Cloud Messaging

<https://developers.google.com/cloud-messaging/>

# **Notifications**

<http://developer.android.com/guide/topics/ui/notifiers/notifications.html>

http://developer.android.com/design/patterns/notifications.html

## Notifications

[Diff for this node](https://github.com/udacity/Sunshine-Version-2/compare/6.05_scheduled_sync...6.06_notifications_quiz).

You’ll be guided through the first part of adding notifications and then do the rest yourself. The changes you will make now can be found in this [diff](https://github.com/udacity/Sunshine-Version-2/compare/6.05_scheduled_sync...6.06_notifications_quiz).

### Instructions

**1. Add Notification Format**

In the string.xml file add the following lines from this [gist](https://gist.github.com/udacityandroid/d710f32240d0ce7c0810).

**2. Add a Projection**

Because your notification data will be pulled from the database, you need to add projection and column indices values in your SunshineSyncAdapter:

private static final String[] NOTIFY\_WEATHER\_PROJECTION = new String[] {

WeatherContract.WeatherEntry.COLUMN\_WEATHER\_ID,

WeatherContract.WeatherEntry.COLUMN\_MAX\_TEMP,

WeatherContract.WeatherEntry.COLUMN\_MIN\_TEMP,

WeatherContract.WeatherEntry.COLUMN\_SHORT\_DESC

};

// these indices must match the projection

private static final int INDEX\_WEATHER\_ID = 0;

private static final int INDEX\_MAX\_TEMP = 1;

private static final int INDEX\_MIN\_TEMP = 2;

private static final int INDEX\_SHORT\_DESC = 3;

**3. Add Related Constants**

You’ll also add some additional constants at the top of SunshineSyncAdapter; DAY\_IN\_MILLIS is the amount of milliseconds in a day and WEATHER\_NOTIFICATION\_ID is an id you create that is matched to your notification so that you can reuse it. If you reuse the notification ID, your application will post at most one notification.

private static final long DAY\_IN\_MILLIS = 1000 \* 60 \* 60 \* 24;

private static final int WEATHER\_NOTIFICATION\_ID = 3004;

**4. Create notifyWeather Function**

Then you will create the notifyWeather function in SunshineSyncAdapter. Right now this function:

1. Checks whether you’ve already shown a notification today.
2. If you haven’t, then it connects to the database and gets a cursor for the current day.
3. Fetches the data from the database.
4. Formats the data in a string contentText. Notice how there is also the iconId and title for you to use.

[Gist for notifyWeather](https://gist.github.com/udacityandroid/e5eb3afa254ca750e083)

**5. Call notifyWeather**

Call notifyWeather(); within the onPerformSync function. You should call it when weather data is inserted into the database (right after bulkInsert is called).

### Your Task

Right now your notifyWeather function is called and pulls data from the database, but it is not actually showing a notification. Your goal is to finish the notifyWeather function.

To do this, you will need to:

1. Create the Notification using NotificationCompat.builder.
2. Create an explicit intent for what the notification should open.
3. Using TaskStackBuilder, create an artificial “backstack” so that when the user clicks the back button, it is clear to Android where the user will go.
4. Tell the NotificationManager to show the notification.

Check out the [Notifications](http://developer.android.com/guide/topics/ui/notifiers/notifications.html) tutorial for a helpful example.

Related classes:

[NotificationCompat.Builder](http://developer.android.com/reference/android/support/v4/app/NotificationCompat.Builder.html)   
[NotificationManager](http://developer.android.com/reference/android/app/NotificationManager.html)   
[PendingIntent](http://developer.android.com/reference/android/app/PendingIntent.html)   
[TaskStackBuilder](http://developer.android.com/reference/android/support/v4/app/TaskStackBuilder.html)

Continue to Quiz