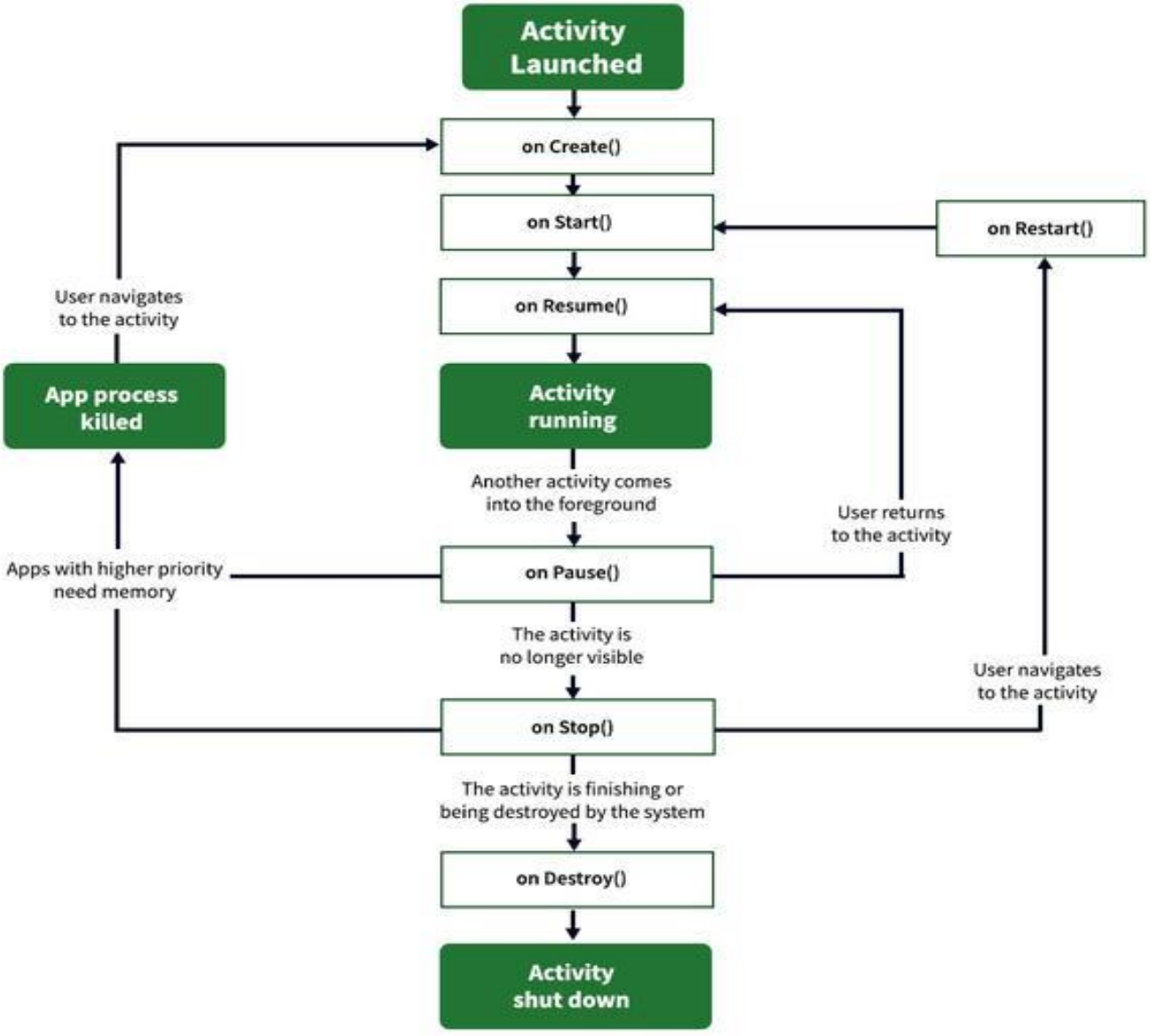
Unit 4 : Android Activity

1. Activity Lifecycle

Every instance od activity has a lifecycle. It is the series of stages that an activity goes through from creation to destruction. During this lifecycle, an activity transitions between three possible states: running, paused and stopped.

Here are the key lifecycle methods:

- **onCreate()**: Called when the activity is first created. You should initialize your UI components here.
- **onStart()**: Called when the activity becomes visible to the user.
- **onResume()**: Called when the activity starts interacting with the user.
- **onPause()**: Called when the system is about to put the activity into the background.
- **onStop()**: Called when the activity is no longer visible to the user.
- **onDestroy()**: Called before the activity is destroyed.
- **onRestart()**: Called when the activity is stopped and then started again.



Activity Lifecycle in Android

Here’s an example in code:

```
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        // Initialization code here
    }
    @Override
    protected void onStart() {
        super.onStart();
        // Activity becomes visible
    }
    @Override
    protected void onResume() {
        super.onResume();
        // Activity starts interacting with the user
    }
    @Override
    protected void onPause() {
        super.onPause();
        // Activity is partially obscured
    }
    @Override
    protected void onStop() {
        super.onStop();
        // Activity is no longer visible
    }
    @Override
    protected void onDestroy() {
        super.onDestroy();
        // Cleanup resources
    }
    @Override
    protected void onRestart() {
        super.onRestart();
        // Activity is being restarted
    }
}
```

2. Creating Multiple Activities

To create multiple activities, you need to define new activities in your project. Here’s how you can create a second activity:(assume you already have MainActivity that contains Button in layout).

1. Create a new Activity:

```
public class SecondActivity extends AppCompatActivity {
```

```
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_second);
    }
}
```

2. Create a corresponding XML layout file activity_second.xml:

```
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center">

    <TextView
        android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="This is the Second Activity"
        android:textSize="24sp" />
</LinearLayout>
```

3. Declaring Activities in the Manifest

To declare an activity in your app, add it to the AndroidManifest.xml file:

```
    <application
        android:allowBackup="true"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".SecondActivity"></activity>
    </application>
```

Here, the MainActivity is the launcher activity, and SecondActivity is also declared in the manifest.

3. Connecting Activities with Intents

You can connect one activity to another using **Intent**. An **Intent** is an object that the component can use to communicate with the OS. Intent tells the ***ActivityManager*** which activity to start. Here's how you can start ***SecondActivity*** from ***MainActivity:***

```
Button btn = findViewById(R.id.button);
btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent i = new Intent(MainActivity.this, SecondActivity.class);
        startActivity(i);
    }
});
```

Here is a complete example:

For this purpose, create two activities named "MainActivity" and "SecondActivity". MainActivity contains TextView and Button. On the other hand, SecondActivity contains a TextView.

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:orientation="vertical"
   xmlns:android="http://schemas.android.com/apk/res/android">
      <TextView
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="This is 1st Activity"
         android:textSize="20sp"
         android:textStyle="bold"
         android:layout_marginTop="100dp"
         android:layout_gravity="center"/>
      <Button
         android:id="@+id/btn"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Click Me"
         android:layout_gravity="center"
         />
</LinearLayout>
```

activity_second.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:orientation="vertical"
   xmlns:android="http://schemas.android.com/apk/res/android">
   <TextView
      android:id="="@+id/text2"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="This is 2nd Activity"
      android:textSize="20sp"
      android:textStyle="bold"
      android:layout_marginTop="100dp"
      android:layout_gravity="center"/>
</LinearLayout>
```

MainActivity.java

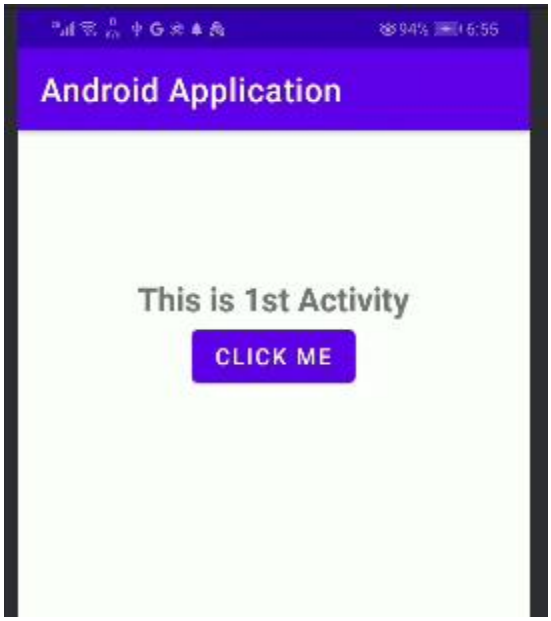
```
package com.example.androidapplication;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

   @Override
   protected void onCreate(Bundle b) {
      super.onCreate(b);
      setContentView(R.layout.activity_main);
   Button btn = findViewById(R.id.btn);
   btn.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View view) {
         Intent i = new Intent(MainActivity.this , SecondActivity.class);
         startActivity(i);
      }
   });

   }
}
```

SecondActivity.java

```
package com.example.androidapplication;

import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;

public class SecondActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_second);
    }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.AndroidApplication"
        tools:targetApi="31">
        <activity
            android:name=".SecondActivity"
            android:exported="true" />
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```



5. Passing Data Between Activities

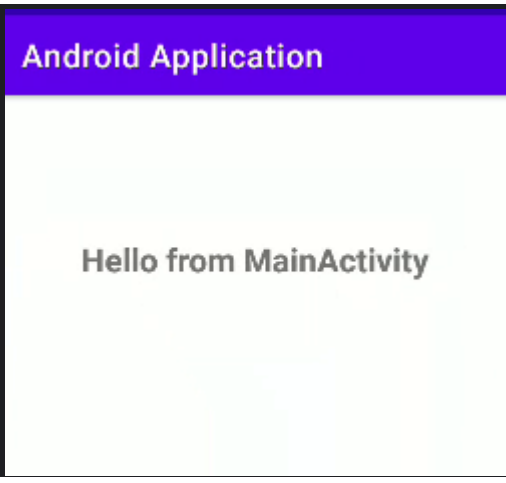
You can pass and receive data between activities using Intent extras. Extras are the arbitrary data that the calling activity can include with an intent. The OS forwards the intent to the recipient activity, which can then access the extra and retrieve the data.
An extra is structure as a key-value pair. To add an extra to an intent, use **Intent.putExtra(Key, Value).**

MainActivity.java

```
Intent i = new Intent(MainActivity.this , SecondActivity.class);
        //passing data
i.putExtra("message", "Hello from MainActivity");
startActivity(i);
```

SecondActivity.java

```
@Override
protected void onCreate(Bundle b) {
    super.onCreate(b);
    setContentView(R.layout.activity_second);
        //receiving data
    Intent i = getIntent();
    String msg = i.getStringExtra("message");
        //displaying received data in TextView of second activity
    TextView txt = findViewById(R.id.text2);
    txt.setText(msg);
}
```



6. Getting a Result Back from a Child Activity

To get a result back from an activity, use startActivityForResult() in the parent activity and setResult() in the child activity:

MainActivity.java

```
public class MainActivity extends AppCompatActivity {
   TextView txt;
   @Override
   protected void onCreate(Bundle b) {
      super.onCreate(b);
      setContentView(R.layout.activity_main);

      txt = findViewById(R.id.text);
      Button btn = findViewById(R.id.btn);
      btn.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View view) {
            Intent i = new Intent(MainActivity.this, SecondActivity.class);
            //starting activity with result code 2
            startActivityForResult(i, 2);
         }
      });
   }//Call Back method to get the message from other Activity
   @Override
   protected void onActivityResult(int requestCode, int resultCode,  Intent data) {
      super.onActivityResult(requestCode, resultCode, data);
      if (requestCode == 2 ) {
         String msg = data.getStringExtra("message");
         txt.setText(msg);
      }
   }
}
```

SecondActivity.java

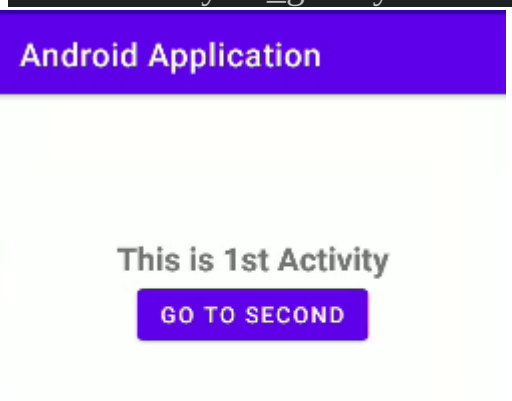
```
public class SecondActivity extends AppCompatActivity {
   Button btn;
    @Override
    protected void onCreate(Bundle b) {
        super.onCreate(b);
        setContentView(R.layout.activity_second);
btn = findViewById(R.id.btn2);
btn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent i = new Intent();
        i.putExtra("message", "Hello from first activity");
        setResult(2,i);
        finish();
            }
        });
    }
}
```

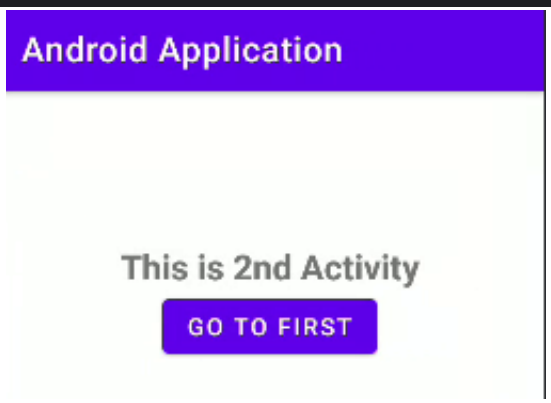
activity_main.xml

```
<TextView
   android:id="@+id/text"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="This is 1st Activity"
   android:textSize="20sp"
   android:textStyle="bold"
   android:layout_marginTop="100dp"
   android:layout_gravity="center"/>
<Button
   android:id="@+id/btn"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="Go to Second"
   android:layout_gravity="center" />
```

activity_second.xml

```
<TextView
   android:id="@+id/text2"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="This is 2nd Activity"
   android:textSize="20sp"
   android:textStyle="bold"
   android:layout_marginTop="100dp"
   android:layout_gravity="center"/>
<Button
   android:id="@+id/btn2"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:text="Go to First"
   android:layout_gravity="center" />
```





7. Getting and Setting Data to/from the Layout File

To get and set data from layout views like TextView, EditText, etc., use the findViewById() method:

```
// Setting data
TextView txt = findViewById(R.id.textView);
txt.setText("New Text");

// Getting data
EditText txt = findViewById(R.id.editText);
String input = txt.getText().toString();
```

These methods allow you to manipulate the UI elements dynamically based on user input or other conditions.

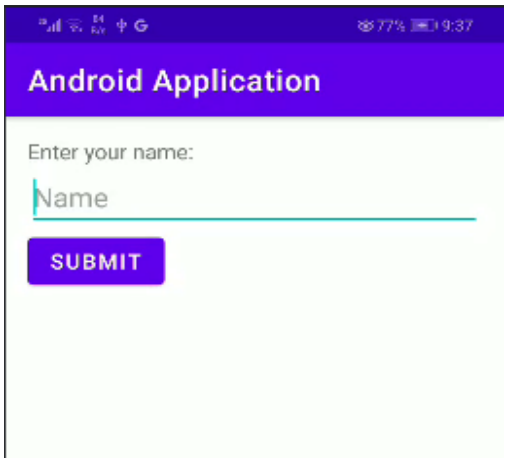
Example:

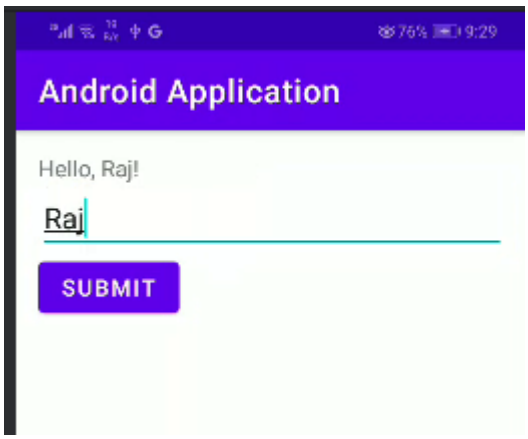
Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_height="match_parent"
   android:layout_width="match_parent"
   android:orientation="vertical"
   android:padding="16dp">

      <TextView
         android:id="@+id/textView"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Enter your name:" />

      <EditText
         android:id="@+id/editText"
         android:layout_width="match_parent"
         android:layout_height="wrap_content"
         android:hint="Name" />

      <Button
         android:id="@+id/button"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Submit" />
</LinearLayout>
```

MainActivity.java

```
package com.example.androidapplication;
import android.os.Bundle;
import android.view.View;
import android.widget.*;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

        EditText editText;
        Button button;
        TextView textView;

   @Override
   protected void onCreate(Bundle b) {
      super.onCreate(b)
      setContentView(R.layout.activity_main);

      editText = findViewById(R.id.editText);
      button = findViewById(R.id.button);
      textView = findViewById(R.id.textView);

      button.setOnClickListener(new View.OnClickListener() {
          @Override
         public void onClick(View v) {
            String name = editText.getText().toString();
            textView.setText("Hello, " + name + "!");
         }
      });
   }
}
```





Example 2:

Activity_main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    xmlns:android="http://schemas.android.com/apk/res/android">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Student Form"
        android:textStyle="bold"
        android:textSize="20sp"
        android:layout_gravity="center"/>
    <EditText
        android:id="@+id/id"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter your ID"
        android:layout_marginTop="50dp"
        android:layout_gravity="center"/>
    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Your name"
        android:layout_gravity="center"/>
<RadioGroup
    android:id="@+id/gender"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Gender : "
        android:textStyle="bold"
        android:textSize="20sp" />
    <RadioButton
        android:id="@+id/male"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Male"/>
    <RadioButton
        android:id="@+id/female"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Female"/>
</RadioGroup>
    <Spinner
        android:id="@+id/subject"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:entries="@array/subjects"/>
    <Button
        android:id="@+id/btn"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit"
        android:layout_gravity="center"/>
    <TextView
        android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=""
        android:textStyle="bold"
        android:textSize="20sp"
        android:layout_gravity="center"/>
</LinearLayout>
```

MainActivity.java

```
package com.example.practiceset;

import android.annotation.SuppressLint;
import android.os.Bundle;
import android.view.View;
import android.widget.*;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
EditText id, name;
RadioGroup gender;
Button btn;
Spinner subject;
TextView text;
    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        id = findViewById(R.id.id);
        name = findViewById(R.id.name);
        gender = findViewById(R.id.gender);
        subject = findViewById(R.id.subject);
        text = findViewById(R.id.text);
        btn = findViewById(R.id.btn);
        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
              String  stdId = id.getText().toString();
              String stdName = name.getText().toString();
              String stdGender = ((RadioButton) findViewById(gender.getCheckedRadioButtonId())).
               getText().toString();
              String stdSubject = subject.getSelectedItem().toString();
              text.setText(" ID : "+stdId +"\n Name : "+stdName+"\n Gender : "+stdGender+"\n subject
               : "+stdSubject);
            }
        });
    }
}
```