

[Technologies](#) ▼[References & Guides](#) ▼[Feedback](#) ▼[Sign in](#)  Search

# Notification

**Note:** This feature is available in [Web Workers](#).

## Secure context

This feature is available only in [secure contexts](#) (HTTPS), in some or all [supporting](#) browsers.

The `Notification` interface of the [Notifications API](#) is used to configure and display desktop notifications to the user. These notifications' appearance and specific functionality vary across platforms but generally they provide a way to asynchronously provide information to the user.

## Constructor

### **Notification()**

Creates a new instance of the `Notification` object.

---

## Properties

### Static properties

These properties are available only on the `Notification` object itself.

#### **Notification.permission** | Read only

A string representing the current permission to display notifications. Possible values are:

- `denied` — The user refuses to have notifications displayed.
- `granted` — The user accepts having notifications displayed.
- `default` — The user choice is unknown and therefore the browser will act as if the value were denied.

### Instance properties

These properties are available only on instances of the `Notification` object.

#### **Notification.actions** | Read only

The actions array of the notification as specified in the constructor's `options` parameter.

#### **Notification.badge** | Read only

The URL of the image used to represent the notification when there is not enough space to display the notification itself.

#### **Notification.body** | Read only

The body string of the notification as specified in the constructor's `options` parameter.

#### **Notification.data** | Read only

Returns a structured clone of the notification's data.

#### **Notification.dir** | Read only

The text direction of the notification as specified in the constructor's `options` parameter.

#### **Notification.lang** | Read only

The language code of the notification as specified in the constructor's `options` parameter.

#### **Notification.tag** | Read only

The ID of the notification (if any) as specified in the constructor's `options` parameter.

#### **Notification.icon** | Read only

The URL of the image used as an icon of the notification as specified in the constructor's `options` parameter.

#### **Notification.image** | Read only

The URL of an image to be displayed as part of the notification, as specified in the constructor's `options` parameter.

#### **Notification.renotify** | Read only

Specifies whether the user should be notified after a new notification replaces an old one.

**Notification.requireInteraction** | Read only

A Boolean indicating that a notification should remain active until the user clicks or dismisses it, rather than closing automatically.

**Notification.silent** | Read only

Specifies whether the notification should be silent — i.e., no sounds or vibrations should be issued, regardless of the device settings.

**Notification.timestamp** | Read only

Specifies the time at which a notification is created or applicable (past, present, or future).

**Notification.title** | Read only

The title of the notification as specified in the first parameter of the constructor.

**Notification.vibrate** | Read only

Specifies a vibration pattern for devices with vibration hardware to emit.

## Event handlers

**Notification.onclick**

A handler for the `click` event. It is triggered each time the user clicks on the notification.

**Notification.onclose**

A handler for the `close` event. It is triggered when the user closes the notification.

**Notification.onerror**

A handler for the `error` event. It is triggered each time the notification encounters an error.

**Notification.onshow**

A handler for the `show` event. It is triggered when the notification is displayed.

---

## Methods [🔗](#)

### Static methods [🔗](#)

These methods are available only on the `Notification` object itself.

#### **`Notification.requestPermission()`**

Requests permission from the user to display notifications.

### Instance methods [🔗](#)

These properties are available only on an instance of the `Notification` object or through its prototype. The `Notification` object also inherits from the `EventTarget` interface.

#### **`Notification.close()`**

Programmatically closes a notification.

---

## Example [🔗](#)

Assume this basic HTML:

```
1 | <button onclick="notifyMe()">Notify me!</button>
```

It's possible to send a notification as follows — here we present a fairly verbose and complete set of code you could use if you wanted to first check whether notifications are supported, then check if permission has been granted for the current origin to send notifications, then request permission if required, before then sending a notification.

```
1 function notifyMe() {
2   // Let's check if the browser supports notifications
3   if (!("Notification" in window)) {
4     alert("This browser does not support desktop notification");
5   }
6
7   // Let's check whether notification permissions have already been granted
8   else if (Notification.permission === "granted") {
9     // If it's okay let's create a notification
10    var notification = new Notification("Hi there!");
11  }
12
13  // Otherwise, we need to ask the user for permission
14  else if (Notification.permission !== "denied") {
15    Notification.requestPermission().then(function (permission) {
16      // If the user accepts, let's create a notification
17      if (permission === "granted") {
18        var notification = new Notification("Hi there!");
19      }
20    });
21  }
22 }
```

```
23 // At last, if the user has denied notifications, and you
24 // want to be respectful there is no need to bother them any more.
25 }
```

Notify me!

### Alternate example: run on page load [↗](#)

In many cases, you don't need to be this verbose. For example, in our [Emogotchi demo](#) (see source code), we simply run `Notification.requestPermission` regardless to make sure we can get permission to send notifications (this uses the newer promise-based method syntax):

```
1 Notification.requestPermission().then(function(result) {
2   console.log(result);
3 });
```

Then we run a simple `spawnNotification()` function when we want to fire a notification — this is passed arguments to specify the body, icon, and title we want. Then it creates the necessary `options` object and fires the notification using the `Notification()` constructor.

```
1 function spawnNotification(body, icon, title) {
2   var options = {
3     body: body,
4     icon: icon
```

```
5   };
6   var n = new Notification(title, options);
7 }
```

## Specifications [↗](#)

Specification	Status	Comment
Notifications API	<a href="#">LS</a> Living Standard	Living standard

## Browser compatibility [↗](#)

[Update compatibility data on GitHub](#)

Notification	
Chrome	22
Edge	14
Firefox	22
IE	No



Opera	25
Safari	6
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	22
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### Available in workers

Chrome	45
Edge	Yes
Firefox	41
IE	No
Opera	32
Safari	?
WebView Android	No
Chrome Android	45
Edge Mobile	Yes

Firefox Android	41
Opera Android	32
Safari iOS	No
Samsung Internet Android	?

#### Secure context required

Chrome	62
Edge	?
Firefox	67
IE	No
Opera	49
Safari	?
WebView Android	No
Chrome Android	62
Edge Mobile	?
Firefox Android	67
Opera Android	46
Safari iOS	No
Samsung Internet Android	?

#### Notification() constructor

	22
--	----

Chrome	
Edge	Yes
Firefox	22
IE	No
Opera	25
Safari	6
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	22
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	?

#### actions

Chrome	53
Edge	18
Firefox	No
IE	No
Opera	39

Safari	?
WebView Android	No
Chrome Android	53
Edge Mobile	No
Firefox Android	No
Opera Android	41
Safari iOS	No
Samsung Internet Android	Yes

#### badge

Chrome	53
Edge	18
Firefox	No
IE	No
Opera	39
Safari	?
WebView Android	No
Chrome Android	53
Edge Mobile	No
Firefox Android	No
Opera Android	41

Safari iOS	No
Samsung Internet Android	Yes

## body

Chrome	Yes
Edge	14
Firefox	Yes
IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

## data

Chrome	Yes
Edge	16
	Yes

IE	No
Opera	Yes
Safari	?
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### dir

Chrome	Yes
Edge	14
Firefox	Yes
IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?

Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### icon

Chrome	22
Edge	14
Firefox	22
IE	No
Opera	25
Safari	No
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	22
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

## image

Chrome	53
Edge	18
Firefox	No
IE	No
Opera	40
Safari	?
WebView Android	No
Chrome Android	53
Edge Mobile	?
Firefox Android	No
Opera Android	41
Safari iOS	No
Samsung Internet Android	Yes

## lang

Chrome	Yes
Edge	14
Firefox	Yes
IE	No
Opera	Yes



Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

**maxActions**

Chrome	Yes
Edge	18
Firefox	No
IE	No
Opera	Yes
Safari	?
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	No
Opera Android	Yes

Safari iOS	No
Samsung Internet Android	Yes

`onclick`

Chrome	Yes
Edge	14
Firefox	No
IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	No
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

`onclose`

Chrome	Yes
Edge	14
	Yes

IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### onerror

Chrome	Yes
Edge	14
Firefox	No
IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?

Firefox Android	No
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### onshow

Chrome	Yes
Edge	14
Firefox	Yes
IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### permission

	Yes
--	-----

Edge	14
Firefox	Yes
IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### renotify

Chrome	50
Edge	No
Firefox	No
IE	No
Opera	37
Safari	No
WebView Android	No

Chrome Android	50
Edge Mobile	No
Firefox Android	No
Opera Android	37
Safari iOS	No
Samsung Internet Android	Yes

#### requireInteraction

Chrome	Yes
Edge	17
Firefox	No
IE	No
Opera	Yes
Safari	?
WebView Android	No
Chrome Android	Yes
Edge Mobile	No
Firefox Android	No
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

## silent

Chrome	43
Edge	17
Firefox	No
IE	No
Opera	30
Safari	No
WebView Android	No
Chrome Android	43
Edge Mobile	No
Firefox Android	No
Opera Android	30
Safari iOS	No
Samsung Internet Android	Yes

## tag

Chrome	Yes
Edge	14
Firefox	Yes
IE	No
Opera	Yes

Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### timestamp

Chrome	Yes
Edge	17
Firefox	No
IE	No
Opera	Yes
Safari	?
WebView Android	No
Chrome Android	Yes
Edge Mobile	No
Firefox Android	No
Opera Android	Yes



Safari iOS	No
Samsung Internet Android	Yes

## title

Chrome	Yes
Edge	14
Firefox	No
IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?
Firefox Android	No
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

## vibrate

Chrome	53
Edge	No
	No

IE	No
Opera	39
Safari	?
WebView Android	No
Chrome Android	53
Edge Mobile	No
Firefox Android	No
Opera Android	41
Safari iOS	No
Samsung Internet Android	Yes

close

Chrome	Yes
Edge	14
Firefox	Yes
IE	No
Opera	Yes
Safari	Yes
WebView Android	No
Chrome Android	Yes
Edge Mobile	?

Firefox Android	Yes
Opera Android	Yes
Safari iOS	No
Samsung Internet Android	Yes

#### requestPermission

Chrome	46
Edge	14
Firefox	47
IE	No
Opera	40
Safari	Yes
WebView Android	No
Chrome Android	46
Edge Mobile	?
Firefox Android	Yes
Opera Android	41
Safari iOS	No
Samsung Internet Android	Yes

Full support



No support



Compatibility unknown

See implementation notes.



Requires a vendor prefix or different name for use.

---

## See also [🔗](#)

- [Using the Notifications API](#)