Notification

Create OS desktop notifications

Process: Main

Using in the renderer process

If you want to show Notifications from a renderer process you should use the HTML5 Notification API

Class: Notification

Create OS desktop notifications

Process: Main

`Notification` is an EventEmitter.

It creates a new `Notification` with native properties as set by the `options`.

Static Methods

The `Notification` class has the following static methods:

`Notification.isSupported()` Returns `boolean` - Whether or not desktop notifications are supported on the current system

new Notification([options])

- `options` Object (optional)
  - `title` string (optional) - A title for the notification, which will be shown at the top of the notification window when it is shown.
  - `subtitle` string (optional) *macOS* - A subtitle for the notification, which will be displayed below the title.
  - `body` string (optional) - The body text of the notification, which will be displayed below the title or subtitle.
  - `silent` boolean (optional) - Whether or not to emit an OS notification noise when showing the notification.
  - `icon` (string | NativeImage) (optional) - An icon to use in the notification.
  - `hasReply` boolean (optional) *macOS* - Whether or not to add an inline reply option to the notification.
  - `timeoutType` string (optional) *Linux Windows* - The timeout duration of the notification. Can be 'default' or 'never'.
  - `replyPlaceholder` string (optional) *macOS* - The placeholder to write in the inline reply input field.
  - `sound` string (optional) *macOS* - The name of the sound file to play when the notification is shown.

- **urgency** string (optional) *Linux* - The urgency level of the notification. Can be 'normal', 'critical', or 'low'.
- **actions** NotificationAction[] (optional) *macOS* - Actions to add to the notification. Please read the available actions and limitations in the `NotificationAction` documentation.
- **closeButtonText** string (optional) *macOS* - A custom title for the close button of an alert. An empty string will cause the default localized text to be used.
- **toastXml** string (optional) *Windows* - A custom description of the Notification on Windows superseding all properties above. Provides full customization of design and behavior of the notification.

Instance Events

Objects created with `new Notification` emit the following events:

Note: Some events are only available on specific operating systems and are labeled as such.

Event: 'show'   Returns:

- **event** Event

Emitted when the notification is shown to the user, note this could be fired multiple times as a notification can be shown multiple times through the `show()` method.

Event: 'click'   Returns:

- **event** Event

Emitted when the notification is clicked by the user.

Event: 'close'   Returns:

- **event** Event

Emitted when the notification is closed by manual intervention from the user.

This event is not guaranteed to be emitted in all cases where the notification is closed.

Event: 'reply' *macOS*   Returns:

- **event** Event
- **reply** string - The string the user entered into the inline reply field.

Emitted when the user clicks the "Reply" button on a notification with `hasReply: true`.

Event: 'action' *macOS* Returns:

- `event` Event
- `index` number - The index of the action that was activated.

Event: 'failed' *Windows* Returns:

- `event` Event
- `error` string - The error encountered during execution of the `show()` method.

Emitted when an error is encountered while creating and showing the native notification.

Instance Methods

Objects created with `new Notification` have the following instance methods:

`notification.show()` Immediately shows the notification to the user, please note this means unlike the HTML5 Notification implementation, instantiating a `new Notification` does not immediately show it to the user, you need to call this method before the OS will display it.

If the notification has been shown before, this method will dismiss the previously shown notification and create a new one with identical properties.

`notification.close()` Dismisses the notification.

Instance Properties

`notification.title` A `string` property representing the title of the notification.

`notification.subtitle` A `string` property representing the subtitle of the notification.

`notification.body` A `string` property representing the body of the notification.

`notification.replyPlaceholder` A `string` property representing the reply placeholder of the notification.

`notification.sound` A `string` property representing the sound of the notification.

`notification.closeButtonText` A `string` property representing the close button text of the notification.

notification.silent   A `boolean` property representing whether the notification is silent.

notification.hasReply   A `boolean` property representing whether the notification has a reply action.

notification.urgency *Linux*   A `string` property representing the urgency level of the notification. Can be 'normal', 'critical', or 'low'.

Default is 'low' - see NotifyUrgency for more information.

notification.timeoutType *Linux Windows*   A `string` property representing the type of timeout duration for the notification. Can be 'default' or 'never'.

If `timeoutType` is set to 'never', the notification never expires. It stays open until closed by the calling API or the user.

notification.actions   A `NotificationAction[]` property representing the actions of the notification.

notification.toastXml *Windows*   A `string` property representing the custom Toast XML of the notification.

Playing Sounds

On macOS, you can specify the name of the sound you'd like to play when the notification is shown. Any of the default sounds (under System Preferences > Sound) can be used, in addition to custom sound files. Be sure that the sound file is copied under the app bundle (e.g., `YourApp.app/Contents/Resources`), or one of the following locations:

- `~/Library/Sounds`
- `/Library/Sounds`
- `/Network/Library/Sounds`
- `/System/Library/Sounds`

See the `NSSound` docs for more information.