 chrome

CHROME   **CHROME OS**   **CHROME APIS**

chrome.notifications

Description:	Use the `chrome.notifications` API to create rich notifications using templates and show these notifications to users in the system tray.
Availability:	Since Chrome 28.
Permissions:	`"notifications"`
Learn More:	**Rich Notifications** **Keep Users Informed** **Chrome Apps Office Hours: Rich Notifications**

Summary

Types
TemplateType
PermissionLevel
NotificationOptions
Methods
create – `chrome.notifications.create(`**string** `notificationId,` **NotificationOptions** `options,` **function** `callback)`
update – `chrome.notifications.update(`**string** `notificationId,` **NotificationOptions** `options,` **function** `callback)`
clear – `chrome.notifications.clear(`**string** `notificationId,` **function** `callback)`
getAll – `chrome.notifications.getAll(`**function** `callback)`
getPermissionLevel – `chrome.notifications.getPermissionLevel(`**function** `callback)`
Events
onClosed
onClicked
onButtonClicked
onPermissionLevelChanged
onShowSettings

Types

TemplateType

Enum
`"basic"`
icon, title, message, expandedMessage, up to two buttons
`"image"`
icon, title, message, expandedMessage, image, up to two buttons
`"list"`
icon, title, message, items, up to two buttons. Users on Mac OS X only see the first item.
`"progress"`
icon, title, message, progress, up to two buttons

PermissionLevel

Enum
`"granted"`
User has elected to show notifications from the app or extension. This is the default at install time.
`"denied"`
User has elected not to show notifications from the app or extension.

NotificationOptions

properties		
TemplateType	(optional) type	Which type of notification to display. *Required for* **notifications.create** method.
string	(optional) iconUrl	A URL to the sender's avatar, app icon, or a thumbnail for image notifications. URLs can be a data URL, a blob URL, or a URL relative to a resource within this extension's .crx file *Required*

		*for **notifications.create** method.
string	(optional) appIconMaskUrl	**Deprecated** since Chrome 59. The app icon mask is not visible for Mac OS X users. A URL to the app icon mask. URLs have the same restrictions as **iconUrl**. The app icon mask should be in alpha channel, as only the alpha channel of the image will be considered.
string	(optional) title	Title of the notification (e.g. sender name for email). *Required for **notifications.create** method.*
string	(optional) message	Main notification content. *Required for **notifications.create** method.*
string	(optional) contextMessage	Since Chrome 31. Alternate notification content with a lower-weight font.
integer	(optional) priority	Priority ranges from -2 to 2. -2 is lowest priority. 2 is highest. Zero is default. On platforms that don't support a notification center (Windows, Linux & Mac), -2 and -1 result in an error as notifications with those priorities will not be shown at all.
double	(optional) eventTime	A timestamp associated with the notification, in milliseconds past the epoch (e.g. `Date.now() + n`).
array of object	(optional) buttons	Text and icons for up to two notification action buttons. **Properties of each object** <table><tr><td>string</td><td>title</td><td></td></tr><tr><td>string</td><td>(optional) iconUrl</td><td></td></tr></table>
string	(optional) imageUrl	**Deprecated** since Chrome 59. The image is not visible for Mac OS X users. A URL to the image thumbnail for image-type notifications. URLs have the same restrictions as

		iconUrl.
array of object	(optional) items	Items for multi-item notifications. Users on Mac OS X only see the first item.

Properties of each object

string	title	Title of one item of a list notification.
string	message	Additional details about this item.

integer	(optional) progress	Since Chrome 30. Current progress ranges from 0 to 100.
boolean	(optional) isClickable	**Deprecated** since Chrome 67. This UI hint is ignored as of Chrome 67
boolean	(optional) requireInteraction	Since Chrome 50. Indicates that the notification should remain visible on screen until the user activates or dismisses the notification. This defaults to false.
boolean	(optional) silent	Since Chrome 70. Indicates that no sounds or vibrations should be made when the notification is being shown. This defaults to false.

Methods

create

```
chrome.notifications.create(string notificationId, NotificationOptions
options, function callback)
```

Creates and displays a notification.

Parameters		
string	(optional) notificationId	Identifier of the notification. If not set or empty, an ID will automatically be generated. If it matches an existing notification, this method first clears that notification before proceeding with the create operation. The identifier may not be longer than 500 characters. The `notificationId` parameter is required before Chrome 42.
NotificationOptions	options	Contents of the notification.
function	(optional) callback	Returns the notification id (either supplied or generated) that represents the created notification. The callback is required before Chrome 42. If you specify the *callback* parameter, it should be a function that looks like this: `function(string notificationId) {...};`

string	notificationId	

update

`chrome.notifications.update(string notificationId, NotificationOptions options, function callback)`

Updates an existing notification.

Parameters		
string	notificationId	The id of the notification to be updated. This is returned by **notifications.create** method.
NotificationOptions	options	Contents of the notification to update to.
function	(optional) callback	Called to indicate whether a matching notification existed. The callback is required before Chrome 42.

If you specify the *callback* parameter, it should be a function that looks like this:

```
function(boolean wasUpdated) {...};
```

boolean	wasUpdated	

clear

```
chrome.notifications.clear(string notificationId, function callback)
```

Clears the specified notification.

Parameters		
string	notificationId	The id of the notification to be cleared. This is returned by **notifications.create** method.
function	(optional) callback	Called to indicate whether a matching notification existed. The callback is required before Chrome 42. If you specify the *callback* parameter, it should be a function that looks like this: `function(boolean wasCleared) {...};` boolean · wasCleared

getAll

```
chrome.notifications.getAll(function callback)
```

Since Chrome 29.

Retrieves all the notifications.

Parameters		
function	callback	Returns the set of notification_ids currently in the system. The *callback* parameter should be a function that looks like this: `function(object notifications) {...};` object · notifications

getPermissionLevel

`chrome.notifications.getPermissionLevel(`**`function`**` callback)`

Since Chrome 32.

Retrieves whether the user has enabled notifications from this app or extension.

Parameters		
function	callback	Returns the current permission level. The *callback* parameter should be a function that looks like this: `function( `**`PermissionLevel`**` level) {...};` <table><tr><td>**PermissionLevel**</td><td>level</td><td></td></tr></table>

Events

onClosed

The notification closed, either by the system or by user action.

addListener
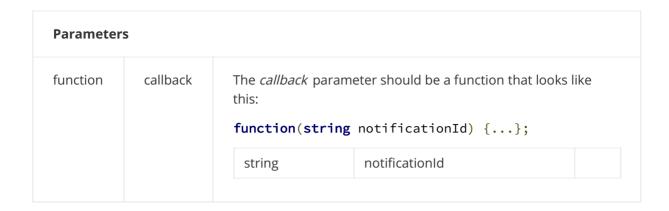
`chrome.notifications.onClosed.addListener(`**`function`**` callback)`

Parameters		
function	callback	The *callback* parameter should be a function that looks like this: `function(`**`string`**` notificationId, `**`boolean`**` byUser) {...};` <table><tr><td>string</td><td>notificationId</td><td></td></tr><tr><td>boolean</td><td>byUser</td><td></td></tr></table>

onClicked

The user clicked in a non-button area of the notification.

addListener

`chrome.notifications.onClicked.addListener(function callback)`

Parameters		
function	callback	The *callback* parameter should be a function that looks like this: `function(string notificationId) {...};`

string	notificationId	

onButtonClicked

The user pressed a button in the notification.

addListener

`chrome.notifications.onButtonClicked.addListener(function callback)`

Parameters		
function	callback	The *callback* parameter should be a function that looks like this: `function(string notificationId, integer buttonIndex) {...};`

string	notificationId	
integer	buttonIndex	

onPermissionLevelChanged

Since Chrome 32.

The user changes the permission level. As of Chrome 47, only ChromeOS has UI that dispatches this event.

addListener

`chrome.notifications.onPermissionLevelChanged.addListener(function callback)`

Parameters

function	callback	The *callback* parameter should be a function that looks like this: `function(` `PermissionLevel` `level) {...};`
			PermissionLevel	level	

onShowSettings

Deprecated since Chrome 65. Custom notification settings button is no longer supported.

The user clicked on a link for the app's notification settings. As of Chrome 47, only ChromeOS has UI that dispatches this event. As of Chrome 65, that UI has been removed from ChromeOS, too.

addListener
`chrome.notifications.onShowSettings.addListener(`**`function`** `callback)`

Parameters		
function	callback	The *callback* parameter should be a function that looks like this: `function() {...};`

Google   **Terms of Service**   **Privacy Policy**   **Report a content bug**   G+ **Add us on** G+