**EX NO: 07 Implement an application that implements Multi-threading.**

**DATE: 06/09/21**

**AIM:**

To Implement an application that implements Multi-threading.

**SOURCE CODE:**

**Activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical" >

<ImageView

android:id="@+id/imageView"

android:layout\_width="250dp"

android:layout\_height="250dp"

android:layout\_margin="50dp"

android:layout\_gravity="center" />

<Button

android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:layout\_gravity="center"

android:text="Load Image 1" />

<Button

android:id="@+id/button2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_margin="10dp"

android:layout\_gravity="center"

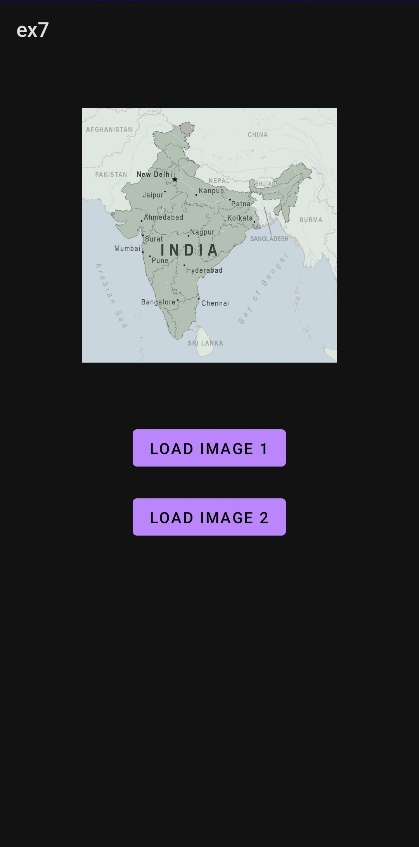
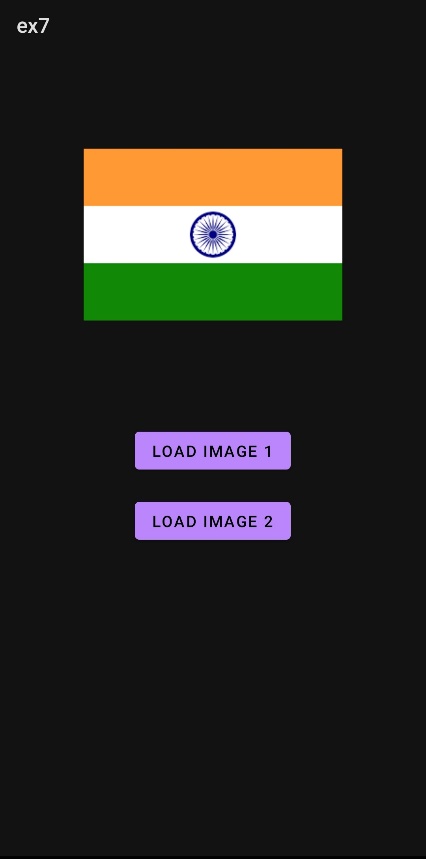
android:text="Load image 2" />

</LinearLayout>

**MainActivity.java**

package com.example.ex7;  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.ImageView;  
public class MainActivity extends AppCompatActivity  
{  
 ImageView img;  
 Button bt1,bt2;  
 @Override  
 protected void onCreate(Bundle savedInstanceState)  
 {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 bt1 = (Button)findViewById(R.id.*button*);  
 bt2= (Button) findViewById(R.id.*button2*);  
 img = (ImageView)findViewById(R.id.*imageView*);  
  
 bt1.setOnClickListener(new View.OnClickListener()  
 {  
 @Override  
 public void onClick(View v)  
 {  
 new Thread(new Runnable()  
 {  
 @Override  
 public void run()  
 {  
 img.post(new Runnable()  
 {  
 @Override  
 public void run()  
 {  
 img.setImageResource(R.drawable.*india1*);  
 }  
 });  
 }  
 }).start();  
 } });  
  
 bt2.setOnClickListener(new View.OnClickListener()  
 {  
 @Override  
 public void onClick(View v)  
 {  
 new Thread(new Runnable()  
 {  
 @Override  
 public void run()  
 {  
 img.post(new Runnable()  
 {  
 @Override  
 public void run()  
 {  
 img.setImageResource(R.drawable.*india2*);  
 }  
 }); }  
 }).start();  
 }  
 });  
 }}

**OUTPUT:**

**RESULT:**

Thus an application that implements Multi-threading has been developed and executed.

**EX NO: 08 Develop a native application that uses GPS location information**

**DATE: 13/09/21**

**AIM:**

To develop a native application that uses GPS location information.

**SOURCE CODE:**

**Activity\_main.xml:**

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout  
 xmlns:android="http://schemas.android.com/apk/res/android" android:id="@+id/relativeLayout1" android:layout\_width="match\_parent" android:layout\_height="match\_parent">  
  
 <Button  
  
 android:id="@+id/show\_Location"  
  
 android:layout\_width="wrap\_content"  
  
 android:layout\_height="wrap\_content"  
  
 android:text="Show\_Location" android:layout\_centerVertical= "true"  
  
 android:layout\_centerHorizontal ="true"  
 />  
</RelativeLayout>

**MainActivity.java:**

package com.example.ex8;  
  
import android.app.Activity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.Toast;  
  
public class MainActivity extends Activity {  
 */\*\* Called when the activity is first created. \*/* Button btnShowLocation;  
 GPStrace gps;  
 @Override  
 public void onCreate (Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 btnShowLocation = (Button) findViewById(R.id.*show\_Location*);  
 btnShowLocation.setOnClickListener(v -> {  
  
 // *TODO Auto-generated method stub* gps = new GPStrace(MainActivity.this);  
 if (gps.canGetLocation()) {  
 double latitude = gps.getLatitude();  
  
 double longitude = gps.getLongitude();  
  
 Toast.*makeText*(getApplicationContext(), "Your Location is \nLat: " + latitude + "\nLong: " + longitude, Toast.*LENGTH\_LONG*).show();  
 } else {  
 gps.showSettingAlert();  
 }  
 });  
 }  
}

**GPSTrace.java**

package com.example.ex8;  
  
import android.Manifest;  
import android.app.AlertDialog;  
import android.app.Service;  
import android.content.Context;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.content.pm.PackageManager;  
import android.location.Location;  
import android.location.LocationListener;  
import android.location.LocationManager;  
import android.os.Bundle;  
import android.os.IBinder;  
import android.provider.Settings;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.app.ActivityCompat;  
  
public class GPStrace extends Service implements LocationListener {  
  
 private final Context context;  
 boolean isGPSEnabled = false;  
  
 boolean canGetLocation = false;  
 boolean isNetworkEnabled = false;  
 Location location;  
 double latitude;  
 double longitude;  
 private static final long  
  
 *MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES* = 10;  
 private static final long *MIN\_TIME\_BW\_UPDATES* = 1000 \* 60 \* 1;  
  
 protected LocationManager locationManager;  
 private Location TODO;  
 public GPStrace(Context context) {  
 this.context = context;  
 getLocation();  
 }  
 public Location getLocation() {  
 try {  
 locationManager = (LocationManager)  
 context.getSystemService(*LOCATION\_SERVICE*);  
 isGPSEnabled = locationManager.isProviderEnabled(LocationManager.*GPS\_PROVIDER*);  
  
 isNetworkEnabled = locationManager.isProviderEnabled(LocationManager.*NETWORK\_PROVIDER*);  
 if (!isGPSEnabled && !isNetworkEnabled) {  
  
 } else {  
 this.canGetLocation = true;  
 if (isNetworkEnabled) {  
 if (ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_FINE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED* && ActivityCompat.*checkSelfPermission*(this, Manifest.permission.*ACCESS\_COARSE\_LOCATION*) != PackageManager.*PERMISSION\_GRANTED*) {  
 // *TODO: Consider calling* // *ActivityCompat#requestPermissions* // here to request the missing permissions, and then overriding  
 // public void onRequestPermissionsResult(int requestCode, String[] permissions,  
 // int[] grantResults)  
 // to handle the case where the user grants the permission. See the documentation  
 // for ActivityCompat#requestPermissions for more details.  
 return TODO;  
 }  
 locationManager.requestLocationUpdates(LocationManager.*NETWORK\_PROVIDER*, *MIN\_TIME\_BW\_UPDATES*, *MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES*, this);  
 }  
 if (locationManager != null) {  
 location = locationManager.getLastKnownLocation(LocationManager.*NETWORK\_PROVIDER*); if (location != null) {  
 latitude = location.getLatitude();  
 longitude = location.getLongitude();  
 }  
 }  
 }  
 if (isGPSEnabled) {  
  
 if (location == null) {  
 locationManager.requestLocationUpdates(LocationManager.*GPS\_PROVIDER*, *MIN\_TIME\_BW\_UPDATES*, *MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES*, this);  
 if (location != null) {  
 latitude = location.getLatitude();  
 longitude = location.getLongitude();  
 }  
 }  
 }  
 }  
 catch (Exception e) {  
 e.printStackTrace();  
 }  
 return location;  
 }  
 public void stopUsingGPS() {  
  
 if (locationManager != null) {  
  
 locationManager.removeUpdates(GPStrace.this);  
 }  
 }  
 public double getLatitude() {  
 if (location != null) {  
  
 latitude = location.getLatitude();  
 }  
 return latitude;  
 }  
 public double getLongitude() {  
 if (location != null) {  
  
 longitude = location.getLatitude();  
 }  
 return longitude;  
 }  
 public boolean canGetLocation() {  
  
 return this.canGetLocation;  
 }  
 public void showSettingAlert () {  
  
 AlertDialog.Builder alertDialog = new AlertDialog.Builder(context);  
  
 alertDialog.setTitle("GPS is settings");  
 alertDialog.setMessage("GPS is not enabled.Do you want to go to setting menu?");  
 alertDialog.setPositiveButton("settings", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialogInterface, int which) {  
  
 Intent intent = new Intent(Settings.*ACTION\_LOCATION\_SOURCE\_SETTINGS*);  
 context.startActivity(intent);  
 }  
 });  
 alertDialog.setNegativeButton("cancel", new DialogInterface.OnClickListener() {  
 @Override  
 public void onClick(DialogInterface dialogInterface, int which) {  
 // *TODO Auto-generated method stub* DialogInterface dialog = null;  
 dialog.cancel();  
 }  
 });  
 alertDialog.show();  
 }  
 @Override  
 public void onLocationChanged (Location Location)  
  
 {  
// *TODO Auto-generated method stub* } @Override  
  
 public void onProviderDisabled (String provider)  
 {// *TODO Auto-generated method stub* }  
 @Override  
  
 public void onProviderEnabled (String provider){  
  
// *TODO Auto-generated method stub* }  
 @Override  
  
 public void onStatusChanged (String provider,int status, Bundle extras){  
  
// *TODO Auto-generated method stub* }  
 @Override  
  
 public IBinder onBind (Intent intent){  
  
// *TODO Auto-generated method stub return null;* return null;  
 }  
}

**OUTPUT:**



**RESULT:**

Thus a native application that uses GPS location information has been developed and executed.

**EX NO: 09 Develop a Android Application that writes data to the SD Card.**

**DATE: 24/10/21**

**AIM:**

To develop a Android Application that writes data to the SD Card.

**SOURCE CODE:**

**MainActivity.java:**

package com.example.ex09;  
  
import android.os.Bundle;  
import androidx.appcompat.app.AppCompatActivity;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
import java.io.BufferedReader;  
import java.io.File;  
import java.io.FileInputStream;  
import java.io.FileOutputStream;  
import java.io.InputStreamReader;  
  
public class MainActivity extends AppCompatActivity  
{  
 EditText e1;  
 Button write,read,clear;  
 @Override  
 protected void onCreate(Bundle savedInstanceState){  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 e1= (EditText) findViewById(R.id.*editText*);  
 write= (Button) findViewById(R.id.*button*);  
 read= (Button) findViewById(R.id.*button2*);  
 clear= (Button) findViewById(R.id.*button3*);  
 write.setOnClickListener(new View.OnClickListener()  
 {@Override  
 public void onClick(View v)  
 {  
 String message=e1.getText().toString();  
 try{  
 File f=new File("/SD Card/myfile.txt");  
 f.createNewFile();  
 FileOutputStream fout=new FileOutputStream(f);  
 fout.write(message.getBytes());  
 fout.close();  
 Toast.*makeText*(getBaseContext(),"Data Written in SDCARD",Toast.*LENGTH\_LONG*).show();  
 }  
 catch (Exception e)  
 {  
 Toast.*makeText*(getBaseContext(),e.getMessage(),Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 });  
 read.setOnClickListener(new View.OnClickListener()  
 {  
 @Override  
 public void onClick(View v)  
 {  
 String message;  
 String buf = "";  
 try  
 {  
 File f = new File("/SD Card/myfile.txt");  
 FileInputStream fin = new FileInputStream(f);  
 BufferedReader br = new BufferedReader(new InputStreamReader(fin));  
 while ((message = br.readLine()) != null)  
 {  
 buf += message;  
 }  
 e1.setText(buf);  
 br.close();  
 fin.close();  
 Toast.*makeText*(getBaseContext(),"Data Received from SDCARD",Toast.*LENGTH\_LONG*).show();  
 }  
 catch (Exception e)  
 {  
 Toast.*makeText*(getBaseContext(), e.getMessage(), Toast.*LENGTH\_LONG*).show();  
 }  
 } });  
 clear.setOnClickListener(new View.OnClickListener()  
 {  
 @Override  
 public void onClick(View v)  
 {  
 e1.setText("");  
 }  
 });  
 }}

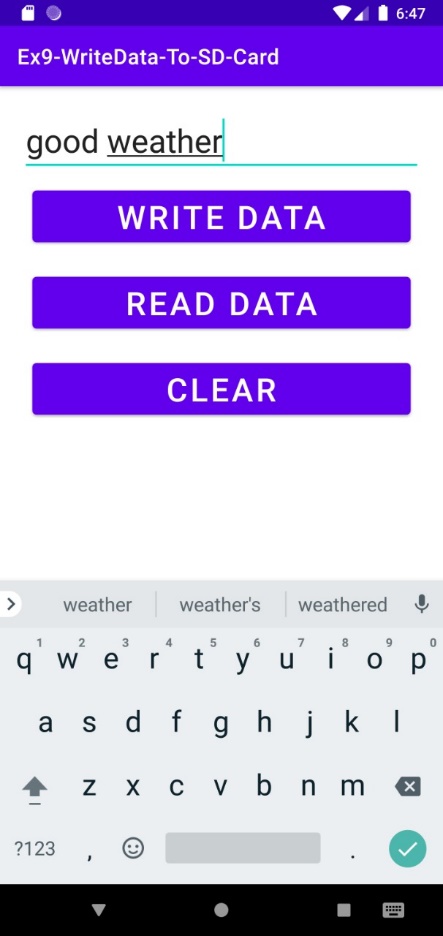
**AndroidManifest.xml:**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.ex09">  
  
 <uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />  
 <uses-permission android:name="android.permission.INTERNET "></uses-permission>  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.Ex09">  
 <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>  
 </activity>  
 </application>  
  
</manifest>

**Activity\_main.xml:**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_margin="20dp"  
 android:orientation="vertical">  
<EditText  
android:id="@+id/editText"  
android:layout\_width="match\_parent"  
android:layout\_height="wrap\_content"  
android:singleLine="true"  
android:textSize="30dp" />  
<Button  
android:id="@+id/button"  
android:layout\_width="match\_parent"  
android:layout\_height="wrap\_content"  
android:layout\_margin="10dp"  
android:text="Write Data"  
android:textSize="30dp" />  
<Button  
android:id="@+id/button2"  
android:layout\_width="match\_parent"  
android:layout\_height="wrap\_content"  
android:layout\_margin="10dp"  
android:text="Read data"  
android:textSize="30dp" />  
<Button  
android:id="@+id/button3"  
android:layout\_width="match\_parent"  
android:layout\_height="wrap\_content"  
android:layout\_margin="10dp"  
android:text="Clear"  
android:textSize="30dp" />  
 </LinearLayout>

**OUTPUT:**





**RESULT:**

Thus Android Application that writes data to the SD Card is developed and executed successfully.

**EX NO: 10 Implement an application that creates an alert upon receiving a message.**

**DATE: 20/09/21**

**AIM:**

To Implement an application that creates an alert upon receiving a message.

**SOURCE CODE:**

**Activity\_main.xml:**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_margin="10dp"  
 android:orientation="vertical">  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Message"  
 android:textSize="30sp" />  
 <EditText  
 android:id="@+id/editText"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:singleLine="true"  
 android:textSize="30sp" />  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_margin="30dp"  
 android:layout\_gravity="center"  
 android:text="Notify"  
 android:textSize="30sp"/>  
</LinearLayout>

**MainActivity.java:**

package com.example.ex10;  
  
import android.app.Notification;  
import android.app.NotificationManager;  
import android.app.PendingIntent;  
import android.content.Intent;  
import android.os.Bundle;  
  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import androidx.appcompat.app.AppCompatActivity;  
public class MainActivity extends AppCompatActivity  
{  
 Button notify;  
 EditText e;  
 @Override  
 protected void onCreate(Bundle savedInstanceState)  
 {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 notify= (Button) findViewById(R.id.*button*);  
 e= (EditText) findViewById(R.id.*editText*);  
  
 notify.setOnClickListener(new View.OnClickListener()  
 {  
 @Override  
 public void onClick(View v)  
 {  
 Intent intent = new Intent(MainActivity.this, SecondActivity.class);  
 PendingIntent pending = PendingIntent.*getActivity*(MainActivity.this, 0, intent, 0);  
 Notification noti = new Notification.Builder(MainActivity.this).setContentTitle("New Message").setContentText(e.getText().toString()).setSmallIcon(R.mipmap.*ic\_launcher*).setContentIntent(pending).build();  
 NotificationManager manager = (NotificationManager) getSystemService(*NOTIFICATION\_SERVICE*);  
 noti.flags |= Notification.*FLAG\_AUTO\_CANCEL*;  
 manager.notify(1, noti);  
 }  
 });  
 }  
}

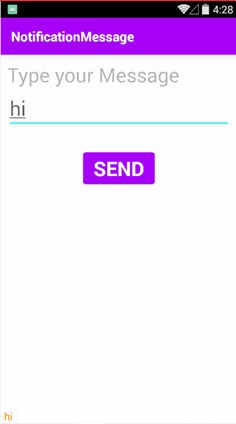
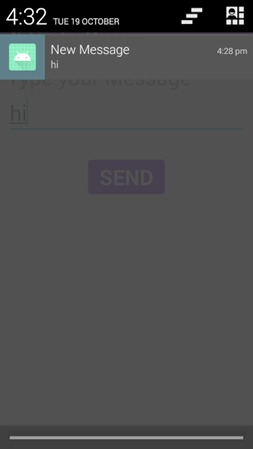
**SecondActivity.java:**

package com.example.ex10;  
import android.os.Bundle;  
  
import androidx.appcompat.app.AppCompatActivity;  
public class SecondActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_second*);  
 }  
}

**activity\_second.java:**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:layout\_margin="10dp"  
 android:orientation="vertical">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="hi"  
 android:textSize="30sp" />  
</LinearLayout>

**OUTPUT:**

**RESULT:**

Thus an application that creates an alert upon receiving a message has been developed and executed successfully.

**EX NO: 11 Write a mobile application that creates an alarm clock.**

**DATE: 4/10/21**

**AIM:**

To Implement a mobile application that creates an alarm clock.

**SOURCE CODE:**

**AndroidManifest.xml:**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.example.ex11">  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.Ex11">  
 <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>  
 </activity>  
 <receiver android:name=".AlarmReceiver" >  
 </receiver>  
 </application>  
  
</manifest>

**Activity\_main:**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <TimePicker  
 android:id="@+id/timePicker"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center" />  
  
 <ToggleButton  
 android:id="@+id/toggleButton"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center"  
 android:layout\_margin="20dp"  
 android:checked="false"  
 android:onClick="OnToggleClicked" />  
  
</LinearLayout>

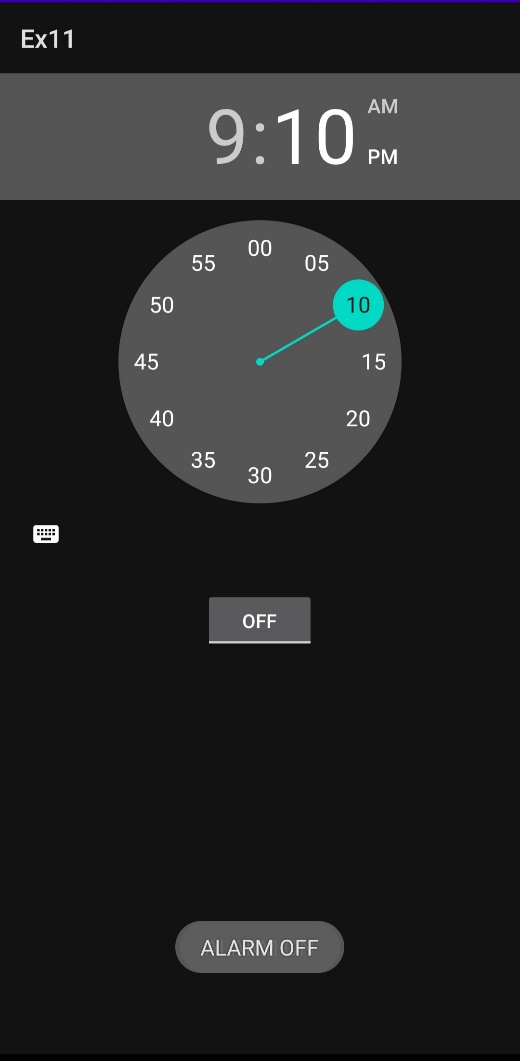
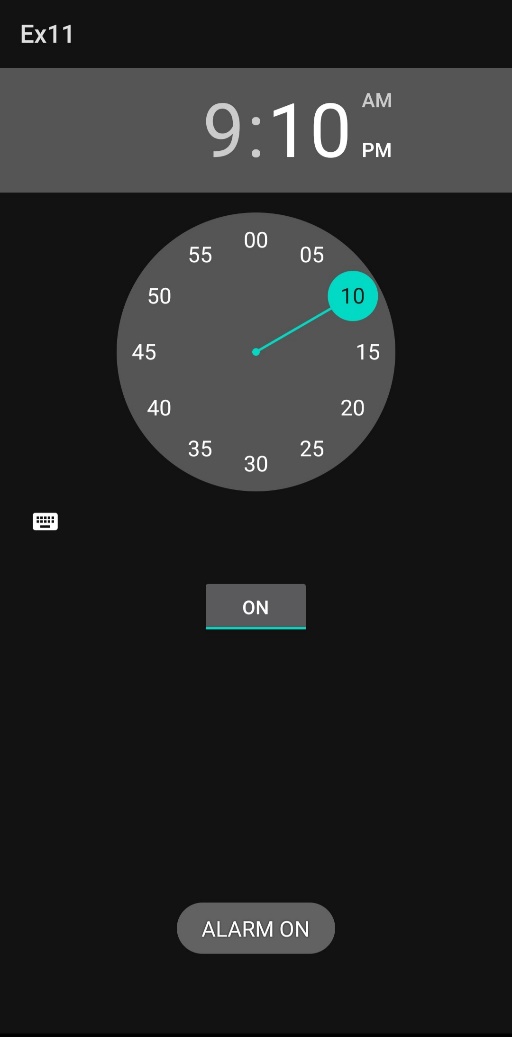
**MainActivity.java:**

package com.example.ex11;  
  
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 androidx.appcompat.app.AppCompatActivity;  
  
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 = (TimePicker) 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.set(Calendar.*HOUR\_OF\_DAY*, alarmTimePicker.getCurrentHour());  
 calendar.set(Calendar.*MINUTE*, alarmTimePicker.getCurrentMinute());  
 Intent intent = new Intent(this, AlarmReceiver.class);  
 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);  
 }  
 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.example.ex11;  
  
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.widget.Toast;  
  
public class AlarmReceiver extends BroadcastReceiver  
{  
 @Override  
 public void onReceive(Context context, Intent intent)  
 {  
 Toast.*makeText*(context, "Alarm! Wake up! Wake up!", Toast.*LENGTH\_LONG*).show();  
 Uri alarmUri = RingtoneManager.*getDefaultUri*(RingtoneManager.*TYPE\_ALARM*);  
 if (alarmUri == null)  
 {  
 alarmUri = RingtoneManager.*getDefaultUri*(RingtoneManager.*TYPE\_NOTIFICATION*);  
 }  
 Ringtone ringtone = RingtoneManager.*getRingtone*(context, alarmUri);  
 ringtone.play();  
 }  
}

**OUTPUT:**

**RESULT:**

Thus an application that creates an alarm clock has been developed and executed successfully.