Window: message event

- Syntax
- Event type
- Examples
- Specifications
- Browser compatibility
- See also

The `message` event is fired on a Window object when the window receives a message, for example from a call to `Window.postMessage()` from another browsing context.

This event is not cancellable and does not bubble.

Syntax

Use the event name in addEventListener():

```
addEventListener("message", (event) => { })
```

Alternatively, assign to the event handler property of the interface:

```
onmessage = (event) => { }
```

Event type

A MessageEvent. Inherits from Event.

Examples

Suppose a script sends a message to a different browsing context, such as another `<iframe>`, using code like this:

```
const targetFrame = window.top.frames[1];
const targetOrigin = "https://example.org";
const windowMessageButton = document.querySelector("#window-message");

windowMessageButton.addEventListener("click", () => {
  targetFrame.postMessage("hello there", targetOrigin);
});
```

The receiver can listen for the message using `addEventListener()` with code like this:

```
window.addEventListener("message", (event) => {
  console.log(`Received message: ${event.data}`);
});
```

Alternatively the listener could use the `onmessage` event handler property (but note that this replaces other event listeners added using `onmessage`, if any):

```
window.onmessage = (event) => {
  console.log(`Received message: ${event.data}`);
};
```

Specifications

Browser compatibility

See also

- Related events: `messageerror`.
- `Window.postMessage()`.

□□□ #3
□ □□□ □□□ 2026-10-31 03:27:53 UTC
□ cocobook-sync □□□ 2026-11-04 17:35:12 UTC