

CODE:

```
import android.app.AlarmManager;
import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.app.Service;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.hardware.SensorEventListener;
import android.os.Build;
import android.os.IBinder;

import java.text.NumberFormat;
import java.util.Date;
import java.util.Locale;

public class SensorListener extends Service implements SensorEventListener {
    public final static int NOTIFICATION_ID = 1;

    private static int steps;

    private void showNotification() {
        if (Build.VERSION.SDK_INT >= 26) {
            startForeground(NOTIFICATION_ID, getNotification(this));
        } else if (getSharedPreferences("pedometer", Context.MODE_PRIVATE)
            .getBoolean("notification", true)) {
            ((NotificationManager)
getSystemService(Context.NOTIFICATION_SERVICE))
                .notify(NOTIFICATION_ID, getNotification(this));
        }
    }

    @Override
    public IBinder onBind(final Intent intent) {
        return null;
    }

    @Override
    public int onStartCommand(final Intent intent, int flags, int startId) {
        // restart service every hour to save the current step count
        long nextUpdate = Math.min(Util.getTomorrow(),
            System.currentTimeMillis() + AlarmManager.INTERVAL_HOUR);
        if (BuildConfig.DEBUG) Logger.log("next update: " + new
Date(nextUpdate).toLocaleString());
        AlarmManager am =
            (AlarmManager)
getApplicationContext().getSystemService(Context.ALARM_SERVICE);
```

```

        PendingIntent pi = PendingIntent
            .getService(getApplicationContext(), 2, new Intent(this,
SensorListener.class),
                PendingIntent.FLAG_UPDATE_CURRENT);
        if (Build.VERSION.SDK_INT >= 23) {
            API23Wrapper.setAlarmWhileIdle(am, AlarmManager.RTC, nextUpdate, pi);
        } else {
            am.set(AlarmManager.RTC, nextUpdate, pi);
        }

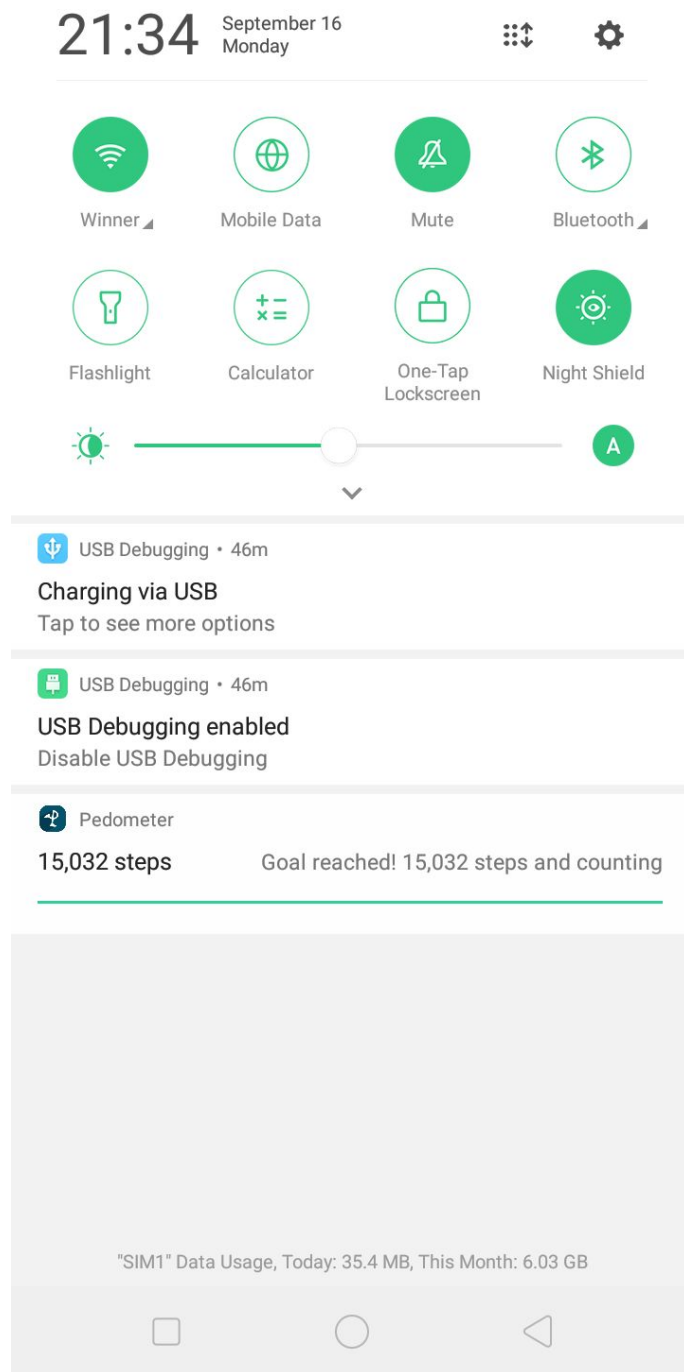
        return START_STICKY;
    }

    public static Notification getNotification(final Context context) {
        if (BuildConfig.DEBUG) Logger.log("getNotification");
        SharedPreferences prefs = context.getSharedPreferences("pedometer",
Context.MODE_PRIVATE);
        int goal = prefs.getInt("goal", 10000);
        Database db = Database.getInstance(context);
        int today_offset = db.getSteps(Util.getToday());
        if (steps == 0)
            steps = db.getCurrentSteps(); // use saved value if we haven't
anything better
        db.close();
        Notification.Builder notificationBuilder =
            Build.VERSION.SDK_INT >= 26 ?
API26Wrapper.getNotificationBuilder(context) :
            new Notification.Builder(context);

        NumberFormat format = NumberFormat.getInstance(Locale.getDefault());
        notificationBuilder.setProgress(goal, today_offset + steps,
false).setContentText(
            today_offset + steps >= goal ?
                context.getString(R.string.goal_reached_notification,
                    format.format((today_offset + steps))) :
                context.getString(R.string.notification_text,
                    format.format((goal - today_offset -
steps))))).setTitle(
            format.format(today_offset + steps) + " " +
context.getString(R.string.steps));
        return notificationBuilder.build();
    }
}

```

OUTPUT:



Output when Goal Steps are complete.