**1. Write a Selenium program using Node.js to automate a Google search.**

const { Builder, By, Key } = require('selenium-webdriver');

async function googleSearch() {

  let driver = await new Builder().forBrowser('chrome').build();

  try {

    await driver.get('https://www.google.com');

    const searchBox = await driver.findElement(By.name('q'));

    await searchBox.sendKeys('Selenium WebDriver', Key.RETURN);

  } finally {

    // Close browser after 5 seconds

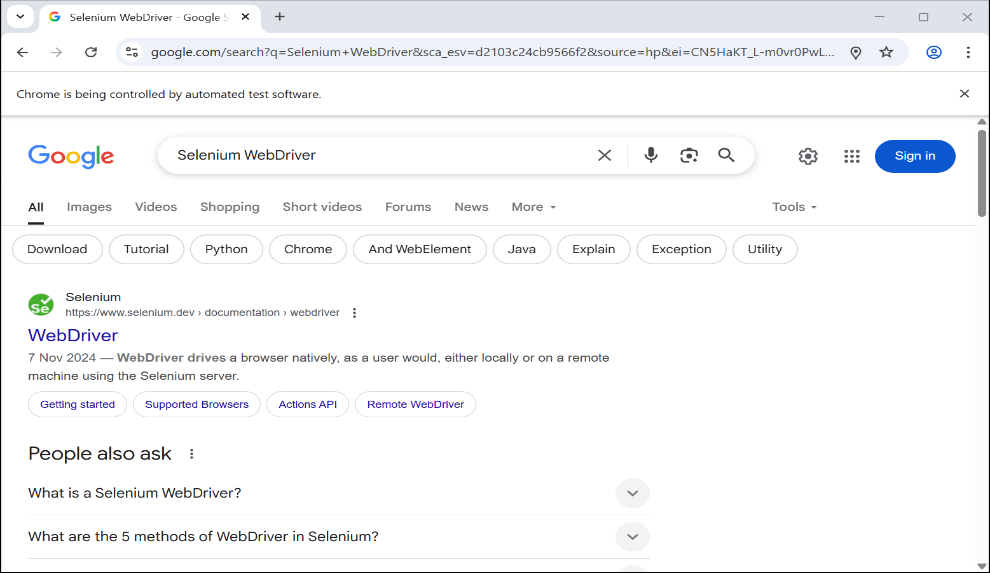
    setTimeout(() => driver.quit(), 1000000);

  }

}

googleSearch();

**Output:**



**2. Create a Selenium script in Node.js to open a web page, extract the text**

const { Builder, By, until } = require('selenium-webdriver');

async function extractWikipedia() {

  const driver = await new Builder().forBrowser('chrome').build();

  try {

    await driver.get('https://en.wikipedia.org/wiki/OpenAI');

    // Wait until the content area is loaded

    await driver.wait(until.elementLocated(By.css('#mw-content-text p')), 10000);

    // Get all <p> elements in the article body

    const paragraphs = await driver.findElements(By.css('#mw-content-text p'));

    for (const para of paragraphs) {

      const text = await para.getText();

      if (text.trim().length > 0) {

        console.log("\n Wikipedia - OpenAI (first non-empty paragraph):\n");

        console.log(text);

        break;

      }

    }

  } catch (err) {

    console.error(" Failed to extract paragraph:", err.message);

  } finally {

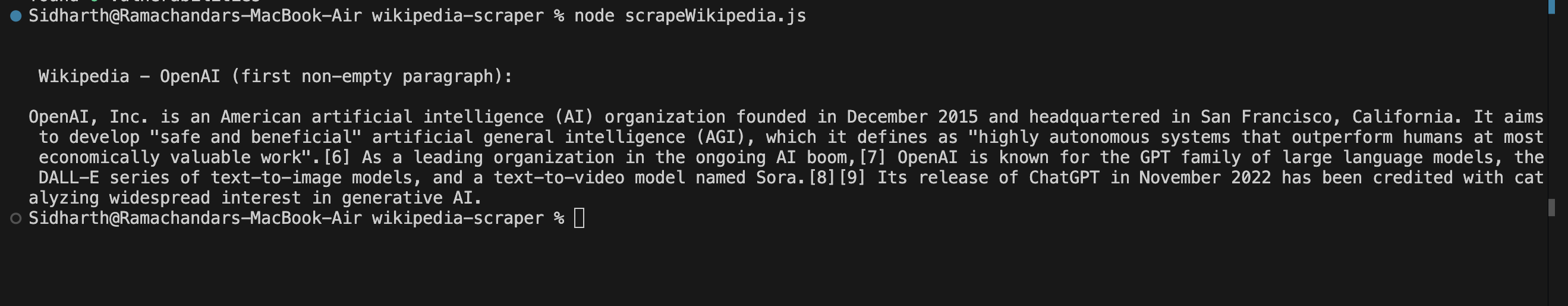
    await driver.quit();

  }

}

extractWikipedia();

**Output:**



**3. Handling browser navigation( back, Forward, refresh) using selenium webdriver**

const { Builder, By, Key, until } = require('selenium-webdriver');

const chrome = require('selenium-webdriver/chrome');

(async function handleBrowserNavigation() {

// Set up the WebDriver instance

let driver = await new Builder()

.forBrowser('chrome')

.setChromeOptions(new chrome.Options())

.build();

try {

// Navigate to the first URL (Google)

console.log('Navigating to Google...');

await driver.get('https://www.google.com');

await driver.sleep(2000); // Wait for 2 seconds for the page to load

// Navigate to a second URL (Example)

console.log('Navigating to Example.com...');

await driver.get('https://www.wikipedia.org/');

await driver.sleep(2000); // Wait for 2 seconds for the page to load

// --- Handle Browser Back Navigation ---

console.log('Navigating back...');

await driver.navigate().back();

await driver.sleep(2000); // Wait for 2 seconds for the page to load

// --- Handle Browser Forward Navigation ---

console.log('Navigating forward...');

await driver.navigate().forward();

await driver.sleep(2000); // Wait for 2 seconds for the page to load

// --- Handle Browser Refresh ---

console.log('Refreshing the page...');

await driver.navigate().refresh();

await driver.sleep(2000); // Wait for 2 seconds for the page to refresh

} catch (err) {

console.error('Error:', err);

} finally {

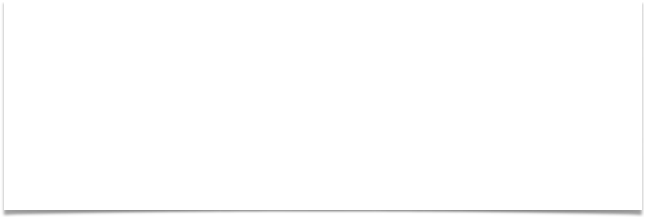
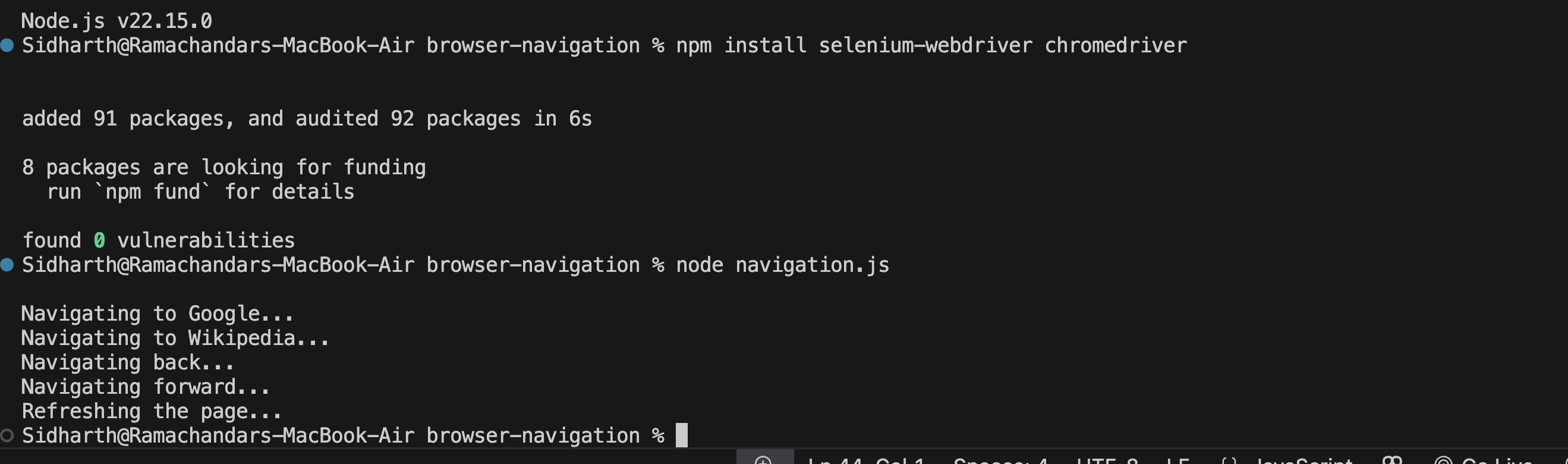
// Close the browser

await driver.quit();

}

})();

**Output:**



**4. Write a Node.js program using Selenium to count the total number of hyperlinks on the page**

const { Builder, By } = require('selenium-webdriver');

async function countLinks() {

let driver = await new Builder().forBrowser('chrome').build();

try {

await driver.get('https://www.wikipedia.org');

const links = await driver.findElements(By.tagName('a'));

console.log("Total hyperlinks:", links.length);

} finally {

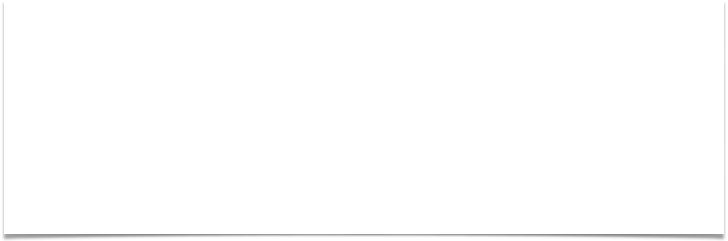
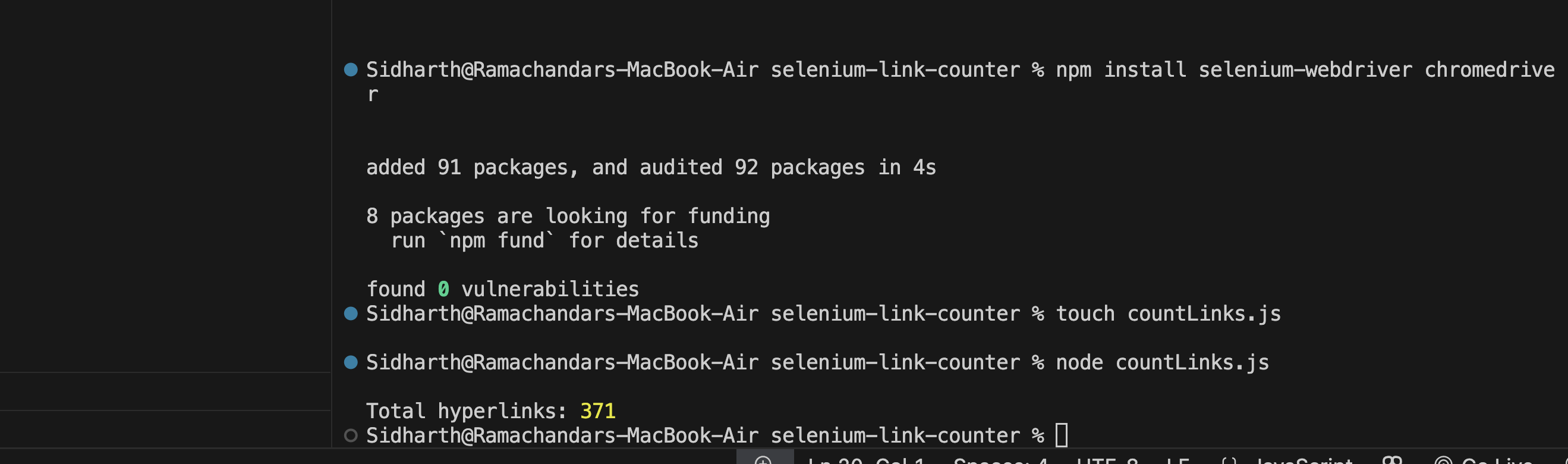
await driver.quit();

}

}

countLinks();

**Output:**



**5.Develop a Selenium script to automate login testing**

**Login.html**

<!DOCTYPE html>

<html>

<head><title>Login Page</title></head>

<body>

  <h2>Login</h2>

  <form id="loginForm">

    <label for="username">Username:</label>

    <input type="text" id="username" name="username"/><br/><br/>

    <label for="password">Password:</label>

    <input type="password" id="password" name="password"/><br/><br/>

    <input type="submit" value="Login" />

  </form>

  <p id="message"></p>

  <script>

    document.getElementById("loginForm").addEventListener("submit", function(event) {

      event.preventDefault();

      const user = document.getElementById("username").value;

      const pass = document.getElementById("password").value;

      if (user === "admin" && pass === "1234") {

        document.getElementById("message").textContent = "Login Successful";

      } else {

        document.getElementById("message").textContent = "Login Failed";

      }

    });

  </script>

</body>

</html>

**Index.js**

const { Builder, By, until } = require('selenium-webdriver');

const path = require('path');

async function testLogin() {

  const driver = await new Builder().forBrowser('chrome').build();

  try {

    const fileUrl = 'file://' + path.resolve('login.html'); // Local login page

    await driver.get(fileUrl);

    // Watch typing step by step

    const usernameInput = await driver.findElement(By.id('username'));

    const passwordInput = await driver.findElement(By.id('password'));

    await usernameInput.sendKeys('admin'); // Typing username

    await driver.sleep(1000); // Pause 1 sec to watch

    await passwordInput.sendKeys('1234');  // Typing password

    await driver.sleep(1000); // Pause 1 sec to watch

    // Click login

    await driver.findElement(By.css('input[type="submit"]')).click();

    // Wait until message shows

    const messageElem = await driver.wait(until.elementLocated(By.id('message')), 5000);

    const message = await messageElem.getText();

    console.log("\nLogin Result:", message);

    // Keep browser open to see result

    await driver.sleep(5000); // Wait 5 seconds

  } catch (err) {

    console.error(" Error:", err.message);

  } finally {

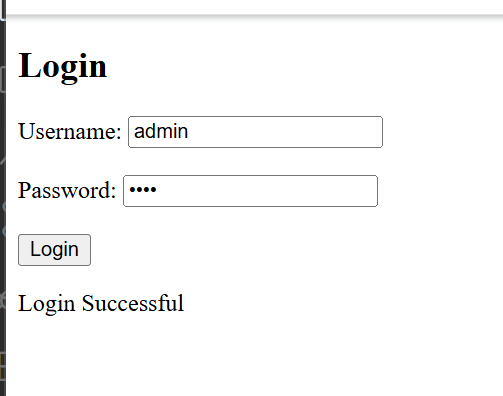
    await driver.quit(); // Now close the browser

  }

}

testLogin();

**Output:**



**6. Taking a screenshot using selenium webdriver**

const { Builder } = require('selenium-webdriver');

const chrome = require('selenium-webdriver/chrome');

const fs = require('fs');

(async function takeScreenshot() {

let driver = await new Builder()

.forBrowser('chrome')

.setChromeOptions(new chrome.Options())

.build();

try {

await driver.get('https://www.wikipedia.org');

// Take a screenshot

let screenshot = await driver.takeScreenshot();

fs.writeFileSync('screenshot.png', screenshot, 'base64');

console.log('Screenshot saved successfully as screenshot.png!');

} catch (err) {

console.error('Error:', err);

} finally {

await driver.quit();

}

})();

**Output:**

