**Lab 13: Android Vertical ScrollView**

# **Introduction**

The **android.widget.ScrollView** class provides the functionality of scroll view. ScrollView is used to scroll the child elements of the palette inside ScrollView. the default scroll view is vertical scroll view to scroll elements vertically.you may use *HorizontalScrollView* for horizontal ScrollView.

**Let’s get Started**

Here is an exercise demonstrating the development of the following vertical ScrollView.

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

Graphical user interface, text, application

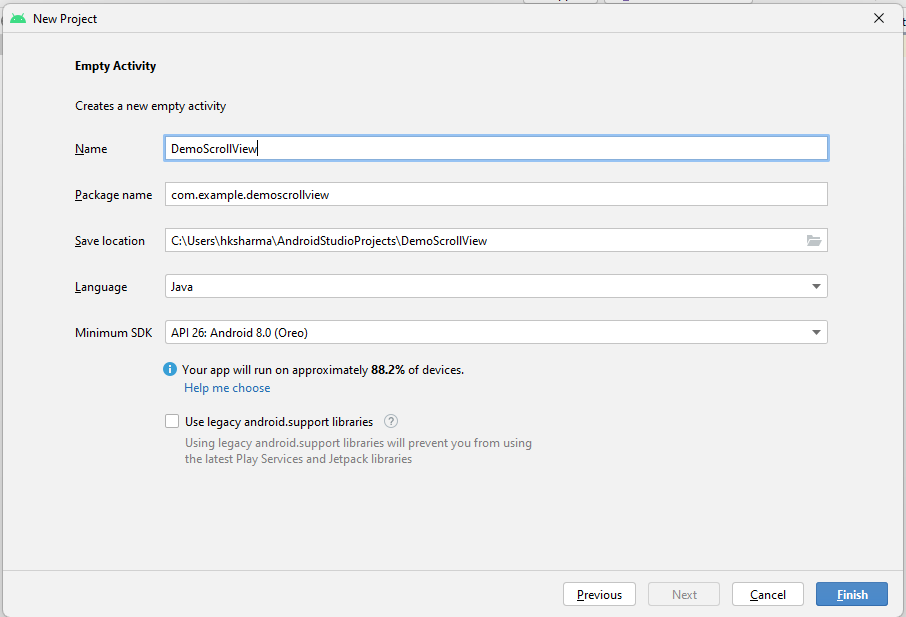
Description automatically generated

**Step 2: Select Empty Activity as shown below**

Graphical user interface, application, shape

Description automatically generated

**Step 3: Provide a Project Name as shown below**

****

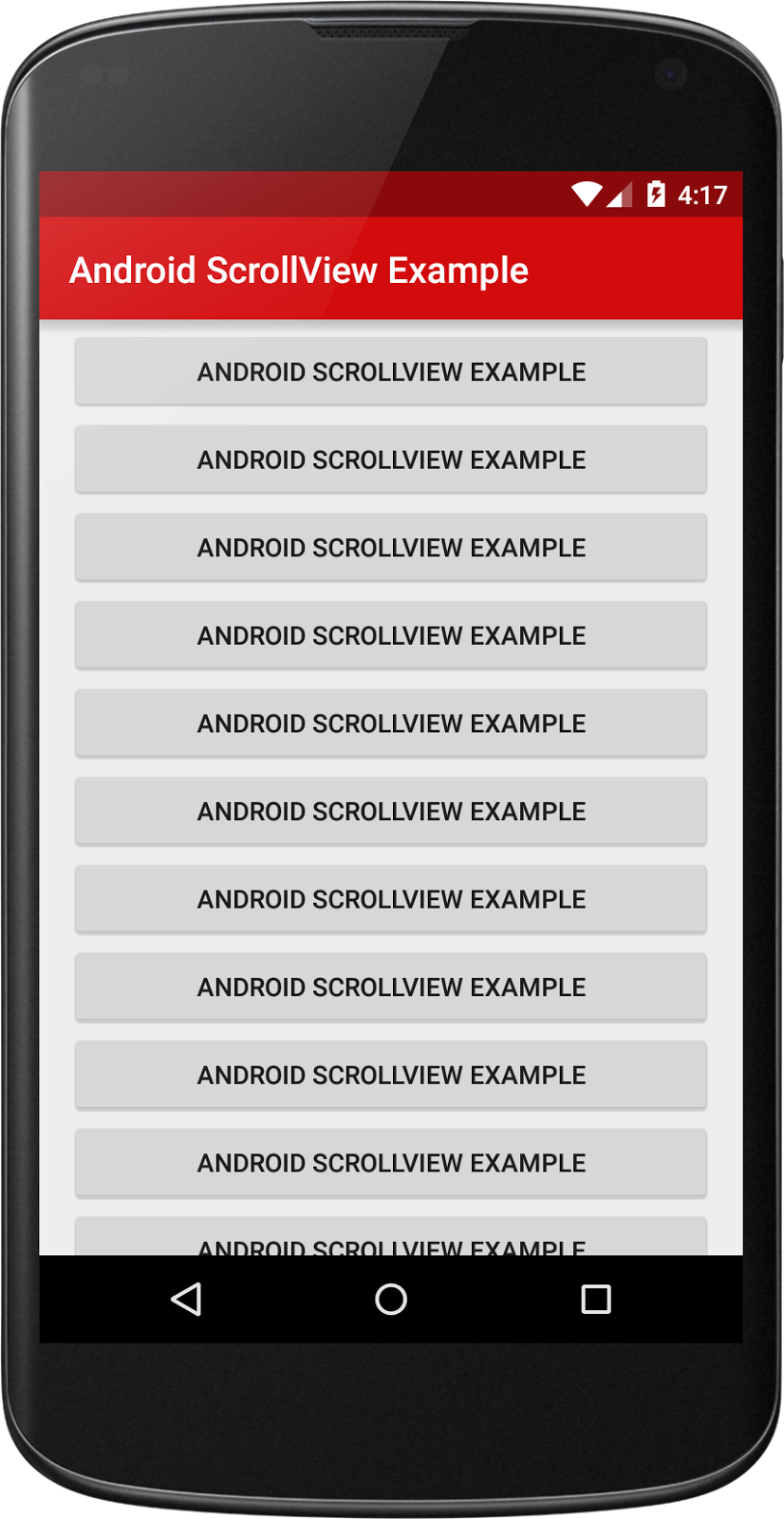
**Step 4: Update MainActivity.java as per the code given below**

**package** com.example.demoscrollview;  
**import** androidx.appcompat.app.AppCompatActivity;  
**import** android.os.Bundle;  
**public class** MainActivity **extends** AppCompatActivity {  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 }  
}

**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"**>  
   
 <**TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textAppearance="?android:attr/textAppearanceMedium"  
 android:text="Android ScrollView Example"  
 android:id="@+id/textView"  
 android:layout\_gravity="center\_horizontal"  
 android:layout\_centerHorizontal="true"  
 android:layout\_alignParentTop="true"** />  
  
 <**ScrollView android:layout\_marginTop="30dp"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/scrollView"**>  
  
 <**LinearLayout  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:orientation="vertical"** >  
  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 <**Button  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="Android ScrollView Example"** />  
 </**LinearLayout**>  
 </**ScrollView**>  
  
</**RelativeLayout**>

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

****

**Voila!!** We have successfully completed this lab.