

HOW TO IMPLEMENT NOTIFICATIONS IN AN INTELLIJ PLUGIN

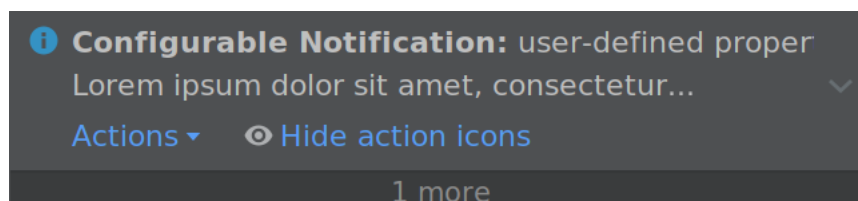
Introduction

This article explains what notifications are, how to implement them in your plugins, and how to interact with them.

There are different types of notifications in your IDE:

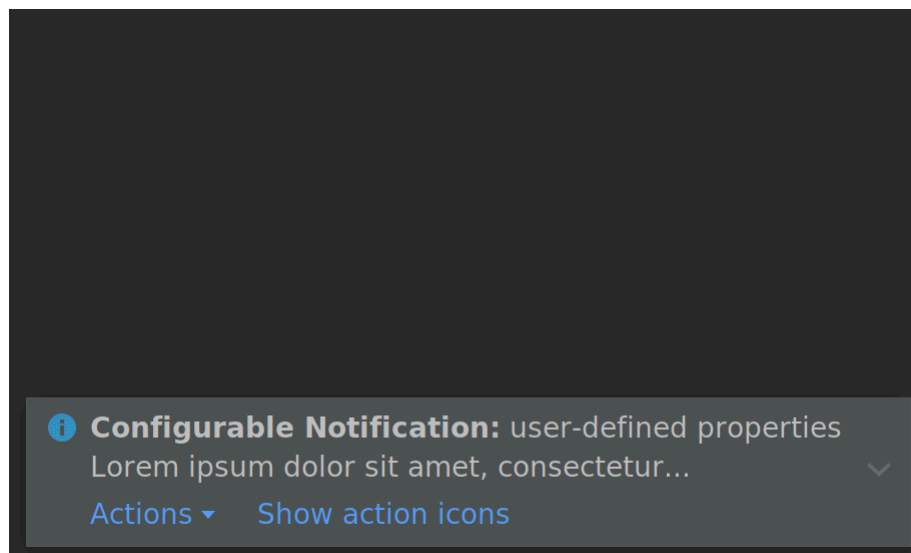
- native system notifications
- editor hints
- modal popups
- non-modal notifications.

We're only discussing the last type here: **non-modal notifications**. These are the balloon popups that appear in the lower right corner of the IDE's main window.



Sample code

If you'd like to play with notifications and to learn what they can do, then checkout the GitHub repository [jansorg/intellij-notifications](https://github.com/jansorg/intellij-notifications)¹. It implements a notification which offers actions to modify all properties and to immediately see the result.



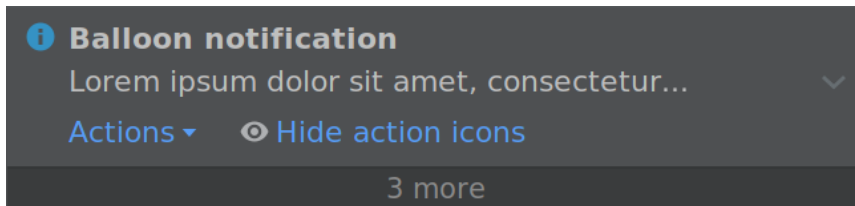
1. <https://github.com/jansorg/intellij-notifications>

Notification display type

A notification can be rendered in several ways. Enum `com.intellij.notification.NotificationDisplayType` defines the available types.

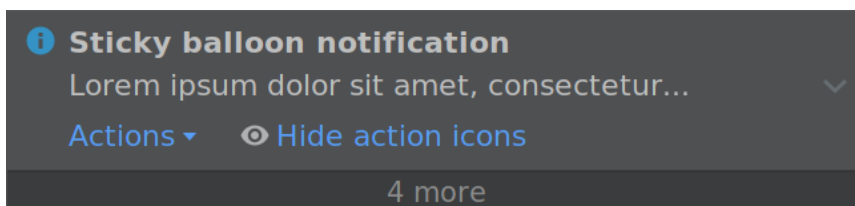
Balloon notifications, *NotificationDisplayType.BALLOON*¹

This is the default representation of notifications. Balloon notifications are rendered in the lower right corner of the main window. They expire after 10 seconds.



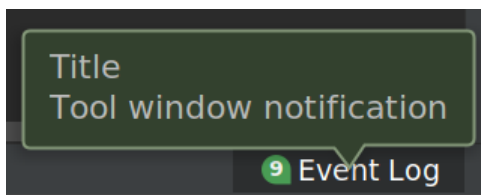
Sticky balloon notifications, *NotificationDisplayType.STICKY_BALLOON*²

A sticky notification is like a regular balloon notification, but it won't expire automatically. It will remain visible on screen until you close it.



Tool window notifications, *NotificationDisplayType.TOOL_WINDOW*³

A tool window notification shows up next to the button of a tool window. It only supports a subset of the available properties.



Logged only, *NotificationDisplayType.NONE*⁴

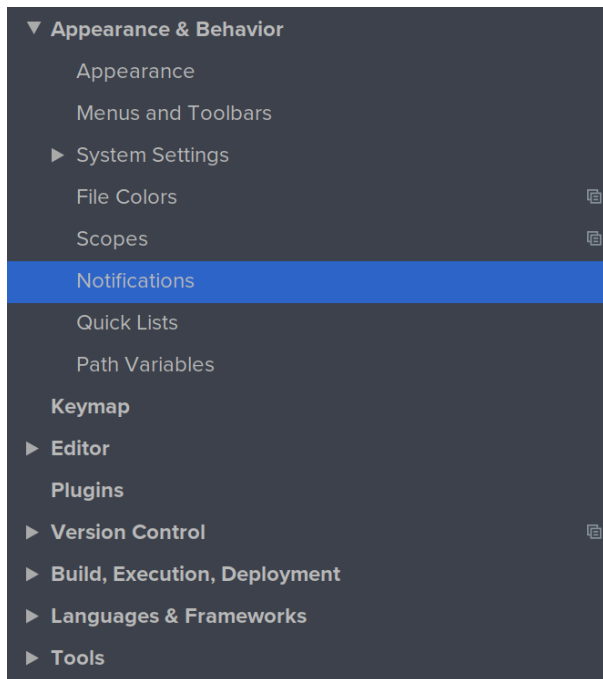
A notification of this type has no graphical representation. It shows up in the event log, though.

1. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/NotificationDisplayType.java#L25>
2. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/NotificationDisplayType.java#L27>
3. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/NotificationDisplayType.java#L28>
4. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/NotificationDisplayType.java#L23>

Notification groups

A notification group has a unique ID and defines shared properties.

Each notification is assigned to a group. This group defines how assigned notifications are displayed. A user can override these defaults in the preferences.



Where to configure notification groups

Creating notification groups

Class `com.intellij.notification.NotificationGroup` defines a group. It provides some overloaded constructors and static methods to create the most-commonly used types of groups.

```
package demo;

import com.intellij.notification.NotificationGroup;

class Demo {
    public void demo() {
        // notification group for non-sticky balloon notifications
        NotificationGroup.balloonGroup("my.group.id");
        // or:
        new NotificationGroup("my.group.id",
            NotificationDisplayType.BALLOON,
            true);

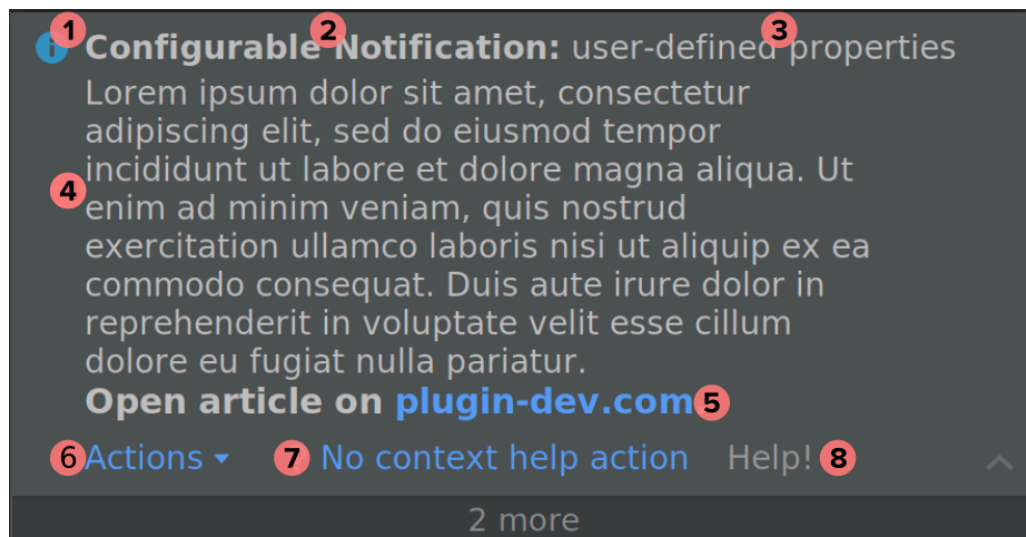
        // notification group for sticky balloon notifications
        new NotificationGroup("my.group.id",
            NotificationDisplayType.STICKY_BALLOON,
            true);

        // tool window notifications
        new NotificationGroup("demo.notifications.toolWindow",
            NotificationDisplayType.TOOL_WINDOW,
            true);
    }
}
```

Notifications

Properties of Notifications

A notification is represented by `com.intellij.notification.Notification`.



Properties of an IntelliJ notification

Group

The group assigned to a notification. See *above*¹ for details. A notification references a group with a string ID. The shared properties, e.g. display type, will be automatically used for the notifications.

Notification type

Every notification has a type. The type defines some of the visual properties, e.g. icon and color. The available types are `Information`, `Warning`, and `Error`. It's easy to mix up notification display type (property of a group) and notification type (property of a notification).

(1) Notification icon

The icon is shown in the top left corner. By default, it's derived from the notification type. Define your own icon to override it.

(2) Title (optional)

A string, which is displayed as the heading of the notification.

(3) Subtitle (optional)

A string, which is displayed next to the title.

(4) Content

A string, which defines the message of the notification. HTML tags are allowed. Only three lines of text are shown by default. That's either one line of title and subtitle plus two lines of content. Or, if there's no title, three lines of content.

Expanded content

You need to implement the marker interface

1. See "Notification groups", page 4

`com.intellij.notification.impl.NotificationFullContent` to show all the content by default.

(5) Content links

HTML links are supported. It's better to use actions whenever possible, though. If you want to use links, then you need to set a *NotificationListener*¹.

(6) Actions dropdown

If the actions of a notification don't fit into the available space, then a dropdown is displayed to show all hidden actions. The dropdown can be displayed on the left or on the right.

(7) Actions

A notification supports a set of action instances. The icon of an action is shown, but the description is not rendered.

(8) Context help

A notification supports a context help action. The title of this action is displayed as a grayed-out label. The action's description is displayed as a tooltip when you hover over the label of the context help.

Classes

These are the most relevant classes of the IntelliJ SDK.

com.intellij.notification.NotificationGroup²

`NotificationGroup` defines a shared properties for notifications assigned to that group, e.g. the display type.

com.intellij.notification.Notification³

`Notification` is the most important class of the notifications package. Please read the *javadoc documentation*⁴ of this class for a helpful introduction to notifications.

com.intellij.notification.impl.NotificationFullContent⁵

A marker interface. A `Notification`, which implements it, is displayed at full height by default. In this case it's not possible to collapse the content. Please note, that this interface isn't part of the public `openapi` package.

com.intellij.notification.NotificationAction⁶

`NotificationAction` simplifies to work with notification actions. Actions of this type have access to the notification they were added to. For example, that's helpful if you the notification should be expired after the action was executed.

`com.intellij.notification.BrowseNotificationAction` can be used to open a link in the user's browser.

com.intellij.notification.Notifications⁷

Interface `Notifications` is intended to be used with the message bus. You can use it to be notified about notifications. Inner class `Notifications.Bus` defines a `notify()` method, which is used to post notifications. `Notification.notify()` is just calling

1. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/NotificationListener.java>

2. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/NotificationGroup.java>

3. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/Notification.java>

4. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/Notification.java#L26>

5. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-impl/src/com/intellij/notification/impl/NotificationFullContent.java>

6. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/NotificationAction.java>

7. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/Notifications.java>

`Notifications.Bus.notify(...)`¹. Checkout the section about *testing notifications*² to learn how to use `Notifications.Bus`.

Personally, I prefer `notification.notify()` instead of `Notifications.Bus`.

Creating notifications

1. At first, decide if you'd like to customize the appearance. If you like the defaults (i.e. non-sticky balloons), then you don't need to create a group. If you'd like to customize it, then define a group and set the properties. The first parameter, `displayId`, has to be unique. Also, make sure that only one instance of a group exists. Each time an instance is created, its *displayId* is registered in a static map³.

```
class Demo {
    private static final NotificationGroup STICKY_GROUP =
        new NotificationGroup("demo.notifications.balloon",
            NotificationDisplayType.STICKY_BALLOON, true);
}
```

2. Create a notification and pass the group's `displayId`.

```
class Demo {
    private static final NotificationGroup STICKY_GROUP =
        new NotificationGroup("demo.notifications.balloon",
            NotificationDisplayType.STICKY_BALLOON, true);

    public static void createNotification() {
        Notification msg = new Notification(
            STICKY_GROUP.getDisplayId(), icon,
            "Title", "Subtitle", "Content",
            NotificationType.INFORMATION, null);
    }
}
```

There are a few helper methods to create notifications. For example, a group defines the method `createNotification()`. This method is just creating a new instance of `Notification` with the `displayId` of the group.

1. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/Notification.java#L343>

2. See "Testing notifications", page 10

3. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/NotificationGroup.java#L66>

```

class Demo {
private static final NotificationGroup STICKY_GROUP =
    new NotificationGroup("demo.notifications.balloon",
        NotificationDisplayType.STICKY_BALLOON, true);

public static void createNotification() {
    Notification msg = STICKY_GROUP.createNotification(
        "Title", "Subtitle", "Content",
        NotificationType.INFORMATION);
}
}

```

3. Display the notification with `Notification.notify(project)` or `Notifications.Bus.notify(notification, project)`.

```

class Demo {
private static final NotificationGroup STICKY_GROUP =
    new NotificationGroup("demo.notifications.balloon",
        NotificationDisplayType.STICKY_BALLOON, true);

public static void createNotification() {
    Notification msg = STICKY_GROUP.createNotification(
        "Title", "Subtitle", "Content",
        NotificationType.INFORMATION);

    // notify() displays the notification
    msg.notify();
}
}

```

Customizing notifications

You can customize a notification. Modifications have to be made before you show it – they are not applied after a notification was displayed.

Class `Notification` defines setter methods for all supported properties. Here are the most commonly used properties:

notification.setTitle()

Set the title. If no title is set, then one more line of content is shown.

notification.setContent()

Set the content. A basic set of HTML tags is supported.

notification.setIcon()

The icon overrides the default icon, which is derived from the notification type.

notification.addAction()

Add an action to the notification. An icon is rendered next to an action when you defined it for the new action.

Testing notifications

You have at least two choices to test notifications:

1. Listen to new notifications on the message bus (*example*¹)
2. Retrieve a list of unexpired notifications from `NotificationsManager` (*example*²)

The example github repository *demonstrates how to test notifications*³.

1. <https://github.com/jansorg/intellij-notifications/blob/master/src/test/java/dev/ja/samples/ConfigurableNotificationTest.java#L34>

2. <https://github.com/jansorg/intellij-notifications/blob/master/src/test/java/dev/ja/samples/ConfigurableNotificationTest.java#L65>

3. <https://github.com/jansorg/intellij-notifications/blob/master/src/test/java/dev/ja/samples/ConfigurableNotificationTest.java>

FAQ

How do I display a notification at full height?

See the *note on NotificationFullContent*¹.

How do I avoid the automatic expiration of notifications?

Create a notification and assign it to group with `NotificationDisplayType.STICKY_BALLOON`: *example*².

Why do I have to pass a project to `Notification.notify()`?

The method `Notification.notify()` takes a nullable `Project`:
`Notifications.Bus.notify(Notification notification, @Nullable Project project)`

Pass a project to make sure that the notification ends up in the expected window. If you don't pass it, then the notification will be shown in the last activated window.

How do I control the default settings of a notification?

Create a `NotificationGroup` and reference it in your notification. Read more in the *section about groups*³. For example:

```
class Component {  
    // STICKY_BALLOON -> sticky notification  
    // true -> log by default  
    private static final NotificationGroup STICKY_GROUP =  
        new NotificationGroup("demo.notifications.balloon",  
            NotificationDisplayType.STICKY_BALLOON, true);  
}
```

For properties of a notification, see the section about *properties*⁴ and the section on *customizing notifications*⁵.

1. See "Classes", page 7

2. See "Creating notification groups", page 4

3. See "Notification groups", page 4

4. See "Properties of Notifications", page 6

5. See "Customizing notifications", page 9

How do I update the content of a notification, e.g. to show a countdown?

Sorry, that's not possible! Although the properties of a notification can be modified any time, e.g. by `setContent(...)`, these changes are not applied if the notification is already visible on screen.

How do I retrieve a list of open notifications?

Use `getNotificationsOfType()` of `com.intellij.notification.NotificationManager` (*example*¹).

There's not a lot you can do with these notifications, though.

You can hide notifications by enforcing expiration:

```
NotificationsManager mgr =
    NotificationsManager.getNotificationsManager();
Notification[] all = mgr.getNotificationsOfType(
    Notification.class, project);

for (Notification notification : all) {
    notification.expire();
}
```

You could also attach an expiration listener to a notification with `notification.whenExpired(...)`.

How do I customize the buttons of a notification?

There are different buttons attached to a notification:

Close button

A click on the close button expires the notification. You can attach an expiration listener to be notified when that happens. Expiration isn't necessarily caused by a click on the close button, though.

Settings button

Sorry, it's not possible to customize it.

Collapse content button

You can't customize the button to expand collapsed content. You can expand it by default by following *these guidelines*².

Actions

Actions are added to an instance of `Notification` by invoking `notification.addAction()`.

1. <https://github.com/jansorg/intellij-notifications/blob/master/src/test/java/dev/ja/samples/ConfigurableNotificationTest.java#L74>

2. See "", page 0

How do I structure the list of groups in the settings?

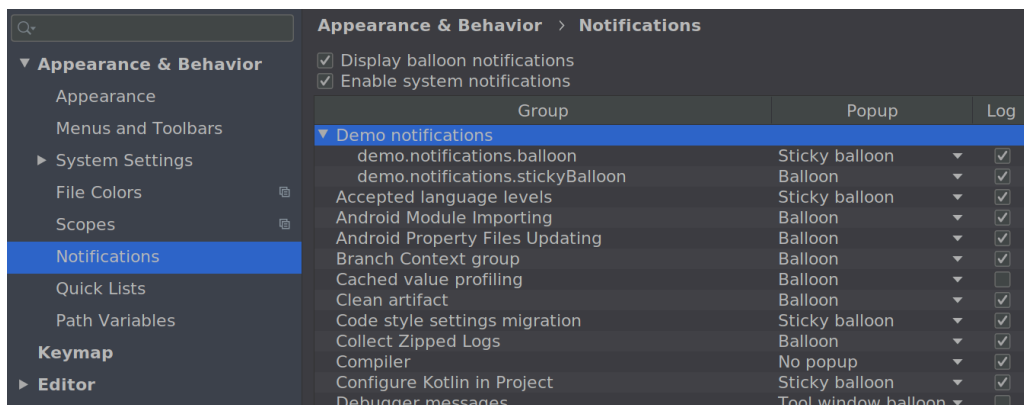
This one is interesting. I noticed this feature while poking around in the sources of the SDK. It's possible to define a hierarchy of notification groups. But, so far, I never have seen this in use.

You need to define an extension for the parent group and to define extensions for each child group:

```
<extensions defaultExtensionNs="com.intellij">
  <notification.parentGroup
    id="demo.notifications"
    title="Demo notifications"/>

  <notification.group
    parentId="demo.notifications"
    groupId="demo.notifications.balloon"/>
  <notification.group
    parentId="demo.notifications"
    groupId="demo.notifications.stickyBalloon"/>
</extensions>
```

This is displayed like this in the settings:



Hierarchy of notification groups

External References

Helpful documentation

- *Top-Level Notifications*¹: Documentation from the IntelliJ Platform SDK DevGuide.
- *Javadoc of class Notification*²: Introduction to the concepts of Notification.

Notifications in the wild

- *BashSupport notification about conflicting plugins*³: Notification which allows the user to disable one of two conflicting plugins.
- *The red exception alert of IntelliJ*⁴: This is how IntelliJ implements the notorious red alert notification. The red background color cannot be implemented with the public API.

Implementation in the SDK

- *NotificationManagerImpl*⁵: This class manages notifications and notification groups. It also receives and handles the messages send via `Notifications.BUS`.
- *package com.intellij.notification.impl*⁶: The package where the internal implementations of the notification system live.

1. http://www.jetbrains.org/intellij/sdk/docs/user_interface_components/notifications.html#top-level-notifications

2. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-api/src/com/intellij/notification/Notification.java#L26>

3. <https://github.com/BashSupport/BashSupport/blob/idea-192.x/src/com/ansorgit/plugins/bash/ide/PluginCompatibilityNotification.java>

4. <https://github.com/JetBrains/intellij-community/blob/6d4dbf40c0a34e1103fbd56609671f81b0f4761/platform/platform-impl/src/com/intellij/ide/notifications/NotificationManagerImpl.java#L197>

5. <https://github.com/JetBrains/intellij-community/blob/193/platform/platform-impl/src/com/intellij/notification/impl/NotificationManagerImpl.java>

6. <https://github.com/JetBrains/intellij-community/tree/193/platform/platform-impl/src/com/intellij/notification/impl>