

[Home](#) > [puppeteer](#) > [Browser](#)

## Browser class

A Browser is created when Puppeteer connects to a Chromium instance, either through [PuppeteerNode.launch\(\)](#) or [Puppeteer.connect\(\)](#).

**Signature:**

```
export declare class Browser extends EventEmitter
```

**Extends:** [EventEmitter](#)

## Remarks

The Browser class extends from Puppeteer's [EventEmitter](#) class and will emit various events which are documented in the [BrowserEmittedEvents](#) enum.

The constructor for this class is marked as internal. Third-party code should not call the constructor directly or create subclasses that extend the `Browser` class.

## Example 1

An example of using a [Browser](#) to create a [Page](#):

```
const puppeteer = require('puppeteer');

(async () => {
  const browser = await puppeteer.launch();
  const page = await browser.newPage();
  await page.goto('https://example.com');
  await browser.close();
})();
```

## Example 2

An example of disconnecting from and reconnecting to a [Browser](#):

```
const puppeteer = require('puppeteer');

(async () => {
  const browser = await puppeteer.launch();
  // Store the endpoint to be able to reconnect to Chromium
  const browserWSEndpoint = browser.wsEndpoint();
  // Disconnect puppeteer from Chromium
  browser.disconnect();

  // Use the endpoint to reestablish a connection
  const browser2 = await puppeteer.connect({browserWSEndpoint});
  // Close Chromium
```

```

    await browser2.close();
  })();

```

## Methods

Method	Modifiers	Description
<a href="#">browserContexts()</a>		Returns an array of all open browser contexts. In a newly created browser, this will return a single instance of <a href="#">BrowserContext</a> .
<a href="#">close()</a>		Closes Chromium and all of its pages (if any were opened). The <a href="#">Browser</a> object itself is considered to be disposed and cannot be used anymore.
<a href="#">createIncognitoBrowserContext()</a>		Creates a new incognito browser context. This won't share cookies/cache with other browser contexts.
<a href="#">defaultBrowserContext()</a>		Returns the default browser context. The default browser context cannot be closed.
<a href="#">disconnect()</a>		Disconnects Puppeteer from the browser, but leaves the Chromium process running. After calling <code>disconnect</code> , the <a href="#">Browser</a> object is considered disposed and cannot be used anymore.
<a href="#">isConnected()</a>		Indicates that the browser is connected.
<a href="#">newPage()</a>		Promise which resolves to a new <a href="#">Page</a> object. The Page is created in a default browser context.
<a href="#">pages()</a>		An array of all open pages inside the Browser.
<a href="#">process()</a>		The spawned browser process. Returns <code>null</code> if the browser instance was created with <a href="#">Puppeteer.connect()</a> .
<a href="#">target()</a>		The target associated with the browser.
<a href="#">targets()</a>		All active targets inside the Browser. In case of multiple browser contexts, returns an array with all the targets in all browser contexts.
<a href="#">userAgent()</a>		The browser's original user agent. Pages can override the browser user agent with <a href="#">Page.setUserAgent()</a> .
<a href="#">version()</a>		A string representing the browser name and version.
<a href="#">waitForTarget(predicate, options)</a>		Searches for a target in all browser contexts.
<a href="#">wsEndpoint()</a>		The browser websocket endpoint which can be used as an argument to <a href="#">Puppeteer.connect()</a> .