select only one option from the set of values.
- Following is the pictorial representation of using **RadioButton** control in android application



Android – UI Control

RadioButton

- In android, we use radio buttons with in a **RadioGroup** to combine multiple radio buttons into one group and it will make sure that users can select only one option from the group of multiple options.
- By default, the android **RadioButton** will be in **OFF (Unchecked)** state. We can change the default state of **RadioButton** by using **android:checked** attribute.
- In case, if we want to change the state of **RadioButton** to **ON (Checked)**, then we need to set **android:checked = "true"** in our XML layout file.
- In android, we can create **RadioButton** control in two ways either in the XML layout file or create it in the Activity file programmatically.

3

Android – UI Control

Create RadioButton in XML Layout File

```
<?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">
    <RadioGroup
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <RadioButton
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Java"
            android:onClick="onRadioButtonClicked"/>
    </RadioGroup>
</RelativeLayout>
```

4

Create RadioButton Control in Activity File

- In Activity that hosts our XML layout file, we need to implement click event method like as shown below.

```
public void onRadioButtonClicked(View view) {  
    // Is the view now checked?  
    boolean checked = ((RadioButton) view).isChecked();  
    // Check which RadioButton was clicked  
    switch(view.getId()) {  
        case R.id.chk1:  
            if (checked)  
                // Do your coding  
            else  
                // Do your coding  
            break;  
        // Perform your logic  
    }  
}
```

5

Define RadioButton Click Event in Activity File

- We can define **RadioButton** click event programmatically in Activity file rather than XML layout file.
- To define **RadioButton** click event programmatically, create **View.OnClickListener** object and assign it to the button by calling **setOnClickListener(View.OnClickListener)** like as shown below.

```
RadioButton rdb = (RadioButton) findViewById(R.id.radiobutton1);  
rdb.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        boolean checked = ((RadioButton) v).isChecked();  
        // Check which radiobutton was pressed  
        if (checked){  
            // Do your coding  
        }  
        else{  
            // Do your coding  
        }  
    }  
});
```

6

Android – UI Control

Android RadioButton Control Attributes

Attribute	Description
android:id	It is used to uniquely identify the control
android:checked	It is used to specify the current state of radio button
android:gravity	It is used to specify how to align the text like left, right, center, top, etc.
android:text	It is used to set the text for the radio button.
android:textColor	It is used to change the color of text.
android:textSize	It is used to specify the size of the text.
android:textStyle	It is used to change the style (bold, italic, bolditalic) of text.

7

Android – UI Control

Android RadioButton Control Attributes

Attribute	Description
android:background	It is used to set the background color for radio button control.
android:padding	It is used to set the padding from left, right, top and bottom.
android:onClick	It's the name of the method to invoke when the radio button clicked.
android:visibility	It is used to control the visibility of control.

8

Android RadioButton Control Example

- In this example we define a multiple **RadioButton** controls, one **TextView** control and one **Button** control in RelativeLayout to get the selected values of RadioButton controls when we click on Button in the android application.
- Create a new android application using android studio and give names as **RadioButtonExample**.
- Now open an **activity_main.xml** file from \res\layout path and write the code like as shown below

9

Android RadioButton Control 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:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="150dp"
        android:layout_marginLeft="100dp"
        android:textSize="18dp"
        android:text="Select Your Course"
        android:textStyle="bold"
        android:id="@+id/txtView"/>
    <RadioGroup
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:id="@+id/rdGroup"
        android:layout_below="@+id/txtView">
        <RadioButton
            android:id="@+id/rdbJava"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:padding="10dp"
            android:layout_marginLeft="100dp"
            android:text="Java"
            android:onClick="onRadioButtonClicked"/>
        <RadioButton
            android:id="@+id/rdbPython"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:padding="10dp"
            android:layout_marginLeft="100dp"
            android:text="Python"
            android:onClick="onRadioButtonClicked"/>
    </RadioGroup>
</RelativeLayout>
```

10

Android – UI Control

Android RadioButton Control Example

```
<RadioButton
    android:id="@+id/rdbAndroid"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="10dp"
    android:layout_marginLeft="100dp"
    android:text="Android"
    android:onClick="onRadioButtonClicked"/>
<RadioButton
    android:id="@+id/rdbAngular"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:padding="10dp"
    android:layout_marginLeft="100dp"
    android:text="AngularJS"
    android:onClick="onRadioButtonClicked"/>
</RadioGroup>

<Button
    android:id="@+id/getBtn"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="100dp"
    android:layout_below="@+id/rdGroup"
    android:text="Get Course" />
</RelativeLayout>
```

11

Android – UI Control

Android RadioButton Control Example

```
MainActivity.java
package com.tutlane.radiobuttonexample;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.RadioButton;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    RadioButton android, java, angular, python;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        android = (RadioButton)findViewById(R.id.rdbAndroid);
        angular = (RadioButton)findViewById(R.id.rdbAngular);
        java = (RadioButton)findViewById(R.id.rdbJava);
        python = (RadioButton)findViewById(R.id.rdbPython);
        Button btn = (Button)findViewById(R.id.getBtn);
```

12

Android – UI Control

Android RadioButton Control Example

```
btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String result = "Selected Course: ";
        result+= (android.isChecked())?"Android":(angular.isChecked())?"AngularJS":
(java.isChecked())?"Java":(python.isChecked())?"Python":"";
        Toast.makeText(getApplicationContext(), result,
            Toast.LENGTH_SHORT).show();
    }
});
}

public void onRadioButtonClicked(View view) {
    boolean checked = ((RadioButton) view).isChecked();
    String str="";
    // Check which radio button was clicked
    switch(view.getId()) {
        case R.id.rdbAndroid:
            if(checked)
                str = "Android Selected";
            break;

```

13

Android – UI Control

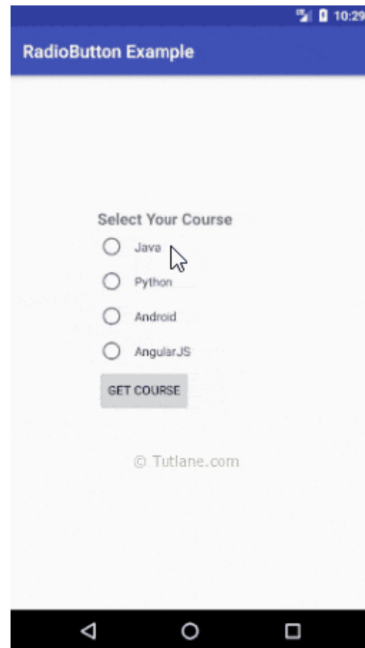
Android RadioButton Control Example

```
        case R.id.rdbAngular:
            if(checked)
                str = "AngularJS Selected";
            break;
        case R.id.rdbJava:
            if(checked)
                str = "Java Selected";
            break;
        case R.id.rdbPython:
            if(checked)
                str = "Python Selected";
            break;
    }
    Toast.makeText(getApplicationContext(), str, Toast.LENGTH_SHORT).show();
}
}
```

14

Android – UI Control

Output of Android RadioButton Example

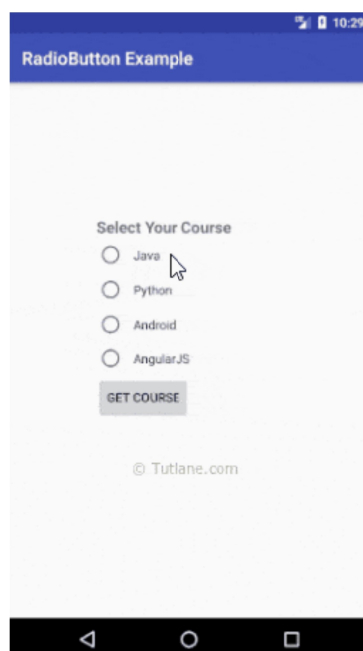


15

Android – UI Control

Output of Android RadioButton Example

(Once we enter details in all fields and click on Button we will get a result like as shown below.)



16

