Ex. No: 10

ALARM CLOCK

AIM

To develop an Android Application that creates an Alarm Clock.

PROCEDURE

1. Creating a New project:
   - Open Android Studio and then click on File -> New -> New project.
   - Then type the Application name and click Next.Then select the Minimum SDK as shown below and click Next.
   - Then select the Empty Activity and click Next.Finally click Finish.
2. Creating Second Activity for the Android Application:
   - Click on File -> New -> Activity -> Empty Activity.
   - Type the Activity Name as AlarmReceiver and click Finish button.
3. Designing layout for the Android Application:
   - Click on app -> res -> layout -> activity_main.xml.
   - Create Relative layout with Toggle button and Time Picker
4. Changes in Manifest for the Android Application:
   - Click on app -> manifests -> AndroidManifest.xml
   - Now change the activity tag to receiver tag in the AndroidManifest.xml file.

SOURCE CODE

activity_main.xml

```
<?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"
   tools:context=".MainActivity">

   <!-- Added Time Picker just to pick alarm time -->
   <TimePicker
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_gravity="center"
      android:id="@+id/timePicker"/>

   <!-- Added Toggle Button to set the alarm on or off-->
   <ToggleButton
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:id="@+id/toggleButton"
      android:layout_gravity="center"
```

```
        android:layout_margin="20dp"
        android:checked="false"
        android:onClick="OnToggleClicked" />
</LinearLayout>
```

MainActivity.java

```
package com.ex.alarm;

import androidx.appcompat.app.AppCompatActivity;

import android.app.AlarmManager;
import android.app.PendingIntent;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.TimePicker;
import android.widget.Toast;
import android.widget.ToggleButton;

import java.util.Calendar;

public class MainActivity extends AppCompatActivity {
    TimePicker alarmTimePicker;
    PendingIntent pendingIntent;
    AlarmManager alarmManager;

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

        alarmTimePicker = findViewById(R.id.timePicker);
        alarmManager = (AlarmManager) getSystemService(ALARM_SERVICE);


    }

    public void OnToggleClicked(View view){
        long time;
        if(((ToggleButton)view).isChecked()){
            Toast.makeText(MainActivity.this,"ALARM ON",Toast.LENGTH_SHORT).show();
            Calendar calendar = Calendar.getInstance();

            // calendar is called to get current time in hour and minute
            calendar.set(Calendar.HOUR_OF_DAY,alarmTimePicker.getCurrentHour());
```

```
        calendar.set(Calendar.MINUTE,alarmTimePicker.getCurrentMinute());

        // using intent i have class AlarmReceiver class which inherits BroadCastReceiver

        Intent intent = new Intent(this,AlarmReceiver.class);

        // we call broadcast using pendingIntent

        pendingIntent = PendingIntent.getBroadcast(this,0,intent,0);

        time = (calendar.getTimeInMillis() - (calendar.getTimeInMillis() % 60000));
        if(System.currentTimeMillis() > time ){
           if(Calendar.AM_PM == 0)
              time = time + (1000*60*60*12);
           else
              time = time + (1000*60*60*24);
        }
        // Alarm rings continuously until toggle button button is turned off
        alarmManager.setRepeating(AlarmManager.RTC_WAKEUP,time,10000,pendingIntent);

      }
      else{
        alarmManager.cancel(pendingIntent);
        Toast.makeText(MainActivity.this,"ALARM OFF",Toast.LENGTH_SHORT).show();
      }
   }
}
```

AlarmReceiver.java

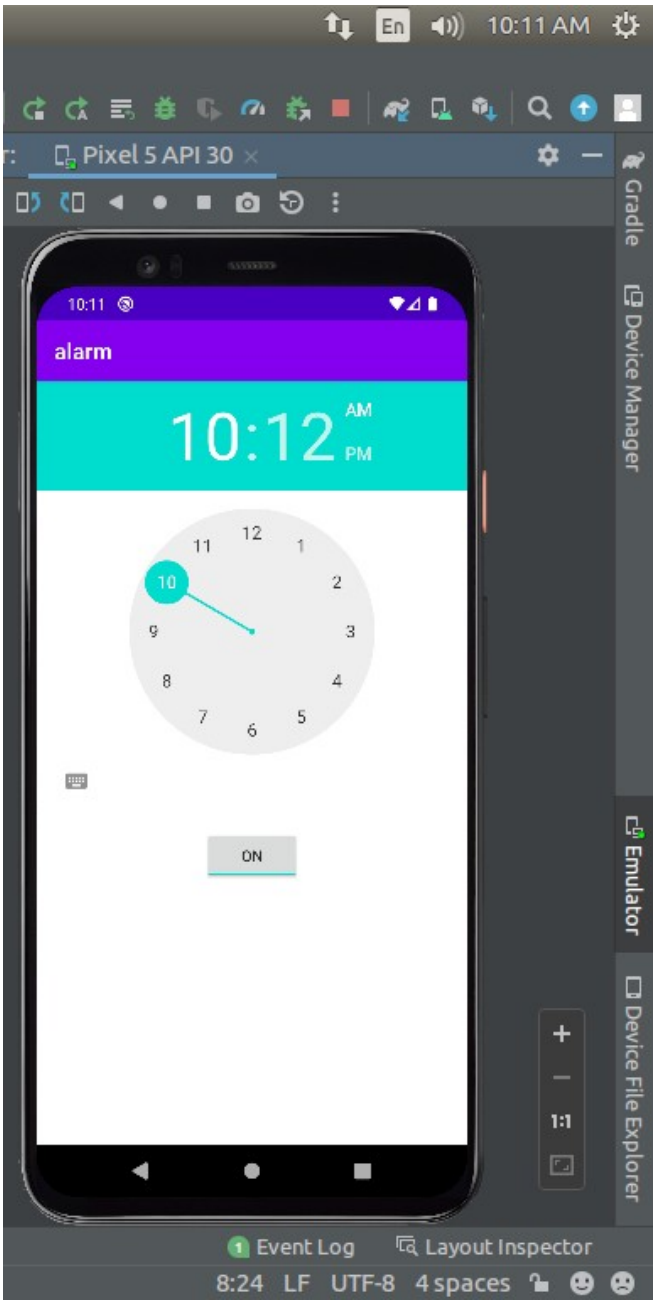
```
package com.ex.alarm;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.media.Ringtone;
import android.media.RingtoneManager;
import android.net.Uri;
import android.os.Vibrator;
import android.widget.Toast;

public class AlarmReceiver extends BroadcastReceiver {

   //implement OnReceive Method

   @Override
   public void onReceive(Context context, Intent intent) {
```

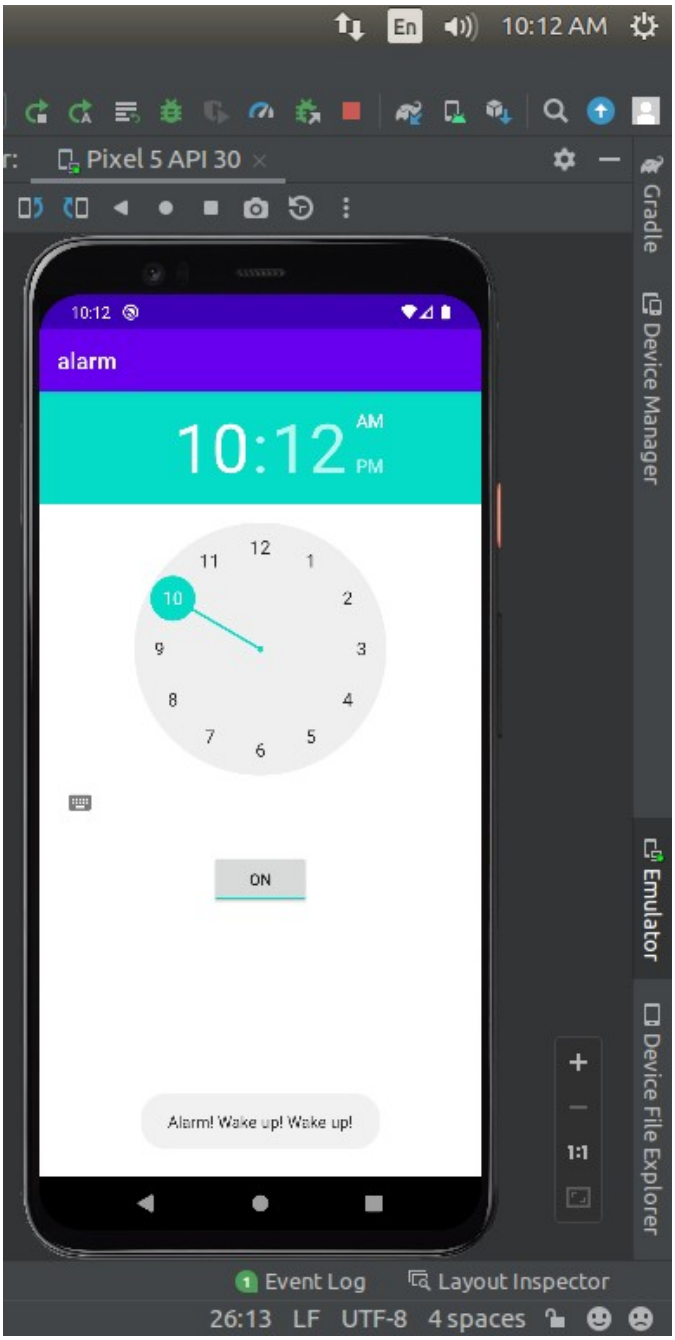
```
        // we will use vibrator first
        Vibrator vibrator = (Vibrator) context.getSystemService(Context.VIBRATOR_SERVICE);
        vibrator.vibrate(4000);

        Toast.makeText(context,"Alarm! Wake up! Wake up!",Toast.LENGTH_SHORT).show();
        Uri alarmUri = RingtoneManager.getDefaultUri(RingtoneManager.TYPE_ALARM);
        if(alarmUri == null){
            alarmUri = RingtoneManager.getDefaultUri(RingtoneManager.TYPE_NOTIFICATION);
        }

        // setting default ringtone
        Ringtone ringtone = RingtoneManager.getRingtone(context,alarmUri);
        ringtone.play();
    }
}
```

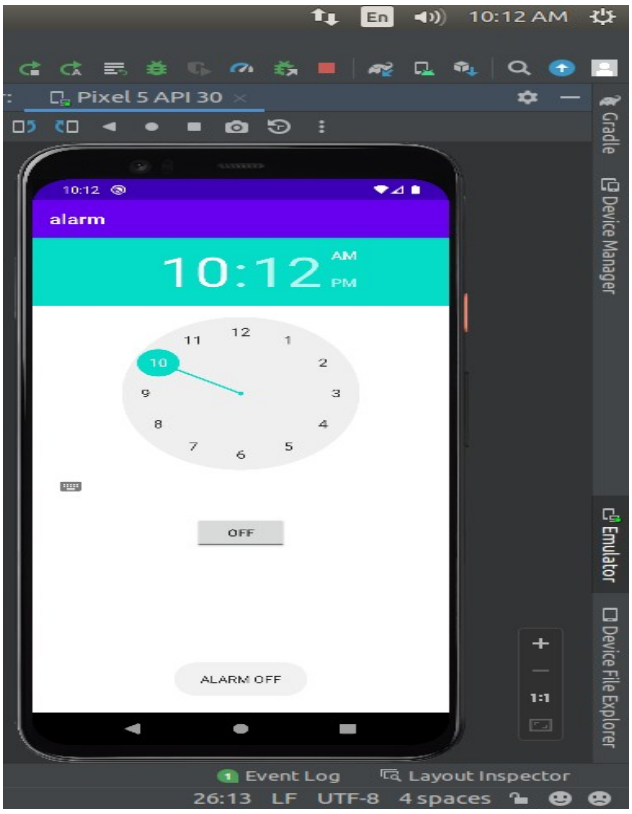
Manifest.xml

```
BEFORE TO  <application TYPE THE FOLLOWING LINE:
    <uses-permission android:name="android.permission.VIBRATE" />

        AFTER TO  </activity> TYPE THE FOLLOWING LINE:
    <receiver android:name=".AlarmReceiver">
    </receiver>
    </application>

</manifest>
```

OUTPUT:


En
10:11 AM
Pixel 5 API 30
Gradle
Device Manager
10:11
alarm
10:12
AM
PM
12
1
2
3
4
5
6
7
8
9
10
11
ON
Emulator
Device File Explorer
1:1
Event Log
Layout Inspector
8:24
LF
UTF-8
4 spaces


En
10:12 AM
Pixel 5 API 30
Gradle
Device Manager
10:12
alarm
10:12
AM
PM
12
1
2
3
4
5
6
7
8
9
10
11
ON
Alarm! Wake up! Wake up!
1:1
Emulator
Device File Explorer
Event Log
Layout Inspector
26:13
LF
UTF-8
4 spaces



RESULT

Thus an Android Application that creates an Alarm Clock is implemented successfully.

Ex. No: 11

NATIVE CALCULATOR

AIM

To develop an Android Application for Native Calculator.

PROCEDURE

1 .Creating a New project:

- Open Android Studio and then click on File -> New -> New project.
- Then type the Application name and click Next.
- Then select the Minimum SDK as shown below and click Next.
- Then select the Empty Activity and click Next.
- Finally click Finish.

2. Designing layout for the Android Application:

- Click on app -> res -> layout -> activity_main.xml
- Create Linear layout with two edit text
- Create another Linear Layout for four buttons add ,sub, multiply and divide

3. Java Coding for the Android Application

- Click on app -> java -> MainActivity.
- Declare edit texts for getting two numbers
- Declare four buttons for addition, subtraction, Multiplication and division operation.
- Buffer the views and set the listeners
- Dheck if the fields are empty and read EditText and fill variables with numbers

Define the button that has been clicked and performs the corresponding operation using switch case to perform particular operation.

SOURCE CODE

MainActivity.java

```
package com.ex.calc;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
public class MainActivity extends AppCompatActivity {
   Button b1,b2,b3,b4,b5,b6,b7,b8,b9,b0,bp,bm,bmu,bdo,bd,be,bclr;
   EditText tx;
   float val1=0,val2=0,res=0;
   String op="";
   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
```

```
    setContentView(R.layout.activity_main);
    b1=(Button)findViewById(R.id.one);
    b2=(Button)findViewById(R.id.two);
    b3=(Button)findViewById(R.id.three);
    b4 =(Button)findViewById(R.id.four);
    b5=(Button)findViewById(R.id.five);
    b6=(Button)findViewById(R.id.six);
    b7=(Button)findViewById(R.id.seven);
    b8=(Button)findViewById(R.id.eight);
    b9=(Button)findViewById(R.id.nine);
    b0=(Button)findViewById(R.id.zero);
    bp=(Button)findViewById(R.id.plus);
    bm=(Button)findViewById(R.id.minus);
    bmu=(Button)findViewById(R.id.multi);
    bdo=(Button)findViewById(R.id.dot);
    bd=(Button)findViewById(R.id.divide);
    be=(Button)findViewById(R.id.equal);
    bclr=(Button)findViewById(R.id.clear);
    tx=(EditText)findViewById(R.id.txt);
}
public void show(View v){
    Button b=(Button)v;
    String set=b.getText().toString();
    tx.append(set);
}
public void calc(View v){
    Button b=(Button)v;
    op=b.getText().toString();
    val1=Float.parseFloat(tx.getText().toString());
    tx.setText("");
}
public void equal(View v){
    val2=Float.parseFloat(tx.getText().toString());
    switch(op) {
        case "+":
            res =val1+val2;
            tx.setText(String.valueOf(res));
            break;
        case "-":
            res =val1-val2;
            tx.setText(String.valueOf(res));
            break;
        case "*":
            res =val1*val2;
            tx.setText(String.valueOf(res));
```

```
            break;
        case "/":
            res =val1/val2;
            tx.setText(String.valueOf(res));
            break;
        default:

    }
  }
}
```

activity_main.xml

```
<?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:paddingLeft="5dp"
  android:paddingRight="5dp"
  android:paddingTop="5dp"
  android:orientation="vertical"
  android:paddingBottom="5dp"
  tools:context=".MainActivity">
  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
    <EditText
      android:layout_width="fill_parent"
      android:layout_height="50dp"
      android:layout_marginLeft="25dp"
      android:gravity="right"
      android:focusableInTouchMode="true"
      android:id="@+id/txt"
      android:layout_marginRight="25dp"
      tools:ignore="RtlHardcoded" />
  </LinearLayout>
  <LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:layout_marginTop="20dp"
    android:orientation="horizontal">
    <Button
      android:layout_width="75dp"
      android:layout_height="75dp"
```

```
        android:text="9"
        android:onClick="show"
        android:id="@+id/nine"
        android:textSize="30dp"/>
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:onClick="show"
        android:id="@+id/eight"
        android:text="8"
        android:textSize="30dp"/>
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="7"
        android:onClick="show"
        android:id="@+id/seven"
        android:textSize="30dp"/>

    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:onClick="calc"
        android:id="@+id/plus"
        android:text="+"
        android:textSize="30dp"/>
</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:orientation="horizontal">
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:onClick="show"
        android:id="@+id/six"
        android:text="6"
        android:textSize="30dp"/>
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="5"
        android:onClick="show"
        android:id="@+id/five"
```

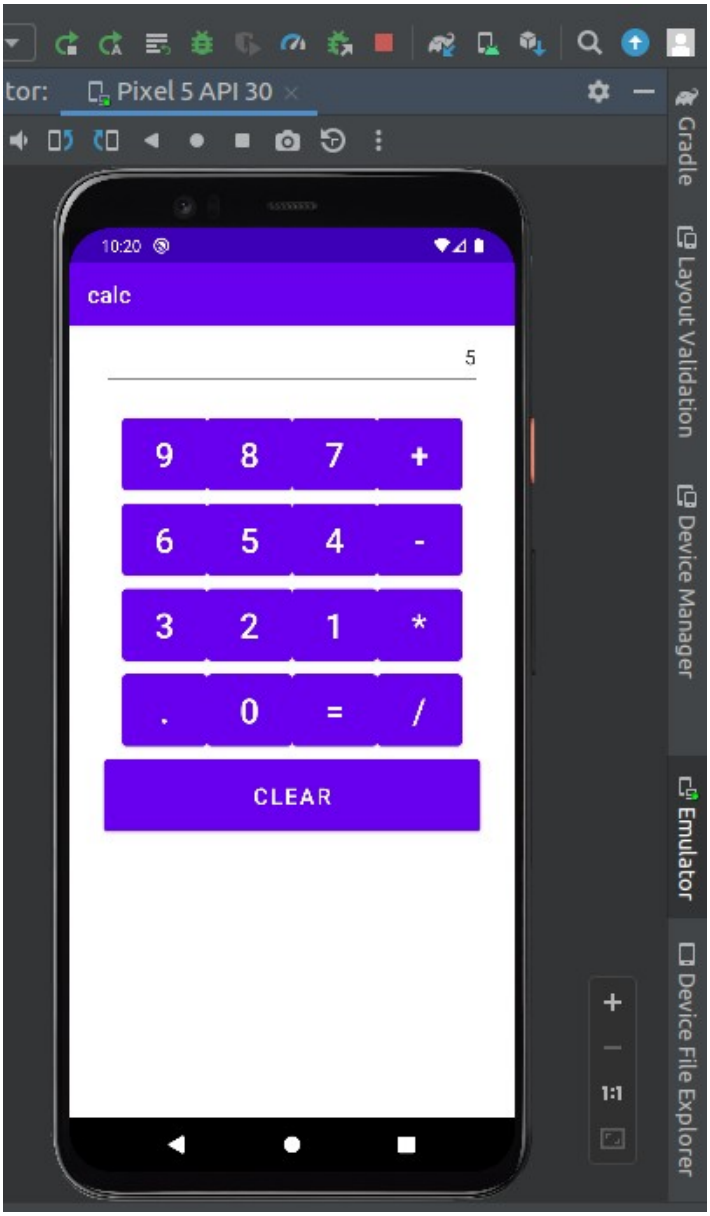
```
        android:textSize="30dp"/>
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="4"
        android:onClick="show"
        android:id="@+id/four"
        android:textSize="30dp"/>
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="-"
        android:onClick="calc"
        android:id="@+id/minus"
        android:textSize="30dp"/>
</LinearLayout>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="center"
    android:orientation="horizontal">
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="3"
        android:onClick="show"
        android:id="@+id/three"
        android:textSize="30dp"/>
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="2"
        android:onClick="show"
        android:id="@+id/two"
        android:textSize="30sp"/>
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="1"
        android:onClick="show"
        android:id="@+id/one"
        android:textSize="30sp"/>
    <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
```

```
        android:text="*"
        android:onClick="calc"
        android:id="@+id/multi"
        android:textSize="30sp"/>
    </LinearLayout>
    <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:gravity="center"
      android:orientation="horizontal">
      <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:onClick="show"
        android:id="@+id/dot"
        android:text="."
        android:textSize="30sp"/>
      <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="0"
        android:onClick="show"
        android:id="@+id/zero"
        android:textSize="30sp"/>
      <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="="
        android:onClick="equal"
        android:id="@+id/equal"
        android:textSize="30sp"/>
      <Button
        android:layout_width="75dp"
        android:layout_height="75dp"
        android:text="/"
        android:onClick="calc"
        android:id="@+id/divide"
        android:textSize="30sp"/>
    </LinearLayout>
    <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content">
      <Button
        android:layout_width="fill_parent"
        android:layout_height="75dp"
```

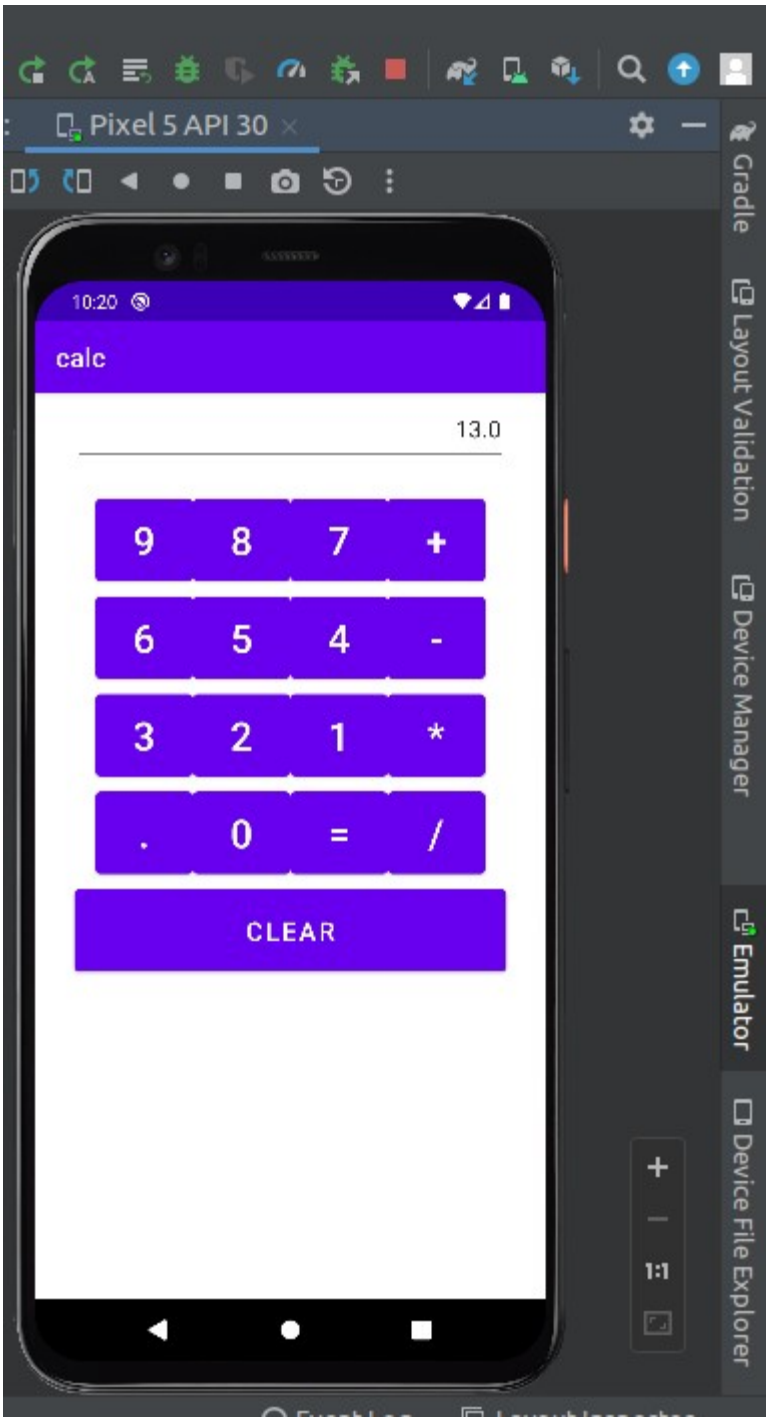
```
        android:text="CLEAR"
        android:onClick="calc"
        android:id="@+id/clear"
        android:layout_marginLeft="25dp"
        android:layout_marginRight="25dp"
        android:textSize="20sp"/>
    </LinearLayout>
</LinearLayout>
```

OUTPUT


Pixel 5 API 30
10:20
calc
65
9
8
7
+
6
5
4
-
3
2
1
*
.
0
=
/
CLEAR



Click “/” first and then type 5.



Output is displayed after clicking "=" : 65/5=13. Click "CLEAR" to do the next calculation.

RESULT

Thus an Android Application for Native Calculator has been implemented successfully.