Text input

**1. fill(selector, value)**

* **Purpose**: Clears the input field and fills it with the specified text.
* **Usage**: For simple input text handling.
* **Example**:

javascript

Copy code

await page.fill('input#username', 'yourUsername');

* **Deprecation**: Not deprecated.

**2. type(selector, text, [options])**

* **Purpose**: Types text character by character, simulating real typing.
* **Usage**: Useful when simulating realistic user input.
* **Example**:

javascript

Copy code

await page.type('input#password', 'yourPassword');

* **Deprecation**: Not deprecated.

**3. press(selector, key, [options])**

* **Purpose**: Simulates pressing a key inside a focused input or element.
* **Usage**: Often used to simulate key presses like "Enter", "Tab", etc.
* **Example**:

javascript

Copy code

await page.press('input#search', 'Enter');

* **Deprecation**: Not deprecated.

**4. focus(selector) + keyboard.type(text)**

* **Purpose**: Focus on an element first, then use the keyboard API to type.
* **Usage**: Allows using the keyboard API after focusing on an element.
* **Example**:

javascript

Copy code

await page.focus('input#email');

await page.keyboard.type('yourEmail@example.com');

* **Deprecation**: Not deprecated.

**5. keyboard.press(key)**

* **Purpose**: Simulates pressing a single key on the keyboard.
* **Usage**: Use for simulating control keys like "Enter", "Tab", "Shift", etc.
* **Example**:

javascript

Copy code

await page.keyboard.press('Tab'); // Moves to the next input field

await page.keyboard.press('Enter'); // Submits the form

* **Deprecation**: Not deprecated.

**6. evaluateHandle(selector) + JavaScript**

* **Purpose**: Directly manipulate the DOM to set the value of an input field.
* **Usage**: For advanced cases where direct JavaScript manipulation is required.
* **Example**:

javascript

Copy code

await page.evaluate(() => document.querySelector('input#name').value = 'John Doe');

* **Deprecation**: Not deprecated.

**7. pressSequentially(selector, text)**

* **Purpose**: Simulates pressing multiple keys in sequence, one character at a time.
* **Usage**: Useful for mimicking key presses in a sequential manner, like a real user typing one key at a time.
* **Example**:

javascript

Copy code

async function pressSequentially(page, selector, text) {

for (let char of text) {

await page.press(selector, char);

}

}

// Usage

await pressSequentially(page, 'input#username', 'HELLO');

* **Deprecation**: Not deprecated (custom function).