

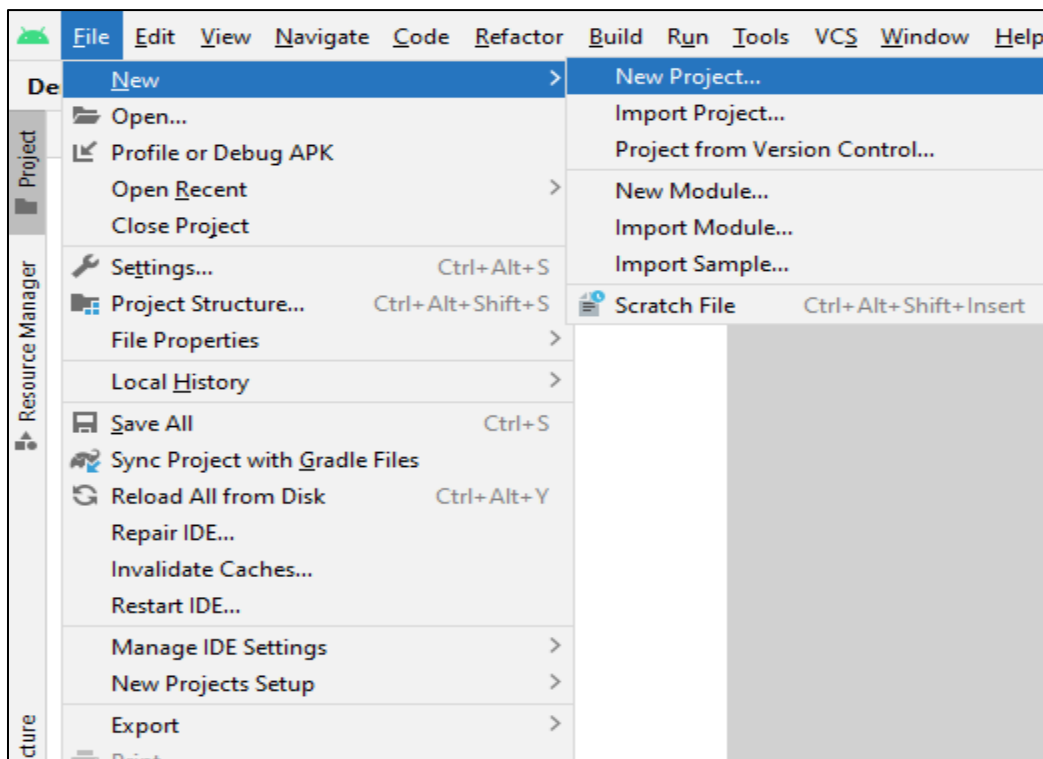
Extra Ex 2: Working with Intent Bundle Passing

Introduction

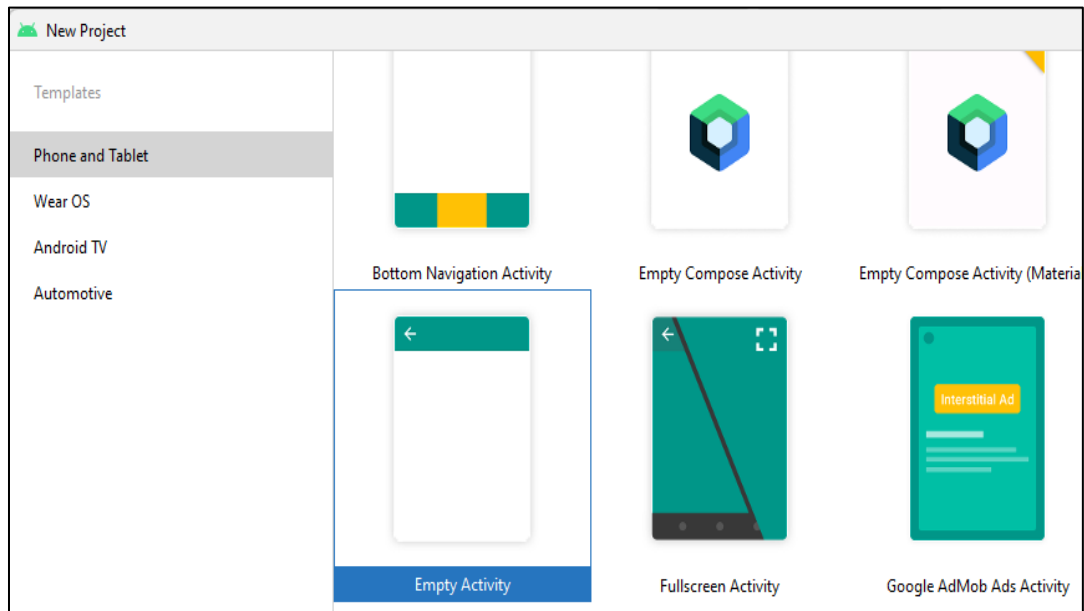
Bundle Passing is used to send data from one Activity to another Activity.

Let's get Started:

Step 1: Create a New Project in Android Studio as shown below



Step 2: Select Empty Activity as shown below



Step 3: Provide a Project Name as shown below

The screenshot shows the 'Empty Activity' configuration screen. At the top, it says 'Empty Activity' and 'Creates a new empty activity'. Below this are several input fields: 'Name' (containing 'DemoIntent'), 'Package name' (containing 'com.example.demointent'), 'Save location' (containing 'C:\Users\hksharma\AndroidStudioProjects\DemoIntent2'), 'Language' (set to 'Java'), and 'Minimum SDK' (set to 'API 26: Android 8.0 (Oreo)'). There is an information icon and text stating 'Your app will run on approximately 88.2% of devices.' with a link 'Help me choose'. Below this is a checkbox 'Use legacy android.support libraries' which is unchecked, with a note that using these libraries prevents the use of the latest Play Services and Jetpack libraries. At the bottom right, there are four buttons: 'Previous', 'Next', 'Cancel', and 'Finish'.

Step 4: Update MainActivity.java as per the code given below

```
package com.example.demointent;
import androidx.appcompat.app.AppCompatActivity;

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

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Button btnNext= findViewById(R.id.btnNext);
        btnNext.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intentNext;
                intentNext = new Intent(MainActivity.this,
SecondActivity.class);
                intentNext.putExtra("myName", "Hitesh");
                startActivity(intentNext);
            }
        });
    }
}
```

Step 5: Update activity_main.xml for Main Activity as per the code given below

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    tools:context=".MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Main Activity"
        android:textSize="19sp"/>
    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="NEXT"
        android:id="@+id/btnNext"/>
</LinearLayout>
```

Step 6: Create second Activity SecondActivity.java as per the code given below

```
package com.example.demointent;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class SecondActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_second);
        Button btnBack = findViewById(R.id.btnBack);
        TextView txtName= findViewById(R.id.txtName);
        Intent formIntent=getIntent();
        String name= formIntent.getStringExtra("myName");
        txtName.setText("Name="+name);
        btnBack.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                Intent intentBack = new
                Intent(SecondActivity.this,MainActivity.class);
                startActivity(intentBack);

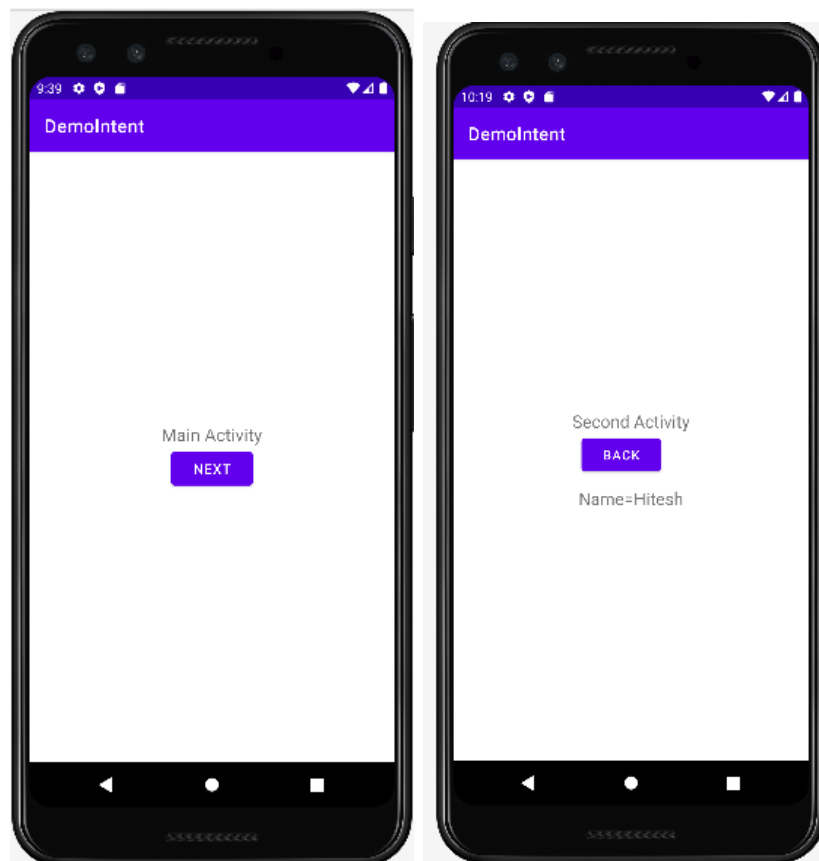
            }
        });
    }
}
```

Step 7: Update activity_Second.xml for Second Activity as per the code given below

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".SecondActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Second Activity"
        android:textSize="19sp">
```

```
        />
        <Button
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="BACK"
            android:layout_marginRight="11dp"
            android:layout_marginBottom="11dp"
            android:id="@+id/btnBack" />
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textSize="19sp"
            android:id="@+id/txtName"
            />
    </LinearLayout>
```

Step 8: Check Output on Android Emulator and it should look like as given below



Voila!! We have successfully completed this lab.