

# NotificationManager

```
public class NotificationManager
extends Object (https://developer.android.com/reference/java/lang/Object.html)

java.lang.Object (https://developer.android.com/reference/java/lang/Object.html)
↳ android.app.NotificationManager
```

added in API level 1  
(https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)  
Summary: Nested Classes (#nestedclasses) | Constants (#constants) | Methods (#publmethods) | Inherited Methods (#inhmethods) | [Expand All] (#)

Class to notify the user of events that happen. This is how you tell the user that something has happened in the background.

Notifications can take different forms:

- A persistent icon that goes in the status bar and is accessible through the launcher, (when the user selects it, a designated Intent can be launched),
- Turning on or flashing LEDs on the device, or
- Alerting the user by flashing the backlight, playing a sound, or vibrating.

Each of the notify methods takes an int id parameter and optionally a **String** (https://developer.android.com/reference/java/lang/String.html) tag parameter, which may be **null**. These parameters are used to form a pair (tag, id), or (**null**, id) if tag is unspecified. This pair identifies this notification from your app to the system, so that pair should be unique within your app. If you call one of the notify methods with a (tag, id) pair that is currently active and a new set of notification parameters, it will be updated. For example, if you pass a new status bar icon, the old icon in the status bar will be replaced with the new one. This is also the same tag and id you pass to the **cancel(int)** (https://developer.android.com/reference/android/app/NotificationManager.html#cancel(int)) OR **cancel(String, int)** (https://developer.android.com/reference/android/app/NotificationManager.html#cancel(java.lang.String, int)) method to clear this notification.

## Developer Guides

For a guide to creating notifications, read the Status Bar Notifications (https://developer.android.com/guide/topics/ui/notifiers/notifications.html) developer guide.

Instances of this class must be obtained using **Context.getSystemService(Class)** (https://developer.android.com/reference/android/content/Context.html#getSystemService(java.lang.Class<T>)) with the argument **NotificationManager.class** or **Context.getSystemService(String)** (https://developer.android.com/reference/android/content/Context.html#getSystemService(java.lang.String)) with the argument **Context.NOTIFICATION\_SERVICE** (https://developer.android.com/reference/android/content/Context.html#NOTIFICATION\_SERVICE).

See also:

**Notification** (https://developer.android.com/reference/android/app/Notification.html)

## Summary

Nested classes	
class	<b>NotificationManager.Policy</b> (https://developer.android.com/reference/android/app/NotificationManager.Policy.html) Notification policy configuration.

Constants	
<b>String</b> (https://developer.android.com/reference/java/lang/String.html)	<b>ACTION_INTERRUPTION_FILTER_CHANGED</b> (https://developer.android.com/reference/android/app/NotificationManager.html#ACTION_INTERRUPTION_FILTER_CHAN Intent that is broadcast when the state of <b>getCurrentInterruptionFilter()</b> changes.
<b>String</b> (https://developer.android.com/reference/java/lang/String.html)	<b>ACTION_NOTIFICATION_POLICY_ACCESS_GRANTED_CHANGED</b> (https://developer.android.com/reference/android/app/NotificationManager.html#ACTION_NOTIFICATION_POLICY_ACCE Intent that is broadcast when the state of <b>isNotificationPolicyAccessGranted()</b> (https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGrant
<b>String</b> (https://developer.android.com/reference/java/lang/String.html)	<b>ACTION_NOTIFICATION_POLICY_CHANGED</b> (https://developer.android.com/reference/android/app/NotificationManager.html#ACTION_NOTIFICATION_POLICY_CHAN Intent that is broadcast when the state of <b>getNotificationPolicy()</b> changes.

int	<b>IMPORTANCE_DEFAULT</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_DEFAULT">https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_DEFAULT</a> ) Default notification importance: shows everywhere, makes noise, but does not visually intrude.
int	<b>IMPORTANCE_HIGH</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_HIGH">https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_HIGH</a> ) Higher notification importance: shows everywhere, makes noise and peeks.
int	<b>IMPORTANCE_LOW</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_LOW">https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_LOW</a> ) Low notification importance: shows everywhere, but is not intrusive.
int	<b>IMPORTANCE_MAX</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_MAX">https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_MAX</a> ) Unused.
int	<b>IMPORTANCE_MIN</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_MIN">https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_MIN</a> ) Min notification importance: only shows in the shade, below the fold.
int	<b>IMPORTANCE_NONE</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_NONE">https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_NONE</a> ) A notification with no importance: does not show in the shade.
int	<b>IMPORTANCE_UNSPECIFIED</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_UNSPECIFIED">https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_UNSPECIFIED</a> ) Value signifying that the user has not expressed an importance.
int	<b>INTERRUPTION_FILTER_ALARMS</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALARMS">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALARMS</a> ) <b>Interruption filter</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()">https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()</a> ) only interruption filter - all notifications except those of category <b>CATEGORY_ALARM</b> ( <a href="https://developer.android.com/reference/android/app/Notification.html#CATEGORY_ALARM">https://developer.android.com/reference/android/app/Notification.html#CATEGORY_ALARM</a> ) are suppressed.
int	<b>INTERRUPTION_FILTER_ALL</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALL">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALL</a> ) <b>Interruption filter</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()">https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()</a> ) interruption filter - no notifications are suppressed.
int	<b>INTERRUPTION_FILTER_NONE</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_NONE">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_NONE</a> ) <b>Interruption filter</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()">https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()</a> ) interruptions filter - all notifications are suppressed and all audio streams (except those used for phone calls) : muted.
int	<b>INTERRUPTION_FILTER_PRIORITY</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_PRIORITY">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_PRIORITY</a> ) <b>Interruption filter</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()">https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()</a> ) interruption filter - all notifications are suppressed except those that match the priority criteria.
int	<b>INTERRUPTION_FILTER_UNKNOWN</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_UNKNOWN">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_UNKNOWN</a> ) <b>Interruption filter</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()">https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter()</a> ) returned when the value is unavailable for any reason.

Public methods	
<b>String</b> ( <a href="https://developer.android.com/reference/java/lang/String.html">https://developer.android.com/reference/java/lang/String.html</a> )	<b>addAutomaticZenRule</b> ( <a href="https://developer.android.com/reference/android/app/AutomaticZenRule.html">https://developer.android.com/reference/android/app/AutomaticZenRule.html</a> ) Creates the given zen rule.
boolean	<b>areNotificationsEnabled</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#areNotificationsEnabled">https://developer.android.com/reference/android/app/NotificationManager.html#areNotificationsEnabled</a> ) Returns whether notifications from the calling package are blocked.
void	<b>cancel</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#cancel(int)">https://developer.android.com/reference/android/app/NotificationManager.html#cancel(int)</a> ) Cancel a previously shown notification.
void	<b>cancel</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#cancel(int,java.lang.String)">https://developer.android.com/reference/android/app/NotificationManager.html#cancel(int,java.lang.String)</a> <b>tag</b> , <b>int</b> ) Cancel a previously shown notification.
void	<b>cancelAll</b> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#cancelAll">https://developer.android.com/reference/android/app/NotificationManager.html#cancelAll</a> ) Cancel all notifications from the calling package.



	<code>notification)</code> Post a notification to be shown in the status bar.
<code>boolean</code>	<code>removeAutomaticZenRule</code> ( <a href="https://developer.android.com/reference/android.app.NotificationManager.removeAutomaticZenRule">https://developer.android.com/reference/android.app.NotificationManager.removeAutomaticZenRule</a> ) ( <a href="https://developer.android.com/reference/java/lang/String.html">https://developer.android.com/reference/java/lang/String.html</a> ) <code>id</code> ) Deletes the automatic zen rule with the given id.
<code>final void</code>	<code>setInterruptionFilter</code> ( <a href="https://developer.android.com/reference/android.app.NotificationManager.setInterruptionFilter">https://developer.android.com/reference/android.app.NotificationManager.setInterruptionFilter</a> ) Sets the current notification interruption filter.
<code>void</code>	<code>setNotificationPolicy</code> ( <a href="https://developer.android.com/reference/android.app.NotificationManager.setNotificationPolicy">https://developer.android.com/reference/android.app.NotificationManager.setNotificationPolicy</a> ) ( <code>NotificationManager.Policy</code> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.Policy">https://developer.android.com/reference/android/app/NotificationManager.Policy</a> ) <code>policy</code> ) Sets the current notification policy.
<code>boolean</code>	<code>updateAutomaticZenRule</code> ( <a href="https://developer.android.com/reference/android.app.NotificationManager.updateAutomaticZenRule">https://developer.android.com/reference/android.app.NotificationManager.updateAutomaticZenRule</a> ) ( <a href="https://developer.android.com/reference/java/lang/String.html">https://developer.android.com/reference/java/lang/String.html</a> ) <code>id</code> , <code>automaticZenRule</code> ) Updates the given zen rule.

Inherited methods
▼ (#)From class <code>java.lang.Object</code> ( <a href="https://developer.android.com/reference/java/lang/Object.html">https://developer.android.com/reference/java/lang/Object.html</a> )

## Constants

### ACTION\_INTERRUPTION\_FILTER\_CHANGEDadded in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`String` (<https://developer.android.com/reference/java/lang/String.html>) `ACTION_INTERRUPTION_FILTER_CHANGED`

Intent that is broadcast when the state of `getCurrentInterruptionFilter()` changes. This broadcast is only sent to registered receivers.

Constant Value: "android.app.action.INTERRUPTION\_FILTER\_CHANGED"

### ACTION\_NOTIFICATION\_POLICY\_ACCESS\_GRANTED\_CHANGEDadded in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`String` (<https://developer.android.com/reference/java/lang/String.html>) `ACTION_NOTIFICATION_POLICY_ACCESS_GRANTED_CHANGED`

Intent that is broadcast when the state of `isNotificationPolicyAccessGranted()` ([https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted())) changes. This broadcast is only sent to registered receivers, and only to the apps that have changed.

Constant Value: "android.app.action.NOTIFICATION\_POLICY\_ACCESS\_GRANTED\_CHANGED"

### ACTION\_NOTIFICATION\_POLICY\_CHANGEDadded in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`String` (<https://developer.android.com/reference/java/lang/String.html>) `ACTION_NOTIFICATION_POLICY_CHANGED`

Intent that is broadcast when the state of `getNotificationPolicy()` changes. This broadcast is only sent to registered receivers.

Constant Value: "android.app.action.NOTIFICATION\_POLICY\_CHANGED"

### IMPORTANCE\_DEFAULTadded in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`int` `IMPORTANCE_DEFAULT`

Default notification importance: shows everywhere, makes noise, but does not visually intrude.

Constant Value: 3 (0x00000003)

## IMPORTANCE\_HIGH

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`int IMPORTANCE_HIGH`

Higher notification importance: shows everywhere, makes noise and peeks. May use full screen intents.

Constant Value: 4 (0x00000004)

## IMPORTANCE\_LOW

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`int IMPORTANCE_LOW`

Low notification importance: shows everywhere, but is not intrusive.

Constant Value: 2 (0x00000002)

## IMPORTANCE\_MAX

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`int IMPORTANCE_MAX`

Unused.

Constant Value: 5 (0x00000005)

## IMPORTANCE\_MIN

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`int IMPORTANCE_MIN`

Min notification importance: only shows in the shade, below the fold.

Constant Value: 1 (0x00000001)

## IMPORTANCE\_NONE

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`int IMPORTANCE_NONE`

A notification with no importance: does not show in the shade.

Constant Value: 0 (0x00000000)

## IMPORTANCE\_UNSPECIFIED

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`int IMPORTANCE_UNSPECIFIED`

Value signifying that the user has not expressed an importance. This value is for persisting preferences, and should never be associated with an actual notification.

Constant Value: -1000 (0xfffffc18)

## INTERRUPTION\_FILTER\_ALARMS

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

int INTERRUPTION\_FILTER\_ALARMS

**Interruption filter** ([https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter())) constant - Alarms only interruption filter - all notifications except those of category **CATEGORY\_ALARM** ([https://developer.android.com/reference/android/app/Notification.html#CATEGORY\\_ALARM](https://developer.android.com/reference/android/app/Notification.html#CATEGORY_ALARM)) are suppressed. Some audio streams are muted.

Constant Value: 4 (0x00000004)

## INTERRUPTION\_FILTER\_ALL

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

int INTERRUPTION\_FILTER\_ALL

**Interruption filter** ([https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter())) constant - Normal interruption filter - no notifications are suppressed.

Constant Value: 1 (0x00000001)

## INTERRUPTION\_FILTER\_NONE

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

int INTERRUPTION\_FILTER\_NONE

**Interruption filter** ([https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter())) constant - No interruptions filter - all notifications are suppressed and all audio streams (except those used for phone calls) and vibrations are muted.

Constant Value: 3 (0x00000003)

## INTERRUPTION\_FILTER\_PRIORITY

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

int INTERRUPTION\_FILTER\_PRIORITY

**Interruption filter** ([https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter())) constant - Priority interruption filter - all notifications are suppressed except those that match the priority criteria. Some audio streams are muted. See **priorityCallSenders** (<https://developer.android.com/reference/android/app/NotificationManager.Policy.html#priorityCallSenders>), **priorityCategories** (<https://developer.android.com/reference/android/app/NotificationManager.Policy.html#priorityCategories>), **priorityMessageSenders** (<https://developer.android.com/reference/android/app/NotificationManager.Policy.html#priorityMessageSenders>) to define or query this criteria. Users can additionally specify packages that can bypass this interruption filter.

Constant Value: 2 (0x00000002)

## INTERRUPTION\_FILTER\_UNKNOWN

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

int INTERRUPTION\_FILTER\_UNKNOWN

**Interruption filter** ([https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#getCurrentInterruptionFilter())) constant - returned when the value is unavailable for any reason.

Constant Value: 0 (0x00000000)

## Public methods

### addAutomaticZenRule

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

String (<https://developer.android.com/reference/java/lang/String.html>) **addAutomaticZenRule** (AutomaticZenRule (<https://developer.android.com/>

◀ ▶

Creates the given zen rule.

Throws a `SecurityException` if policy access is granted to this package. See `isNotificationPolicyAccessGranted()`

([https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted())).

Parameters	
<code>automaticZenRule</code>	AutomaticZenRule: the rule to create.
Returns	
<code>String</code> ( <a href="https://developer.android.com/reference/java/lang/String.html">https://developer.android.com/reference/java/lang/String.html</a> )	The id of the newly created rule; null if the rule could not be created.

areNotificationsEnabled

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`boolean areNotificationsEnabled ()`

Returns whether notifications from the calling package are blocked.

Returns	
<code>boolean</code>	

cancel

added in API level 1 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`void cancel (int id)`

Cancel a previously shown notification. If it's transient, the view will be hidden. If it's persistent, it will be removed from the status bar.

Parameters	
<code>id</code>	<code>int</code>

cancel

added in API level 5 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`void cancel (String (https://developer.android.com/reference/java/lang/String.html) tag,  
int id)`

Cancel a previously shown notification. If it's transient, the view will be hidden. If it's persistent, it will be removed from the status bar.

Parameters	
<code>tag</code>	<code>String</code>
<code>id</code>	<code>int</code>

cancelAll

added in API level 1 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`void cancelAll ()`

Cancel all previously shown notifications. See `cancel(int)` ([https://developer.android.com/reference/android/app/NotificationManager.html#cancel\(int\)](https://developer.android.com/reference/android/app/NotificationManager.html#cancel(int))) for the detailed behavior.

createNotificationChannel

added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`void createNotificationChannel (NotificationChannel (https://developer.android.com/reference/android/app/NotificationChannel.html) channel)`

Creates a notification channel that notifications can be posted to. This can also be used to restore a deleted channel and to update an existing channel's name and description.

The name and description should only be changed if the locale changes or in response to the user renaming this channel. For example, if a user has a channel named 'John Doe' that represents messages from a 'John Doe', and 'John Doe' changes his name to 'John Smith,' the channel can be renamed to match. All other fields are ignored for channels that already exist.

Parameters	
channel	NotificationChannel: the channel to create. Note that the created channel may differ from this value. If the provided channel is malformed, a RemoteException will be thrown. This value must never be null.

createNotificationChannelGroup

added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
void createNotificationChannelGroup (NotificationChannelGroup (https://developer.android.com/reference/android/app/NotificationChannelGroup)
```

Creates a group container for NotificationChannel (<https://developer.android.com/reference/android/app/NotificationChannel.html>) objects. This can be used to rename an existing group.

Group information is only used for presentation, not for behavior. Groups are optional for channels, and you can have a mix of channels that belong to groups and channels that do not.

For example, if your application supports multiple accounts, and those accounts will have similar channels, you can create a group for each account with account specific labels instead of appending account information to each channel's label.

Parameters	
group	NotificationChannelGroup: The group to create This value must never be null.

createNotificationChannelGroups

added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
void createNotificationChannelGroups (List (https://developer.android.com/reference/java/util/List.html)<NotificationChannelGroup (https://developer.android.com/reference/android/app/NotificationChannelGroup.html)
```

Creates multiple notification channel groups.

Parameters	
groups	List: The list of groups to create This value must never be null.

createNotificationChannels

added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
void createNotificationChannels (List (https://developer.android.com/reference/java/util/List.html)<NotificationChannel (https://developer.android.com/reference/android/app/NotificationChannel.html)
```

Creates multiple notification channels that different notifications can be posted to. See createNotificationChannel(NotificationChannel) ([https://developer.android.com/reference/android/app/NotificationManager.html#createNotificationChannel\(android.app.NotificationChannel\)](https://developer.android.com/reference/android/app/NotificationManager.html#createNotificationChannel(android.app.NotificationChannel))).

Parameters	
channels	List: the list of channels to attempt to create. This value must never be null.

deleteNotificationChannel

added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
void deleteNotificationChannel (String (https://developer.android.com/reference/java/lang/String.html) channelId)
```



Deletes the given notification channel.

If you create

([https://developer.android.com/reference/android/app/NotificationManager.html#createNotificationChannel\(android.app.NotificationChannel\)](https://developer.android.com/reference/android/app/NotificationManager.html#createNotificationChannel(android.app.NotificationChannel))) a new channel with this same id, the deleted channel will be un-deleted with all of the same settings it had before it was deleted.

Parameters	
channelId	String

## deleteNotificationChannelGroup

added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

void deleteNotificationChannelGroup (String (<https://developer.android.com/reference/java/lang/String.html>) groupId)

Deletes the given notification channel group, and all notification channels that belong to it.

Parameters	
groupId	String

## getActiveNotifications

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

StatusBarNotification[] (<https://developer.android.com/reference/android/service/notification/StatusBarNotification.html>) getActiveNotificat

Recover a list of active notifications: ones that have been posted by the calling app that have not yet been dismissed by the user or `cancel(String, int)` ([https://developer.android.com/reference/android/app/NotificationManager.html#cancel\(java.lang.String, int\)](https://developer.android.com/reference/android/app/NotificationManager.html#cancel(java.lang.String, int)))ed by the app. Each notification is embedded in a `StatusBarNotification` (<https://developer.android.com/reference/android/service/notification/StatusBarNotification.html>) object, including the original `tag` and `id` supplied to `notify()` ([https://developer.android.com/reference/android/app/NotificationManager.html#notify\(java.lang.String, int, android.app.Notification\)](https://developer.android.com/reference/android/app/NotificationManager.html#notify(java.lang.String, int, android.app.Notification))) (via `getTag()` ([https://developer.android.com/reference/android/service/notification/StatusBarNotification.html#getTag\(\)](https://developer.android.com/reference/android/service/notification/StatusBarNotification.html#getTag())) and `getId()` ([https://developer.android.com/reference/android/service/notification/StatusBarNotification.html#getId\(\)](https://developer.android.com/reference/android/service/notification/StatusBarNotification.html#getId()))) as well as a copy of the original `Notification` (<https://developer.android.com/reference/android/app/Notification.html>) object (via `getNotification()` ([https://developer.android.com/reference/android/service/notification/StatusBarNotification.html#getNotification\(\)](https://developer.android.com/reference/android/service/notification/StatusBarNotification.html#getNotification()))).

Returns	
StatusBarNotification[] ( <a href="https://developer.android.com/reference/android/service/notification/StatusBarNotification.html">https://developer.android.com/reference/android/service/notification/StatusBarNotification.html</a> )	An array of StatusBarNotification ( <a href="https://developer.android.com/reference/android/service/notification/StatusBarNotification.html">https://developer.android.com/reference/android/service/notification/StatusBarNotification.html</a> )

## getAutomaticZenRule

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

AutomaticZenRule (<https://developer.android.com/reference/android/app/AutomaticZenRule.html>) getAutomaticZenRule (String (<https://developer.android.com/reference/java/lang/String.html>) id)

Returns the AutomaticZenRule with the given id, if it exists and the caller has access.

Throws a SecurityException if policy access is granted to this package. See `isNotificationPolicyAccessGranted()` ([https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted())).

Returns null if there are no zen rules that match the given id, or if the calling package doesn't own the matching rule. See `getOwner()` ([https://developer.android.com/reference/android/app/AutomaticZenRule.html#getOwner\(\)](https://developer.android.com/reference/android/app/AutomaticZenRule.html#getOwner())).

Parameters	
id	String

Returns	
AutomaticZenRule ( <a href="https://developer.android.com/reference/android/app/AutomaticZenRule.html">https://developer.android.com/reference/android/app/AutomaticZenRule.html</a> )	

getAutomaticZenRules

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

Map (https://developer.android.com/reference/java/util/Map.html)<String (https://developer.android.com/reference/java/lang/String.html), AutomaticZenRule (https://developer.android.com/reference/android/app/AutomaticZenRule.html)>

Returns AutomaticZenRules owned by the caller.

Throws a SecurityException if policy access is granted to this package. See isNotificationPolicyAccessGranted() (https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted()).

Returns	
Map (https://developer.android.com/reference/java/util/Map.html)<String (https://developer.android.com/reference/java/lang/String.html), AutomaticZenRule (https://developer.android.com/reference/android/app/AutomaticZenRule.html)>	

getCurrentInterruptionFilter

added in API level 23 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

int getCurrentInterruptionFilter ()

Gets the current notification interruption filter.

The interruption filter defines which notifications are allowed to interrupt the user (e.g. via sound & vibration) and is applied globally.

Returns	
int	Value is INTERRUPTION_FILTER_NONE (https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_NONE), INTERRUPTION_FILTER_PRIORITY (https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_PRIORITY), INTERRUPTION_FILTER_ALARMS (https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALARMS), INTERRUPTION_FILTER_ALL (https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALL) OR INTERRUPTION_FILTER_UNKNOWN (https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_UNKNOWN).

getImportance

added in API level 24 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

int getImportance ()

Returns the user specified importance for notifications from the calling package.

Returns	
int	Value is IMPORTANCE_UNSPECIFIED (https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_UNSPECIFIED), IMPORTANCE_NONE (https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_NONE), IMPORTANCE_MIN (https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_MIN), IMPORTANCE_LOW (https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_LOW), IMPORTANCE_DEFAULT (https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_DEFAULT) OR IMPORTANCE_HIGH (https://developer.android.com/reference/android/app/NotificationManager.html#IMPORTANCE_HIGH).

getNotificationChannel

added in API level 26 (https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels)

NotificationChannel (https://developer.android.com/reference/android/app/NotificationChannel.html) getNotificationChannel (String (https://developer.android.com/reference/java/lang/String.html))

Returns the notification channel settings for a given channel id. The channel must belong to your package, or it will not be returned.

Parameters	
channelId	String

Returns	
<code>NotificationChannel</code> ( <a href="https://developer.android.com/reference/android/app/NotificationChannel.html">https://developer.android.com/reference/android/app/NotificationChannel.html</a> )	

getNotificationChannelGroups

added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

List (<https://developer.android.com/reference/java/util/List.html>)<NotificationChannelGroup (<https://developer.android.com/reference/android/app/NotificationChannelGroup.html>)>

Returns all notification channel groups belonging to the calling app.

Returns	
<code>List</code> ( <a href="https://developer.android.com/reference/java/util/List.html">https://developer.android.com/reference/java/util/List.html</a> ) <NotificationChannelGroup ( <a href="https://developer.android.com/reference/android/app/NotificationChannelGroup.html">https://developer.android.com/reference/android/app/NotificationChannelGroup.html</a> )>	

getNotificationChannels

added in API level 26 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

List (<https://developer.android.com/reference/java/util/List.html>)<NotificationChannel (<https://developer.android.com/reference/android/app/NotificationChannel.html>)>

Returns all notification channels belonging to the calling package.

Returns	
<code>List</code> ( <a href="https://developer.android.com/reference/java/util/List.html">https://developer.android.com/reference/java/util/List.html</a> ) <NotificationChannel ( <a href="https://developer.android.com/reference/android/app/NotificationChannel.html">https://developer.android.com/reference/android/app/NotificationChannel.html</a> )>	

getNotificationPolicy

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

NotificationManager.Policy (<https://developer.android.com/reference/android/app/NotificationManager.Policy.html>) getNotificationPolicy ()

Gets the current notification policy.

Only available if policy access is granted to this package. See `isNotificationPolicyAccessGranted()`  
([https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted())).

Returns	
<code>NotificationManager.Policy</code> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.Policy.html">https://developer.android.com/reference/android/app/NotificationManager.Policy.html</a> )	

isNotificationPolicyAccessGranted

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

boolean isNotificationPolicyAccessGranted ()

Checks the ability to read/modify notification policy for the calling package.

Returns true if the calling package can read/modify notification policy.

Request policy access by sending the user to the activity that matches the system intent action `ACTION_NOTIFICATION_POLICY_ACCESS_SETTINGS`  
([https://developer.android.com/reference/android/provider/Settings.html#ACTION\\_NOTIFICATION\\_POLICY\\_ACCESS\\_SETTINGS](https://developer.android.com/reference/android/provider/Settings.html#ACTION_NOTIFICATION_POLICY_ACCESS_SETTINGS)).

Use `ACTION_NOTIFICATION_POLICY_ACCESS_GRANTED_CHANGED`  
([https://developer.android.com/reference/android/app/NotificationManager.html#ACTION\\_NOTIFICATION\\_POLICY\\_ACCESS\\_GRANTED\\_CHANGED](https://developer.android.com/reference/android/app/NotificationManager.html#ACTION_NOTIFICATION_POLICY_ACCESS_GRANTED_CHANGED)) to listen for user grant or denial of this access.

--

Returns	
<code>boolean</code>	

notify

added in API level 1 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
void notify (int id,
            Notification (https://developer.android.com/reference/android/app/Notification.html) notification)
```

Post a notification to be shown in the status bar. If a notification with the same id has already been posted by your application and has not yet been canceled, it will be replaced by the updated information.

Parameters	
<code>id</code>	<code>int</code> : An identifier for this notification unique within your application.
<code>notification</code>	<code>Notification</code> : A <code>Notification</code> ( <a href="https://developer.android.com/reference/android/app/Notification.html">https://developer.android.com/reference/android/app/Notification.html</a> ) object describing what to show the user. Must not be null.

notify

added in API level 5 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
void notify (String (https://developer.android.com/reference/java/lang/String.html) tag,
            int id,
            Notification (https://developer.android.com/reference/android/app/Notification.html) notification)
```

Post a notification to be shown in the status bar. If a notification with the same tag and id has already been posted by your application and has not yet been canceled, it will be replaced by the updated information.

Parameters	
<code>tag</code>	<code>String</code> : A string identifier for this notification. May be <code>null</code> .
<code>id</code>	<code>int</code> : An identifier for this notification. The pair (tag, id) must be unique within your application.
<code>notification</code>	<code>Notification</code> : A <code>Notification</code> ( <a href="https://developer.android.com/reference/android/app/Notification.html">https://developer.android.com/reference/android/app/Notification.html</a> ) object describing what to show the user. Must not be null.

removeAutomaticZenRule

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
boolean removeAutomaticZenRule (String (https://developer.android.com/reference/java/lang/String.html) id)
```

Deletes the automatic zen rule with the given id.

Throws a `SecurityException` if policy access is granted to this package. See `isNotificationPolicyAccessGranted()` ([https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted())).

Callers can only delete rules that they own. See `getOwner()` ([https://developer.android.com/reference/android/app/AutomaticZenRule.html#getOwner\(\)](https://developer.android.com/reference/android/app/AutomaticZenRule.html#getOwner())).

Parameters	
<code>id</code>	<code>String</code> : the id of the rule to delete.

Returns	
<code>boolean</code>	Whether the rule was successfully deleted.

setInterruptionFilter

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

```
void setInterruptionFilter (int interruptionFilter)
```

Sets the current notification interruption filter.

The interruption filter defines which notifications are allowed to interrupt the user (e.g. via sound & vibration) and is applied globally.

Only available if policy access is granted to this package. See `isNotificationPolicyAccessGranted()`

([https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted())).

Parameters	
<code>interruptionFilter</code>	<code>int</code> Value is <code>INTERRUPTION_FILTER_NONE</code> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_NONE">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_NONE</a> ), <code>INTERRUPTION_FILTER_PRIORITY</code> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_PRIORITY">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_PRIORITY</a> ), <code>INTERRUPTION_FILTER_ALARMS</code> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALARMS">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALARMS</a> ), <code>INTERRUPTION_FILTER_ALL</code> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALL">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_ALL</a> ) OR <code>INTERRUPTION_FILTER_UNKNOWN</code> ( <a href="https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_UNKNOWN">https://developer.android.com/reference/android/app/NotificationManager.html#INTERRUPTION_FILTER_UNKNOWN</a> ).

## setNotificationPolicy

added in API level 23 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`void setNotificationPolicy (NotificationManager.Policy (https://developer.android.com/reference/android/app/NotificationManager.Policy.html)`

Sets the current notification policy.

Only available if policy access is granted to this package. See `isNotificationPolicyAccessGranted()`

([https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted())).

Parameters	
<code>policy</code>	<code>NotificationManager.Policy</code> : The new desired policy. This value must never be <code>null</code> .

## updateAutomaticZenRule

added in API level 24 (<https://developer.android.com/guide/topics/manifest/uses-sdk-element.html#ApiLevels>)

`boolean updateAutomaticZenRule (String (https://developer.android.com/reference/java/lang/String.html) id,`  
`AutomaticZenRule (https://developer.android.com/reference/android/app/AutomaticZenRule.html) automaticZenRule)`

Updates the given zen rule.

Throws a `SecurityException` if policy access is granted to this package. See `isNotificationPolicyAccessGranted()`

([https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted\(\)](https://developer.android.com/reference/android/app/NotificationManager.html#isNotificationPolicyAccessGranted())).

Callers can only update rules that they own. See `getOwner()` ([https://developer.android.com/reference/android/app/AutomaticZenRule.html#getOwner\(\)](https://developer.android.com/reference/android/app/AutomaticZenRule.html#getOwner())).

Parameters	
<code>id</code>	<code>String</code> : The id of the rule to update
<code>automaticZenRule</code>	<code>AutomaticZenRule</code> : the rule to update.

Returns	
<code>boolean</code>	Whether the rule was successfully updated.



