Lab 20: Android Shared Preferences

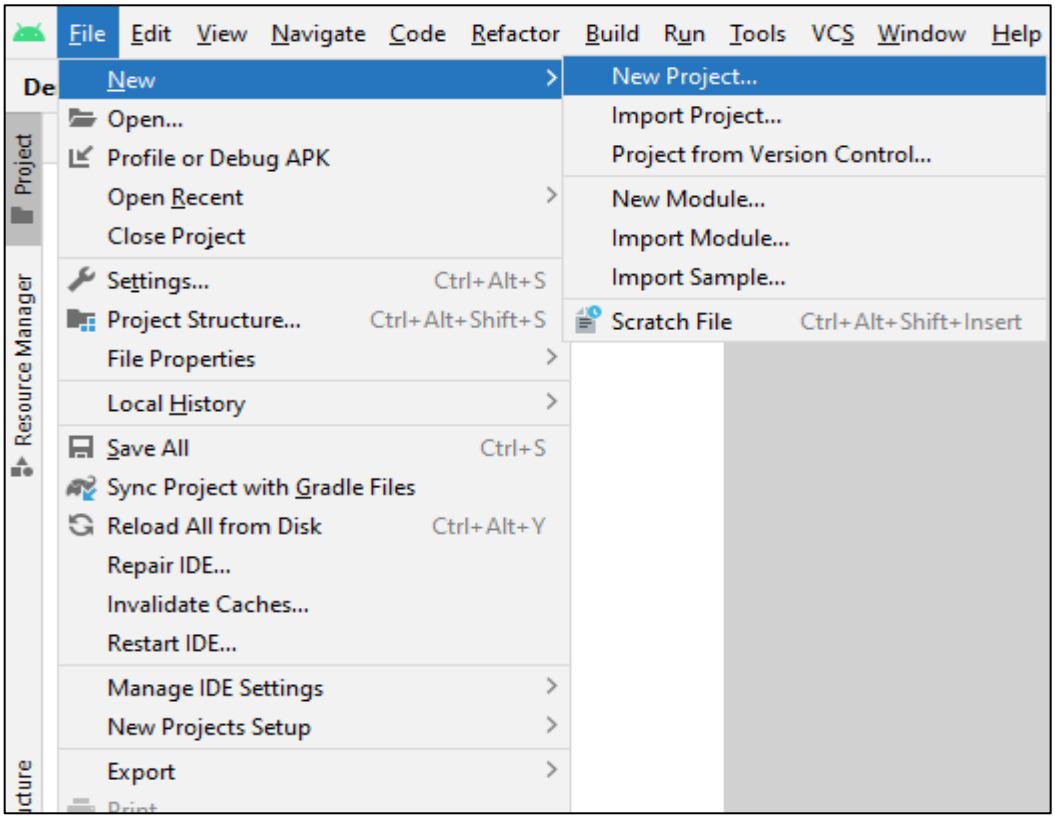
Introduction

Key-value pairs of basic data are stored and retrieved using the Android SharedPreference class. String, integer, long, and number are regarded as primitive data types in Android. We can access the values based on their keys thanks to the key-value pair. It is frequently employed to gather data from users, such as that related to application or device settings.

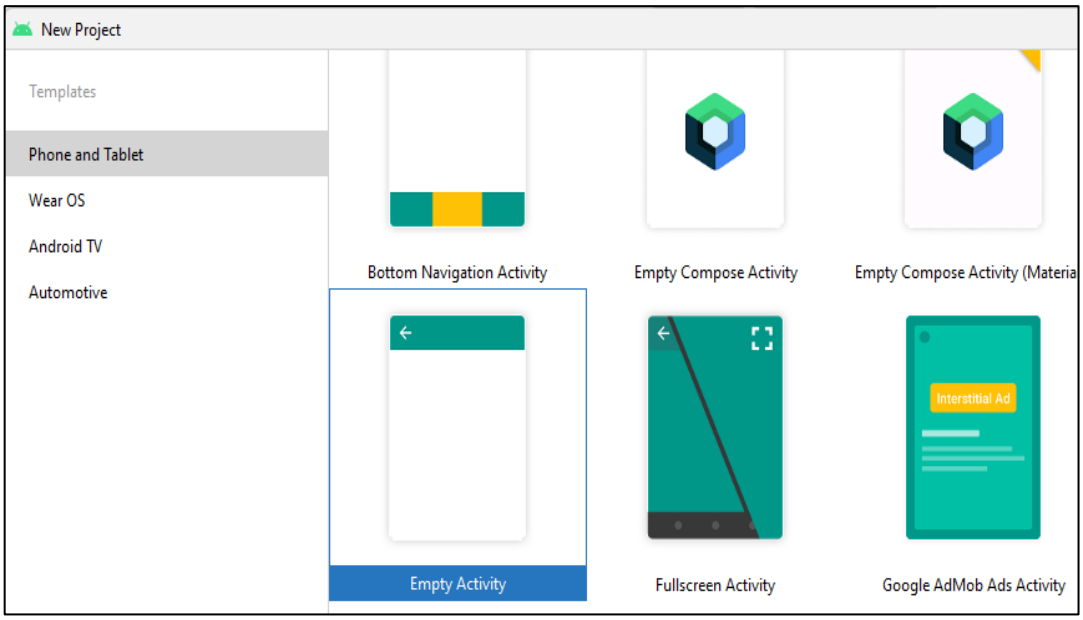
Let's get Started:

In this Exercise we will develop an Android App to demonstrate the use of Android Shared Preferences.

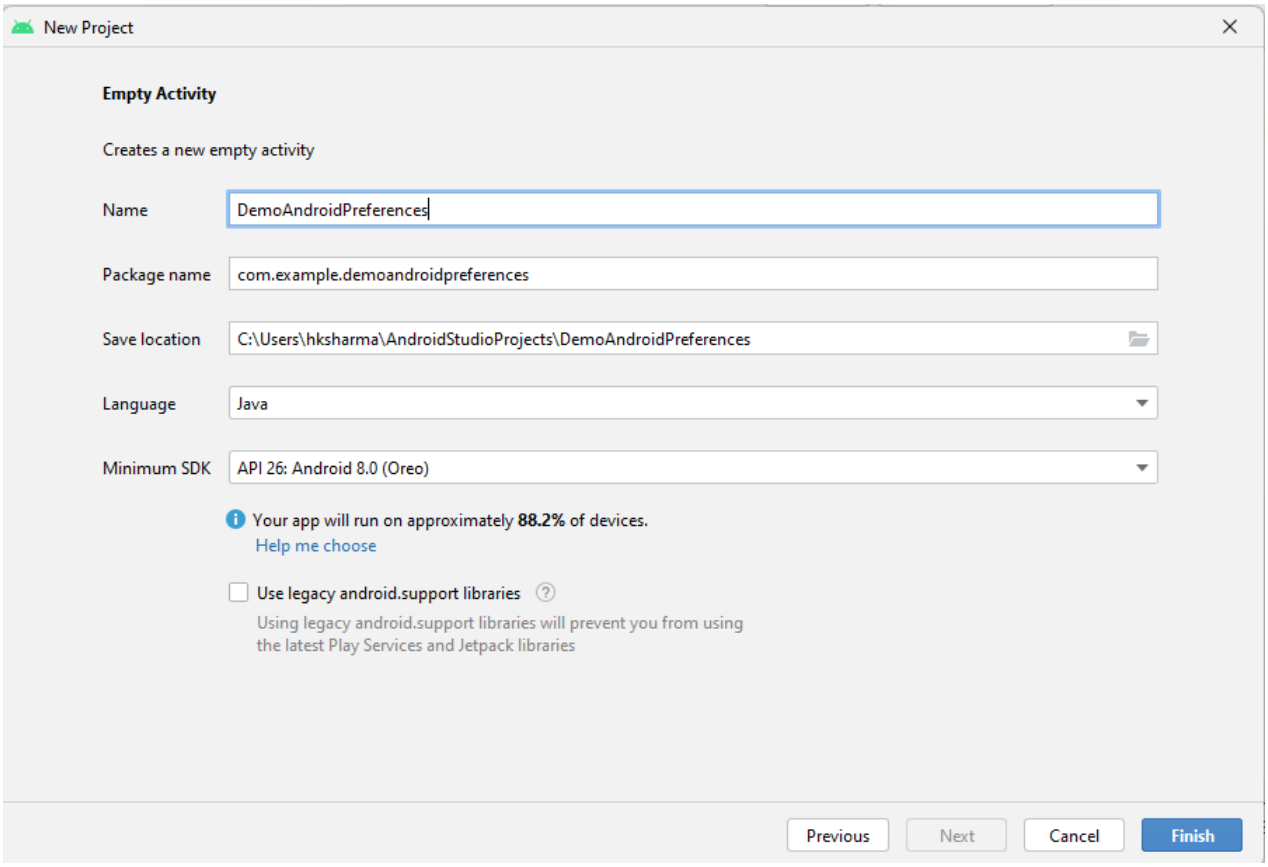
Step 1: Create a New Project in Android Studio as shown below



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.demosharedpref2;
import androidx.appcompat.app.AppCompatActivity;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.widget.EditText;

public class MainActivity extends AppCompatActivity {
    private EditText name, age;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        name = findViewById(R.id.edit1);
        age = findViewById(R.id.edit2);
    }
    @Override
    protected void onResume() {
        super.onResume();
        SharedPreferences sh = getSharedPreferences("MySharedPref",
MODE_PRIVATE);

        String s1 = sh.getString("name", "");
        int a = sh.getInt("age", 0);

        name.setText(s1);
        age.setText(String.valueOf(a));
    }
    @Override
    protected void onPause() {
        super.onPause();
        SharedPreferences sharedPreferences =
getSharedPreferences("MySharedPref", MODE_PRIVATE);
        SharedPreferences.Editor myEdit = sharedPreferences.edit();
        myEdit.putString("name", name.getText().toString());
        myEdit.putInt("age", Integer.parseInt(age.getText().toString()));
        myEdit.apply();
    }
}
```

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

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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"
    tools:context=".MainActivity"
    tools:ignore="HardcodedText">
```

```
    <TextView
        android:id="@+id/textview"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="32dp"
        android:text="Shared Preferences Demo"
        android:textColor="@android:color/black"
        android:textSize="24sp" />

    <!--EditText to take the data from the user
        and save the data in SharedPreferences-->
    <EditText
        android:id="@+id/edit1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textview"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:hint="Enter your Name"
        android:padding="10dp" />

    <!--EditText to take the data from the user and
        save the data in SharedPreferences-->
    <EditText
        android:id="@+id/edit2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/edit1"
        android:layout_marginStart="16dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="16dp"
        android:hint="Enter your Age"
        android:padding="10dp"
        android:inputType="number" />

</RelativeLayout>
```

Step 6: Check Output on Android Emulator and it should look like as given below

Voila!! We have successfully completed this lab.