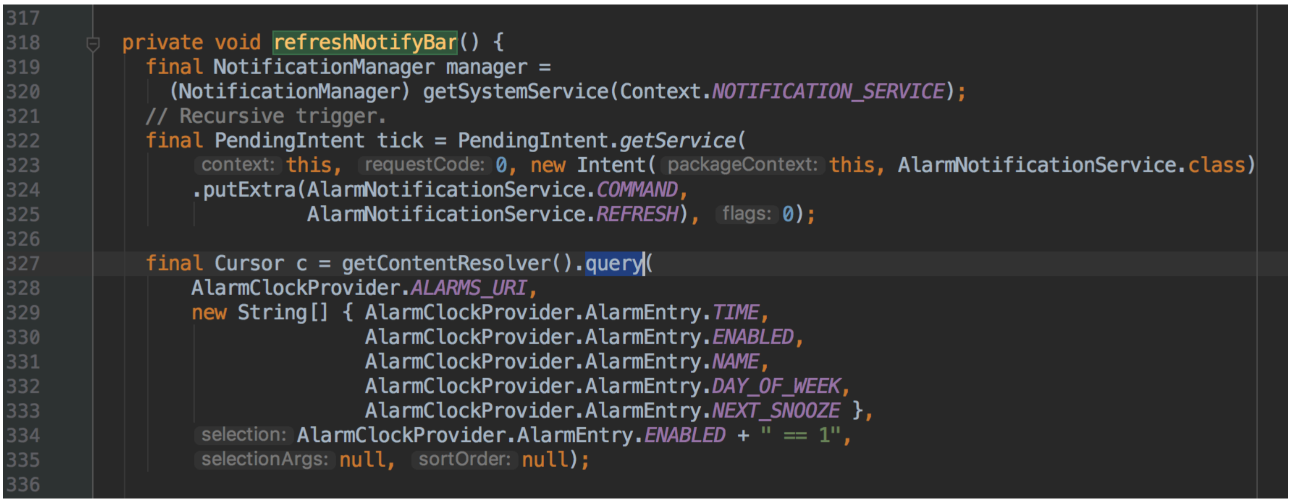
Id : fd-34

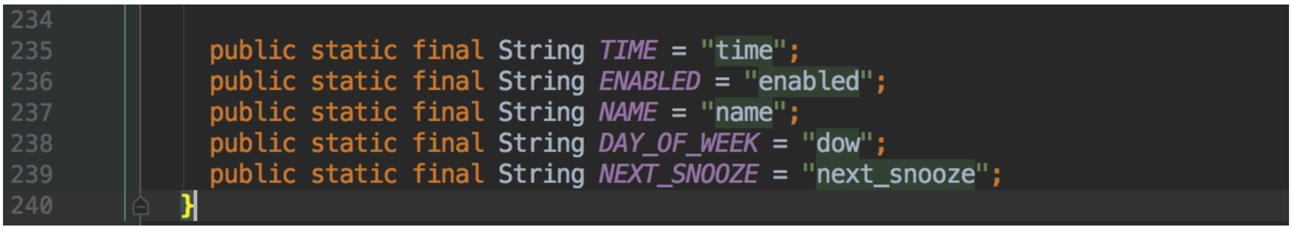
Result : False Positive

Sink statement : Line 327 in below screenshot is the sink statement.



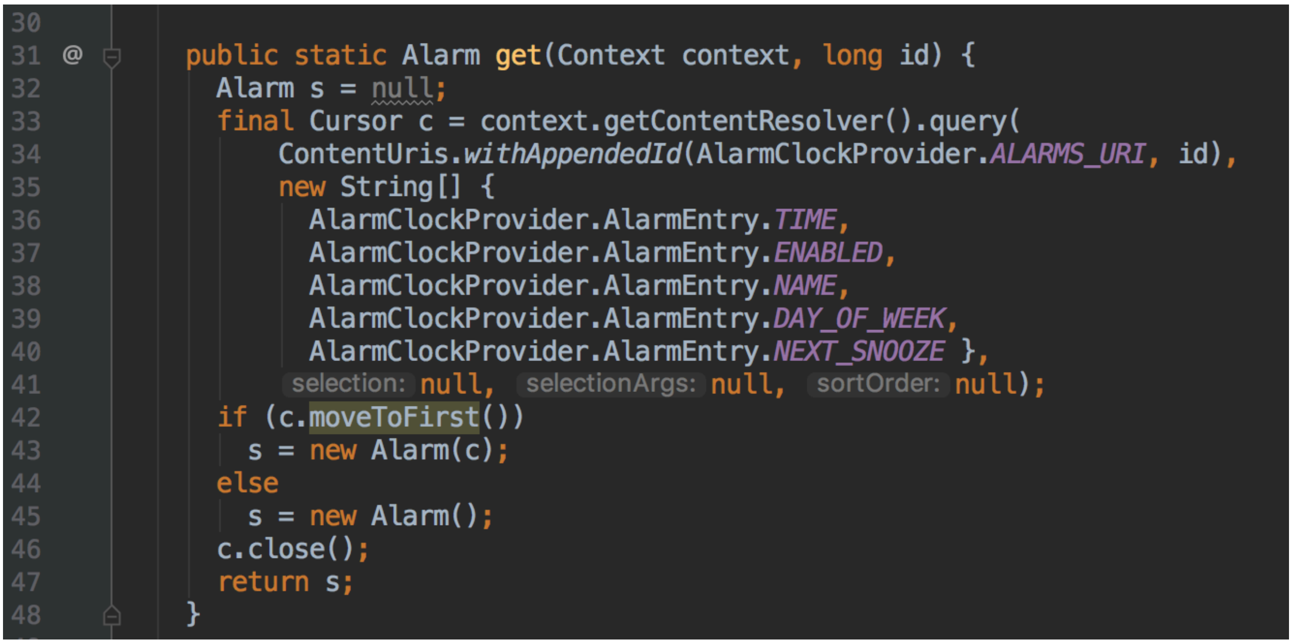
```
317
318   private void refreshNotifyBar() {
319     final NotificationManager manager =
320       (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);
321     // Recursive trigger.
322     final PendingIntent tick = PendingIntent.getService(
323         context: this,  requestCode: 0, new Intent( packageContext: this, AlarmNotificationService.class)
324         .putExtra(AlarmNotificationService.COMMAND,
325                   AlarmNotificationService.REFRESH),  flags: 0);
326
327     final Cursor c = getContentResolver().query(
328         AlarmClockProvider.ALARMS_URI,
329         new String[] { AlarmClockProvider.AlarmEntry.TIME,
330                        AlarmClockProvider.AlarmEntry.ENABLED,
331                        AlarmClockProvider.AlarmEntry.NAME,
332                        AlarmClockProvider.AlarmEntry.DAY_OF_WEEK,
333                        AlarmClockProvider.AlarmEntry.NEXT_SNOOZE },
334         selection: AlarmClockProvider.AlarmEntry.ENABLED + " == 1",
335         selectionArgs: null,  sortOrder: null);
336
```

All the variables to the sink statement
AlarmClockProvider.ALARMS_URI,
AlarmClockProvider.AlarmEntry.TIME,
AlarmClockProvider.AlarmEntry.ENABLED,
AlarmClockProvider.AlarmEntry.NAME,
AlarmClockProvider.AlarmEntry.DAY_OF_WEEK,
Are final static constants



```
234
235     public static final String TIME = "time";
236     public static final String ENABLED = "enabled";
237     public static final String NAME = "name";
238     public static final String DAY_OF_WEEK = "dow";
239     public static final String NEXT_SNOOZE = "next_snooze";
240   }
```

Source Statement : Line 33 in following screenshot



```
30
31 @   public static Alarm get(Context context, long id) {
32       Alarm s = null;
33       final Cursor c = context.getContentResolver().query(
34           ContentUris.withAppendedId(AlarmClockProvider.ALARMS_URI, id),
35           new String[] {
36             AlarmClockProvider.AlarmEntry.TIME,
37             AlarmClockProvider.AlarmEntry.ENABLED,
38             AlarmClockProvider.AlarmEntry.NAME,
39             AlarmClockProvider.AlarmEntry.DAY_OF_WEEK,
40             AlarmClockProvider.AlarmEntry.NEXT_SNOOZE },
41           selection: null,  selectionArgs: null,  sortOrder: null);
42       if (c.moveToFirst())
43         s = new Alarm(c);
44       else
45         s = new Alarm();
46       c.close();
47       return s;
48     }
```