Playwright Actions – Input box, Radio buttons, Check boxes

1. Text Input / Text Box Handling

Text input boxes are used to enter user data. In Playwright, we use the fill() or type() methods to enter text.

✅ **Key Actions:**

- **Check if visible or enabled** using toBeVisible() and toBeEnabled()
- **Get attribute** like maxlength using getAttribute()
- **Set value** using fill()
- **Get entered value** using inputValue()

🔍 **Example:**

```
const textBox = page.locator('#name');

await expect(textBox).toBeVisible();

await expect(textBox).toBeEnabled();

const maxLength = await textBox.getAttribute("maxlength");

expect(maxLength).toBe('15');

await textBox.fill("John Canedy");

const enteredValue = await textBox.inputValue();

expect(enteredValue).toBe("John Canedy");
```

2. Radio Button Handling

Radio buttons let users choose **one option** from a set. Use .check() to select.

✅ **Key Actions:**

- **Check visibility and enabled state**
- **Use .check() to select**
- **Verify selection using .isChecked() or .toBeChecked()**

🔍 **Example:**

```
const maleRadio = page.locator('#male');

await expect(maleRadio).toBeVisible();

await expect(maleRadio).toBeEnabled();

expect(await maleRadio.isChecked()).toBe(false);
```

```
await maleRadio.check();

await expect(maleRadio).toBeChecked();
```

3. Checkbox Handling

Checkboxes allow selecting **multiple** options. You can check, uncheck, or toggle them.

✅ **Scenarios Covered:**

1. Select a specific checkbox

```
const sundayCheckbox = page.getByLabel('Sunday');

await sundayCheckbox.check();

await expect(sundayCheckbox).toBeChecked();
```

2. Select all checkboxes

```
const days = ['Sunday', 'Monday', ...];

const checkboxes = days.map(day => page.getByLabel(day));

for (const checkbox of checkboxes) {

  await checkbox.check();

  await expect(checkbox).toBeChecked();

}
```

3. Uncheck last 3 checkboxes

```
for (const checkbox of checkboxes.slice(-3)) {

  await checkbox.uncheck();

  await expect(checkbox).not.toBeChecked();

}
```

4. Toggle checkboxes

```
for (const checkbox of checkboxes) {

  if (await checkbox.isChecked()) {

    await checkbox.uncheck();

    await expect(checkbox).not.toBeChecked();

  } else {

    await checkbox.check();

    await expect(checkbox).toBeChecked();

  }

}
```

5. Select by specific indexes (e.g. 1, 3, 6)

```
const indexes = [1, 3, 6];
for (const i of indexes) {
  await checkboxes[i].check();
  await expect(checkboxes[i]).toBeChecked();
}
```

6. Select checkbox by label name

```
const weekname = "Friday";
for (const label of days) {
  if (label.toLowerCase() === weekname.toLowerCase()) {
    const checkbox = page.getByLabel(label);
    await checkbox.check();
    await expect(checkbox).toBeChecked();
  }
}
```

Summary Table

Element	Action	Playwright Method
Text Input	Enter text	fill(), type()
	Get value	inputValue()
Radio Button	Select option	check()
Checkbox	Select	check()
	Unselect	uncheck()
	Check status	isChecked(), toBeChecked()
All Elements	Visibility	toBeVisible(), toBeEnabled()