

Home > puppeteer > BrowserContext

BrowserContext class

BrowserContexts provide a way to operate multiple independent browser sessions. When a browser is launched, it has a single BrowserContext used by default. The method Browser.newPage creates a page in the default browser context.

Signature:

```
export declare class BrowserContext extends EventEmitter
```

Extends: EventEmitter

Remarks

The Browser class extends from Puppeteer's EventEmitter class and will emit various events which are documented in the BrowserContextEmittedEvents enum.

If a page opens another page, e.g. with a `window.open` call, the popup will belong to the parent page's browser context.

Puppeteer allows creation of “incognito” browser contexts with Browser.createIncognitoBrowserContext method. “Incognito” browser contexts don't write any browsing data to disk.

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

Example

```
// Create a new incognito browser context
const context = await browser.createIncognitoBrowserContext();
// Create a new page inside context.
const page = await context.newPage();
// ... do stuff with page ...
await page.goto('https://example.com');
// Dispose context once it's no longer needed.
await context.close();
```

Methods

Method	Modifiers	Description
browser()		The browser this browser context belongs to.

Method	Modifiers	Description
<code>clearPermissionOverrides()</code>		Clears all permission overrides for the browser context.
<code>close()</code>		Closes the browser context. All the targets that belong to the browser context will be closed.
<code>isIncognito()</code>		Returns whether <code>BrowserContext</code> is incognito. The default browser context is the only non-incognito browser context.
<code>newPage()</code>		Creates a new page in the browser context.
<code>overridePermissions(origin, permissions)</code>		
<code>pages()</code>		An array of all pages inside the browser context.
<code>targets()</code>		An array of all active targets inside the browser context.
<code>waitForTarget(predicate, options)</code>		This searches for a target in this specific browser context.