Lab 19: Android External Storage

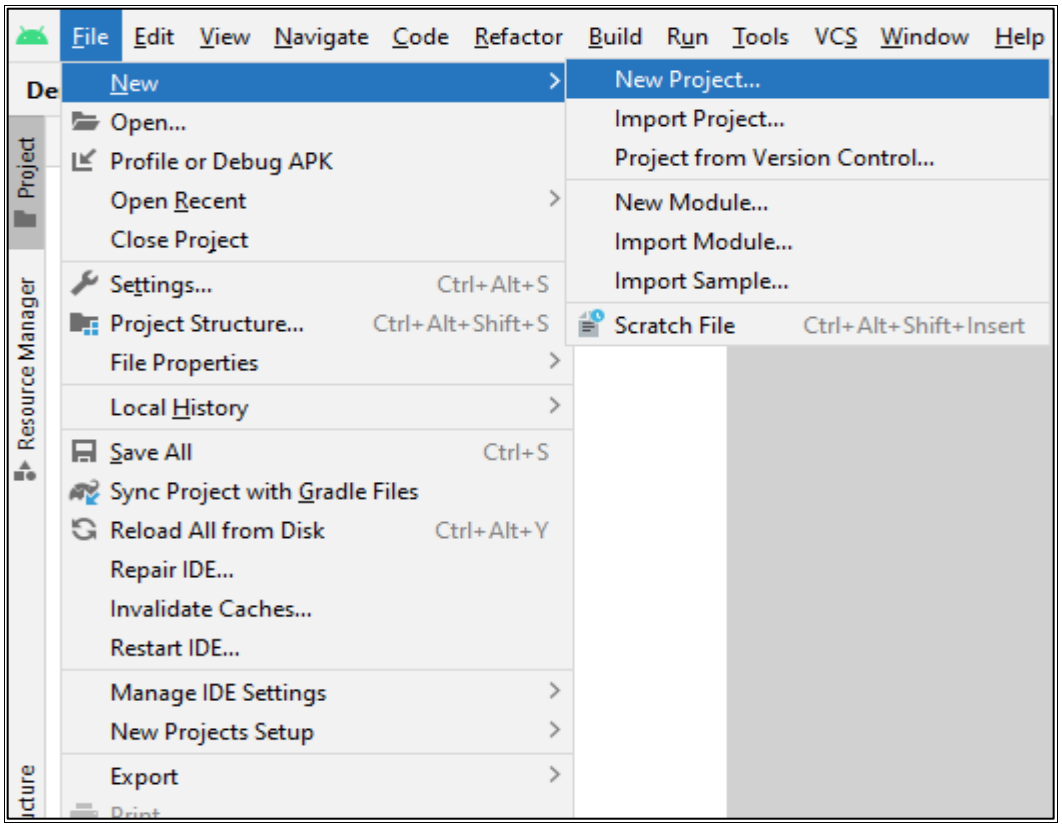
Introduction

We can save or read data from the device's external memory, such as an SD card, just like we can from internal storage. The FileInputStream and FileOutputStream classes can be used to read from and write to files kept on external storage, respectively.

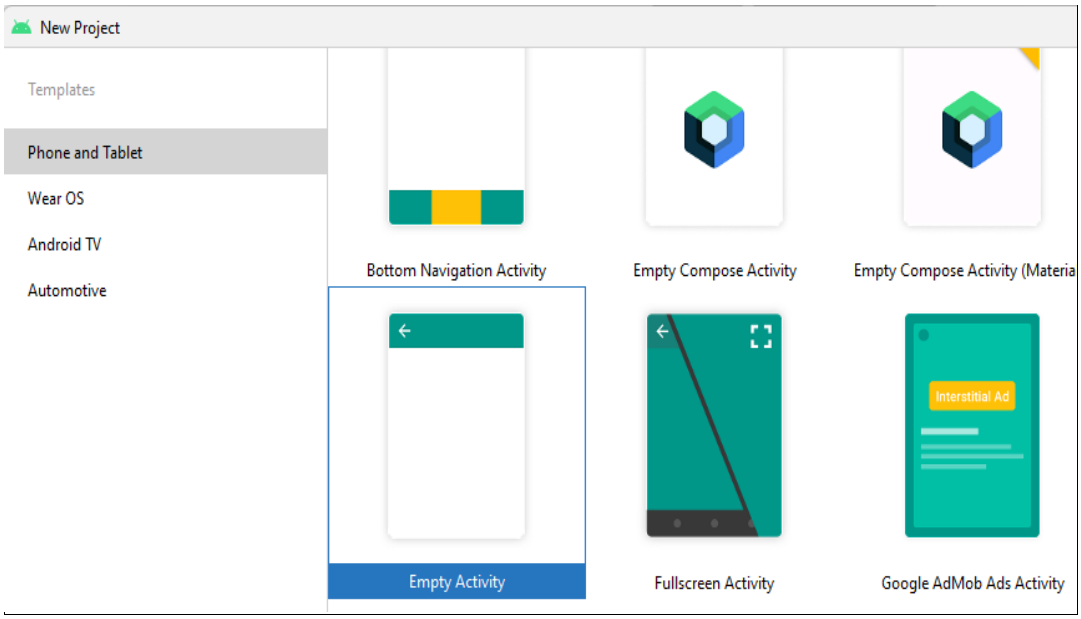
Let’s get Started:

In this experiment we will develop an Android App to demonstrate the use of Android external storage.

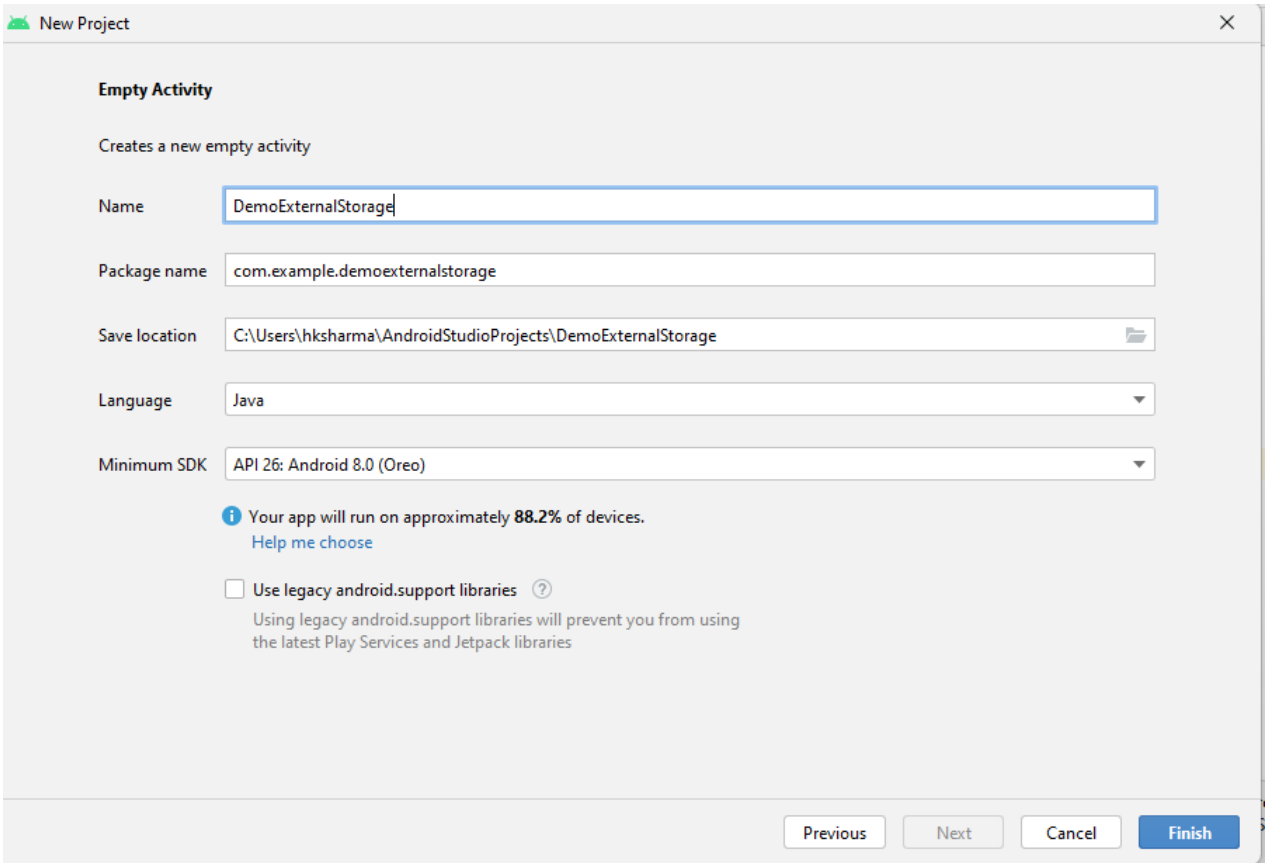
Step 1: Create a New Project in Android Studio as shown below



Step 2: Select Empty Activity as shown below



Step 3: Provide a Project Name as shown below



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

```
package com.example.demoexternalstorage2;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import java.io.BufferedReader;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;

public class MainActivity extends AppCompatActivity {
    EditText editTextFileName,editTextData;
    Button saveButton,readButton;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        editTextFileName=findViewById(R.id.editText1);
        editTextData=findViewById(R.id.editText2);
        saveButton=findViewById(R.id.button1);
        readButton=findViewById(R.id.button2);

        //Performing action on save button
        saveButton.setOnClickListener(new View.OnClickListener(){

            @Override
            public void onClick(View arg0) {
                String filename=editTextFileName.getText().toString();
                String data=editTextData.getText().toString();

                FileOutputStream fos;
                try {
                    File myFile = new File("/sdcard/"+filename);
                    myFile.createNewFile();
                    FileOutputStream fOut = new FileOutputStream(myFile);
                    OutputStreamWriter myOutWriter = new
OutputStreamWriter(fOut);
                    myOutWriter.append(data);
                    myOutWriter.close();
                    fOut.close();
                    Toast.makeText(getApplicationContext(),filename +
"saved",Toast.LENGTH_LONG).show();
                } catch (FileNotFoundException e) {e.printStackTrace();}
                catch (IOException e) {e.printStackTrace();}
```

```
            }
        });

        //Performing action on Read Button
        readButton.setOnClickListener(new View.OnClickListener(){
            @Override
            public void onClick(View arg0) {
                String filename=editTextFileName.getText().toString();
                StringBuffer stringBuffer = new StringBuffer();
                String aDataRow = "";
                String aBuffer = "";
                try {
                    File myFile = new File("/sdcard/"+filename);
                    FileInputStream fIn = new FileInputStream(myFile);
                    BufferedReader myReader = new BufferedReader(
                            new InputStreamReader(fIn));
                    while ((aDataRow = myReader.readLine()) != null) {
                        aBuffer += aDataRow + "\n";
                    }
                    myReader.close();
                } catch (IOException e) {
                    e.printStackTrace();
                }

Toast.makeText(getApplicationContext(),aBuffer,Toast.LENGTH_LONG).show();
            }
        });
    }
}
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    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="External Storage Example"
        android:layout_centerHorizontal="true"
        android:textSize="30dp"
        />

    <EditText
        android:id="@+id/editText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:hint="Enter a File Name "
        android:layout_marginTop="24dp"
        android:ems="10" >
```

```
        <requestFocus />
    </EditText>

    <EditText
        android:id="@+id/editText2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="24dp"
        android:hint="Enter Data to Store"
        android:ems="10" />
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Write Data" />
    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Read Data" />

</LinearLayout>
```

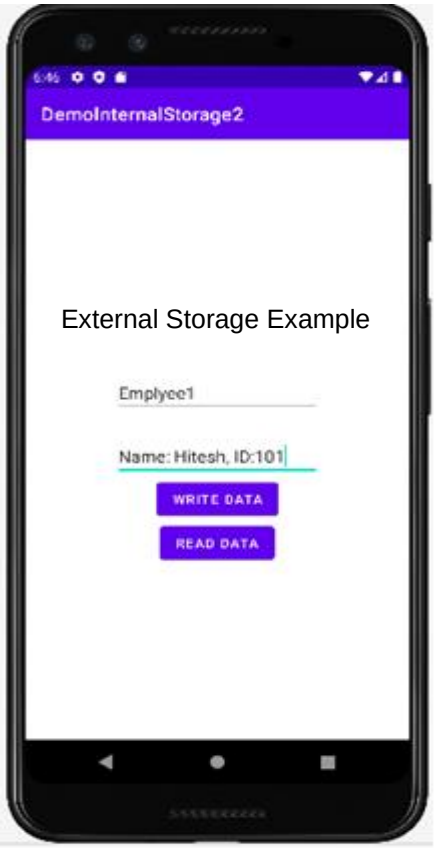
Step 6: Update AndroidManifest.xml as per the code given below

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <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.DemoExternalStorage2"
        tools:targetApi="31">
        <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>

            <meta-data
                android:name="android.app.lib_name"
                android:value="" />
        </activity>
    </application>
</manifest>
```

Step 7: Check Output on Android Emulator and it should look like as given below



Voila!! We have successfully completed this lab.