EX 9 (not working)

ANDROID MANIFEST

```
<?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.REQUEST_IGNORE_BATTERY_OPTIMIZATIONS" />

    <uses-permission android:name="android.permission.VIBRATE"/>
    <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.Ex9"
            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>
        </activity>
        <receiver android:name=".AlarmReceiver"/>

    </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">

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

    <!--Added Toggle Button to set the alarm on or off-->
    <!--ByDefault toggleButton is set to false-->
    <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" />
    <!--"OnToggleClicked" method will be implemented in MainActivity.java -->

</LinearLayout>
```

MAIN ACTIVITY

```
package com.example.ex9

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.*

class MainActivity : AppCompatActivity() {
    private lateinit var alarmTimePicker: TimePicker
    private lateinit var pendingIntent: PendingIntent
    private lateinit var alarmManager: AlarmManager

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

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

    // onToggleClicked() method is implemented for the time functionality
    fun onToggleClicked(view: View) {
        var time: Long
        if ((view as ToggleButton).isChecked) {
            Toast.makeText(this@MainActivity, "ALARM ON",
Toast.LENGTH_SHORT).show()
            val calendar = Calendar.getInstance()

            // calendar is called to get the current time in hour and minute
            calendar[Calendar.HOUR_OF_DAY] = alarmTimePicker.currentHour
            calendar[Calendar.MINUTE] = alarmTimePicker.currentMinute

            // using intent I have class AlarmReceiver class which inherits
            // BroadcastReceiver
            val intent = Intent(this, AlarmReceiver::class.java)

            // we call broadcast using pendingIntent
            pendingIntent = PendingIntent.getBroadcast(this, 0, intent, 0)

            time = calendar.timeInMillis - calendar.timeInMillis % 60000
            if (System.currentTimeMillis() > time) {
                // setting time as AM and PM
                if (calendar[Calendar.AM_PM] == 0) time += 1000 * 60 * 60 *
12.toLong()
                else time += 1000 * 60 * 60 * 24.toLong()
            }
            // Alarm rings continuously until the toggle button is turned off
            alarmManager.setRepeating(AlarmManager.RTC_WAKEUP, time, 10000,
pendingIntent)
            // alarmManager.set(AlarmManager.RTC_WAKEUP, System.currentTimeMillis()
+ (time * 1000), pendingIntent);
        } else {
            alarmManager.cancel(pendingIntent)
            Toast.makeText(this@MainActivity, "ALARM OFF",
Toast.LENGTH_SHORT).show()
        }
    }
}
```

ALARM RECEIVER.KT

```
package com.example.ex9
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.Build
import android.os.Vibrator
import android.widget.Toast

class AlarmReceiver : BroadcastReceiver() {
    override fun onReceive(context: Context, intent: Intent) {
        // We will use vibrator first
        val vibrator = context.getSystemService(Context.VIBRATOR_SERVICE) as
Vibrator
        vibrator.vibrate(4000)

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

        // Setting default ringtone
        val ringtone: Ringtone = RingtoneManager.getRingtone(context, alarmUri)

        // Play ringtone
        ringtone.play()
    }
}
```