PuppeteerNode class

Extends the main Puppeteer class with Node specific behaviour for fetching and downloading browsers.

If you're using Puppeteer in a Node environment, this is the class you'll get when you run `require('puppeteer')` (or the equivalent ES `import`).

Signature:

```
export declare class PuppeteerNode extends Puppeteer
```

Extends: Puppeteer

Remarks

The most common method to use is launch, which is used to launch and connect to a new browser instance.

See the main Puppeteer class for methods common to all environments, such as Puppeteer.connect().

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

Example

The following is a typical example of using Puppeteer to drive automation:

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

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

Once you have created a `page` you have access to a large API to interact with the page, navigate, or find certain elements in that page. The 'page' documentation lists all the available methods.

Properties

Property	Modifiers	Type	Description
product		string	The name of the browser that is under automation ("chrome" or "firefox")

Methods

Method	Modifiers	Description
connect(options)		This method attaches Puppeteer to an existing browser instance.
createBrowserFetcher(options)		
defaultArgs(options)		
executablePath(channel)		
launch(options)		Launches puppeteer and launches a browser instance with given arguments and options when specified.