Page

* A **Page** refers to a single browser tab or popup window within a **BrowserContext**.
* Pages are used to navigate URLs, fill forms, and interact with web content.

**Example**:

javascript

Copy code

const page = await context.newPage();

await page.goto('http://example.com');

await page.locator('#search').fill('query');

await page.locator('#submit').click();

console.log(page.url());

**Multiple Pages**

* **BrowserContexts** can manage multiple pages (tabs).
* Pages inherit context-level settings like viewport size or network conditions.

**Example**:

javascript

Copy code

const pageOne = await context.newPage();

const pageTwo = await context.newPage();

const allPages = context.pages();

**Handling New Pages**

* Use the page event to manage popups opened by links (target="\_blank").

**Example**:

javascript

Copy code

const pagePromise = context.waitForEvent('page');

await page.getByText('open new tab').click();

const newPage = await pagePromise;

await newPage.getByRole('button').click();

console.log(await newPage.title());

For unknown triggers:

javascript

Copy code

context.on('page', async page => {

await page.waitForLoadState();

console.log(await page.title());

});