Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:app="http://schemas.android.com/apk/res-auto"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  tools:context=".MainActivity">
  <ToggleButton
    android:id="@+id/alarmToggle"
    android:layout_width="wrap_content"
    android:layout height="wrap content"
    android:textOff="@string/alarm_off_toast"
    android:textOn="@string/alarm_on_toast"
    app:layout constraintBottom toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Mainactivity.java

```
package com.example.standup;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.NotificationCompat;
import android.app.AlarmManager;
import android.app.NotificationChannel;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.graphics.Color;
import android.os.Bundle;
import android.os.SystemClock;
import android.widget.CompoundButton;
import android.widget.Toast;
import android.widget.ToggleButton;
public class MainActivity extends AppCompatActivity {
  // Notification ID.
  private static final int NOTIFICATION_ID = 0;
  // Notification channel ID.
  private static final String PRIMARY_CHANNEL_ID =
```

```
"primary_notification_channel";
private NotificationManager mNotificationManager;
/**
* Initializes the activity.
* @param savedInstanceState The current state data.
*/
@Override
protected void onCreate(Bundle savedInstanceState) {
  super.onCreate(savedInstanceState);
  setContentView(R.layout.activity main);
 mNotificationManager = (NotificationManager)
      getSystemService(NOTIFICATION SERVICE);
  ToggleButton alarmToggle = findViewById(R.id.alarmToggle);
 // Set up the Notification Broadcast Intent.
  Intent notifyIntent = new Intent(this, AlarmReceiver.class);
  boolean alarmUp = (PendingIntent.getBroadcast(this, NOTIFICATION_ID,
      notifyIntent, PendingIntent.FLAG NO CREATE) != null);
  alarmToggle.setChecked(alarmUp);
```

```
final PendingIntent notifyPendingIntent = PendingIntent.getBroadcast
    (this, NOTIFICATION_ID, notifyIntent,
        PendingIntent.FLAG_UPDATE_CURRENT);
final AlarmManager alarmManager = (AlarmManager) getSystemService
    (ALARM_SERVICE);
// Set the click listener for the toggle button.
alarm Toggle. set On Checked Change Listener\\
    (new CompoundButton.OnCheckedChangeListener() {
      @Override
      public void onCheckedChanged
          (CompoundButton buttonView, boolean isChecked) {
        String toastMessage;
        if (isChecked) {
          long repeatInterval = AlarmManager.INTERVAL_FIFTEEN_MINUTES/15;
          long triggerTime = SystemClock.elapsedRealtime()
              + repeatInterval;
          // If the Toggle is turned on, set the repeating alarm with
          // a 15 minute interval.
          if (alarmManager != null) {
            alarmManager.setInexactRepeating
```

});

```
(AlarmManager.ELAPSED_REALTIME_WAKEUP,
               triggerTime, repeatInterval,
               notifyPendingIntent);
    }
    // Set the toast message for the "on" case.
    toastMessage = getString(R.string.alarm_on_toast);
  } else {
    // Cancel notification if the alarm is turned off.
    mNotificationManager.cancelAll();
    if (alarmManager != null) {
      alarmManager.cancel(notifyPendingIntent);
    }
    // Set the toast message for the "off" case.
    toastMessage = getString(R.string.alarm_off_toast);
  }
  // Show a toast to say the alarm is turned on or off.
  Toast.makeText(MainActivity.this, toastMessage,
      Toast.LENGTH_SHORT).show();
}
```

```
// Create the notification channel.
 createNotificationChannel();
}
/**
* Creates a Notification channel, for OREO and higher.
*/
public void createNotificationChannel() {
 // Create a notification manager object.
  mNotificationManager =
      (NotificationManager) getSystemService(NOTIFICATION_SERVICE);
 // Notification channels are only available in OREO and higher.
 // So, add a check on SDK version.
  if (android.os.Build.VERSION.SDK_INT >=
      android.os.Build.VERSION_CODES.O) {
    // Create the NotificationChannel with all the parameters.
    NotificationChannel notificationChannel = new NotificationChannel
        (PRIMARY_CHANNEL_ID,
             "Stand up notification",
             NotificationManager.IMPORTANCE_HIGH);
```

AlarmReciver.java

```
package com.example.standup;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import androidx.core.app.NotificationCompat;
public class AlarmReceiver extends BroadcastReceiver {
  private NotificationManager mNotificationManager;
  private static final int NOTIFICATION_ID = 0;
  // Notification channel ID.
  private static final String PRIMARY_CHANNEL_ID =
      "primary_notification_channel";
  @Override
  public void onReceive(Context context, Intent intent) {
    mNotificationManager = (NotificationManager)
        context.getSystemService(Context.NOTIFICATION_SERVICE);
    deliverNotification(context);
  }
  private void deliverNotification(Context context) {
```

```
Intent contentIntent = new Intent(context, MainActivity.class);

PendingIntent contentPendingIntent = PendingIntent.getActivity

(context, NOTIFICATION_ID, contentIntent, PendingIntent.FLAG_UPDATE_CURRENT);

NotificationCompat.Builder builder = new NotificationCompat.Builder(context,
PRIMARY_CHANNEL_ID)

.setSmallIcon(R.drawable.ic_launcher_foreground)

.setContentTitle("Stand Up Alert")

.setContentText("You should stand up and walk around now!")

.setContentIntent(contentPendingIntent)

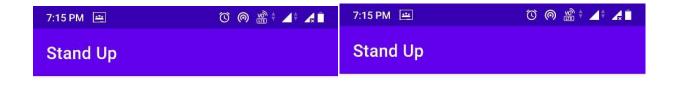
.setPriority(NotificationCompat.PRIORITY_HIGH)

.setAutoCancel(true)

.setDefaults(NotificationCompat.DEFAULT_ALL);

mNotificationManager.notify(NOTIFICATION_ID, builder.build());

}
```



ALARM ON

ALARM OFF

