

ТЗЯПЗ  
ФБ-24 Тихонова Анастасія  
Лабораторна робота №7

Виконання коду за прикладом

## Завдання 1

Перевірка завантаження головної сторінки

- Мета: Переконаватися, що головна сторінка сайту успішно завантажується без помилок.
- Завдання: Відкрити вебсайт <https://kpi.ua> та перевірити код відповіді сервера (200 ОК).

JS task1.test.js > ...

```
1  const { test, expect } = require('@playwright/test');
2  test('Перевірка завантаження головної сторінки', async ({ page }) => {
3    const response = await page.goto('https://kpi.ua');
4    expect(response.status()).toBe(200);
5  });
```

```
● PS D:\3 курс\5 семестр\тзпз\lab7> npm run test
Debugger attached.

> lab7@1.0.0 test
> playwright test

Debugger attached.

Running 1 test using 1 worker

Debugger attached.
✓ 1 task1.test.js:2:1 › Перевірка завантаження головної сторінки (3.5s)
Waiting for the debugger to disconnect...

1 passed (5.7s)
Waiting for the debugger to disconnect...
Waiting for the debugger to disconnect...
```

## Завдання 2

Перевірка коректності HTTP → HTTPS перенаправлення

- Мета: Перевірити, чи перенаправляється запит з <http://kpi.ua> на <https://kpi.ua>.
- Завдання:
  - о Відкрити сайт за допомогою HTTP (<http://kpi.ua>).
  - о Переконаватися, що автоматично відбувається перенаправлення на HTTPS (<https://kpi.ua>).

```
JS task2.test.js > test('Перевірка HTTP → HTTPS перенаправлення') callback
1  const { test, expect } = require('@playwright/test');
2  test('Перевірка HTTP → HTTPS перенаправлення', async ({ page }) => {
3    const response = await page.goto('http://kpi.ua');
4    expect(response.status()).toBe(200); // Перевірка на перенаправлення на HTTPS
5    expect(page.url()).toBe('https://kpi.ua'); // Перевірка, що URL змінився на HTTPS
6  });
```

Debugger attached.

Running 2 tests using 2 workers

Debugger attached.

Debugger attached.

✓ 1 task1.test.js:2:1 › Перевірка завантаження головної сторінки (3.9s)

✗ 2 task2.test.js:2:1 › Перевірка HTTP → HTTPS перенаправлення (3.9s)

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

1) task2.test.js:2:1 › Перевірка HTTP → HTTPS перенаправлення

Error: expect(received).toBe(expected) // Object.is equality

Expected: "https://kpi.ua"

Received: "https://kpi.ua"

```
3 |   const response = await page.goto('http://kpi.ua');
4 |   expect(response.status()).toBe(200); // Перевірка на перенаправлення на HTTPS
> 5 |   expect(page.url()).toBe('https://kpi.ua'); // Перевірка, що URL змінився на HTTPS
    |                               ^
6 |   });
   at D:\3 курс\5 семестр\тз\lab7\task2.test.js:5:21
```

1 failed

task2.test.js:2:1 › Перевірка HTTP → HTTPS перенаправлення

1 passed (6.4s)

Waiting for the debugger to disconnect...

У результаті вивело помилку перенаправлення, але якщо в кінці посилання додати «/», то спрацює все

## Завдання 3

: Тестування роботи меню навігації

- Мета: Переконаватися, що всі посилання у головному меню працюють коректно.
- Завдання:

о Перейти до кожного пункту меню.

о Перевірити, що сторінка за відповідним посиланням завантажується без помилок.

```
JS task3.test.js > ...
1  const { test, expect } = require('@playwright/test');
2  test('Тестування роботи меню навігації', async ({ page }) => {
3    await page.goto('https://kpi.ua');
4    const aboutUsLink = await page.locator('a[href="/about"]');
5    // Замість '/about' використовуйте реальний селектор
6    await aboutUsLink.click();
7    expect(await page.title()).toContain('Про університет');
8  });
```

Running 3 tests using 3 workers

```
Debugger attached.
Debugger attached.
Debugger attached.
✓ 1 task3.test.js:2:1 > Тестування роботи меню навігації (4.5s)
✓ 2 task1.test.js:2:1 > Перевірка завантаження головної сторінки (4.0s)
✗ 3 task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення (1.5s)
Waiting for the debugger to disconnect...
Waiting for the debugger to disconnect...
Waiting for the debugger to disconnect...
```

1) task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення

Error: expect(received).toBe(expected) // Object.is equality

Expected: 200

Received: 503

```
2 | test('Перевірка HTTP → HTTPS перенаправлення', async ({ page }) => {
3 |   const response = await page.goto('http://kpi.ua');
> 4 |   expect(response.status()).toBe(200); // Перевірка на перенаправлення на HTTPS
    |                               ^
5 |   expect(page.url()).toBe('https://kpi.ua'); // Перевірка, що URL змінився на HTTPS
6 | });
    at D:\3 курс\5 семестр\тязпз\lab7\task2.test.js:4:28
```

1 failed

task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення

2 passed (8.1s)

Waiting for the debugger to disconnect...

## Завдання 4

### Перевірка внутрішніх посилань

- Мета: Упевнитися, що всі внутрішні посилання на сторінці є працездатними.

- Завдання:

- о Знайти всі внутрішні посилання на головній сторінці.

- о Перевірити, що кожне з них веде до сторінки, яка успішно завантажується (код відповіді 200 OK).

JS task4.test.js > ...

```
1  const { test, expect } = require('@playwright/test');
2  test('Перевірка внутрішніх посилань', async ({ page }) => {
3    await page.goto('https://kpi.ua');
4    const links = await page.locator('a[href^="/"]'); // Всі внутрішні посилання
5    for (let i = 0; i < await links.count(); i++) {
6      const link = links.nth(i);
7      const href = await link.getAttribute('href');
8      const response = await page.goto(`https://kpi.ua${href}`);
9      expect(response.status()).toBe(200);
10   }
11  });
```

Debugger attached.

Running 4 tests using 4 workers

Debugger attached.

Debugger attached.

Debugger attached.

Debugger attached.

```
X 1 task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення (17.2s)
X 2 task3.test.js:2:1 > Тестування роботи меню навігації (30.0s)
✓ 3 task1.test.js:2:1 > Перевірка завантаження головної сторінки (3.9s)
✓ 4 task4.test.js:2:1 > Перевірка внутрішніх посилань (1.6s)
```

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

1) task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення

Error: expect(received).toBe(expected) // Object.is equality

Expected: "https://kpi.ua"

Received: "https://kpi.ua/"

```
3 |   const response = await page.goto('http://kpi.ua');
4 |   expect(response.status()).toBe(200); // Перевірка на перенаправлення на HTTPS
> 5 |   expect(page.url()).toBe('https://kpi.ua'); // Перевірка, що URL змінився на HTTPS
    |                               ^
```

## Завдання 5

### Тестування адаптивності сторінки

- Мета: Перевірити, як сайт виглядає на пристроях з різними розмірами екрану.
- Завдання:

о Емулювати пристрої (смартфон, планшет, десктоп) і перевірити, чи коректно відображаються всі елементи сайту

JS task5.test.js > ...

```
1  const { test, expect } = require('@playwright/test');
2  test('Тестування адаптивності сторінки', async ({ page }) => {
3    await page.goto('https://kpi.ua');
4    await page.setViewportSize({ width: 1200, height: 800 });
5    expect(await page.isVisible('header')).toBe(true); // Перевірка, чи видимий header
6    await page.setViewportSize({ width: 375, height: 667 });
7    expect(await page.isVisible('header')).toBe(true);
8  });
```

Debugger attached.

Debugger attached.

Debugger attached.

```
X 1 task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення (1.5s)
✓ 2 task1.test.js:2:1 > Перевірка завантаження головної сторінки (4.3s)
✓ 3 task5.test.js:2:1 > Тестування адаптивності сторінки (4.6s)
✓ 4 task3.test.js:2:1 > Тестування роботи меню навігації (5.2s)
✓ 5 task4.test.js:2:1 > Перевірка внутрішніх посилань (1.4s)
```

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

Waiting for the debugger to disconnect...

1) task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення

Error: expect(received).toBe(expected) // Object.is equality

Expected: 200

Received: 503

```
2 |   test('Перевірка HTTP → HTTPS перенаправлення', async ({ page }) => {
3 |     const response = await page.goto('http://kpi.ua');
> 4 |     expect(response.status()).toBe(200); // Перевірка на перенаправлення на HTTPS
    |                               ^
5 |     expect(page.url()).toBe('https://kpi.ua'); // Перевірка, що URL змінився на HTTPS
6 |   });
    at D:\3 курс\5 семестр\тяяпа\lab7\task2.test.js:4:28
```

1 failed

task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення

4 passed (7.2s)

## Завдання для самостійного написання коду

### Завдання 6

#### Тестування часу завантаження сайту

- Мета: Переконаватися, що час завантаження сторінки є оптимальним.
- Завдання: Виміряти час завантаження сторінки та порівняти його з рекомендаціями (менше 3 секунд).

```
JS task6.test.js > ...
1  const { test, expect } = require('@playwright/test');
2  test('Тестування часу завантаження сторінки', async ({ page }) => {
3      const startTime = Date.now();
4      await page.goto('https://kpi.ua');
5      const loadTime = Date.now() - startTime;
6      console.log(`Час завантаження: ${loadTime} мс`);
7      expect(loadTime).toBeLessThan(3000); // Час завантаження має бути менше 3 секунд
8  });
9

Running 6 tests using 6 workers

Debugger attached.
Debugger attached.
Debugger attached.
Debugger attached.
Debugger attached.
Debugger attached.
X 1 task4.test.js:2:1 > Перевірка внутрішніх посилань (30.0s)
X 2 task3.test.js:2:1 > Тестування роботи меню навігації (30.0s)
X 3 task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення (1.4s)
✓ 4 task1.test.js:2:1 > Перевірка завантаження головної сторінки (4.1s)
✓ 5 task5.test.js:2:1 > Тестування адаптивності сторінки (4.4s)
✓ 6 task6.test.js:2:1 > Тестування часу завантаження сторінки (1.2s)
Час завантаження: 67 мс
Waiting for the debugger to disconnect...
Waiting for the debugger to disconnect...
Waiting for the debugger to disconnect...
Waiting for the debugger to disconnect...
Waiting for the debugger to disconnect...
Waiting for the debugger to disconnect...
```

### Завдання 7

#### Перевірка наявності всіх основних елементів на головній сторінці

- Мета: Переконаватися, що на головній сторінці присутні всі основні елементи, такі як:
  - о Логотип сайту.
  - о Головне меню навігації.
  - о Блок новин або оголошень.
  - о Пошуковий рядок.
- Завдання: Візуально перевірити, що ці елементи коректно відображаються на сторінці та не зникають при зміні розміру екрана

JS task7.test.js > ...

```
1  const { test, expect } = require('@playwright/test');
2
3  test('Перевірка наявності основних елементів', async ({ page }) => {
4      await page.goto('https://kpi.ua');
5
6      // Логотип сайту
7      expect(await page.isVisible('img[alt="Логотип"]')).toBe(true);
8
9      // Головне меню навігації
10     expect(await page.isVisible('nav')).toBe(true);
11
12     // Блок новин або оголошень
13     expect(await page.isVisible('section#news')).toBe(true);
14
15     // Пошуковий рядок
16     expect(await page.isVisible('input[type="search"]')).toBe(true);
17 }
```

Debugger attached.

Debugger attached.

Debugger attached.

Debugger attached.

Debugger attached.

Debugger attached.

- ✓ 1 task5.test.js:2:1 > Тестування адаптивності сторінки (6.2s)
- ✗ 2 task6.test.js:2:1 > Тестування часу завантаження сторінки (5.1s)
- ✓ 3 task4.test.js:2:1 > Перевірка внутрішніх посилань (1.4s)
- ✓ 4 task1.test.js:2:1 > Перевірка завантаження головної сторінки (5.2s)
- ✗ 5 task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення (5.0s)
- ✓ 6 task3.test.js:2:1 > Тестування роботи меню навігації (6.0s)
- ✗ 7 task7.test.js:3:1 > Перевірка наявності основних елементів (4.5s)

## Завдання 8

### Тестування доступності сайту

- Мета: Перевірити, чи відповідає сайт основним принципам доступності (accessibility).
- Завдання:
  - о Переконалися, що кожен елемент має коректні атрибути alt, aria-label або аналогічні.
  - о Використати інструменти тестування доступності.

JS task8.test.js > ...

```
1  const { test, expect } = require('@playwright/test');
2
3  test('Тестування доступності сайту', async ({ page }) => {
4      await page.goto('https://kpi.ua');
5
6      // Перевірка наявності атрибутів alt у зображеннях
7      const images = await page.locator('img');
8      for (let i = 0; i < await images.count(); i++) {
9          const alt = await images.nth(i).getAttribute('alt');
10         expect(alt).not.toBeNull(); // Кожне зображення має атрибут alt
11     }
12
13     // Перевірка наявності aria-label у кнопках
14     const buttons = await page.locator('button');
15     for (let i = 0; i < await buttons.count(); i++) {
16         const ariaLabel = await buttons.nth(i).getAttribute('aria-label');
17         expect(ariaLabel).not.toBeNull(); // Кожна кнопка має атрибут aria-label
18     }
19 });
```

```

-----
✓ 1 task1.test.js:2:1 > Перевірка завантаження головної сторінки (5.0s)
✓ 2 task5.test.js:2:1 > Тестування адаптивності сторінки (5.6s)
✗ 3 task4.test.js:2:1 > Перевірка внутрішніх посилань (30.0s)
✗ 4 task6.test.js:2:1 > Тестування часу завантаження сторінки (4.5s)
✗ 5 task3.test.js:2:1 > Тестування роботи меню навігації (30.0s)
✗ 6 task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення (1.2s)
Waiting for the debugger to disconnect...
Debugger attached.
✗ 7 task7.test.js:3:1 > Перевірка наявності основних елементів (4.6s)
Час завантаження: 3336 мс
✓ 8 task8.test.js:3:1 > Тестування доступності сайту (4.4s)
Waiting for the debugger to disconnect...

```

## Завдання 9

### Перевірка завантаження сторінок іншими мовами

- Мета: Переконаватися, що сайт коректно відображається при виборі іншої мови інтерфейсу (наприклад, англійської).

- Завдання:

- о Знайти перемикач мови на сайті.

- о Вибрати англійську мову (або будь-яку іншу доступну).

- о Перевірити, що всі основні елементи сайту (меню, заголовки, кнопки) відображаються обраною мовою.

```

JS task9.test.js > ...
1  const { test, expect } = require('@playwright/test');
2
3  test('Перевірка завантаження сторінок іншими мовами', async ({ page }) => {
4    await page.goto('https://kpi.ua');
5
6    // Перемикач мови (припустимо, це кнопка з текстом "EN")
7    const languageSwitch = await page.locator('button:has-text("EN")');
8    await languageSwitch.click();
9
10   // Перевірка, що текст на сторінці змінився на англійську
11   expect(await page.locator('h1').innerText()).toContain('University');
12 });
13
-----
✓ 1 task3.test.js:2:1 > Тестування роботи меню навігації (6.8s)
✓ 2 task4.test.js:2:1 > Перевірка внутрішніх посилань (1.4s)
✓ 3 task5.test.js:2:1 > Тестування адаптивності сторінки (5.6s)
✓ 4 task1.test.js:2:1 > Перевірка завантаження головної сторінки (5.1s)
✗ 5 task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення (1.3s)
✓ 6 task6.test.js:2:1 > Тестування часу завантаження сторінки (1.2s)
Час завантаження: 44 мс
✗ 7 task7.test.js:3:1 > Перевірка наявності основних елементів (4.7s)
✓ 8 task8.test.js:3:1 > Тестування доступності сайту (5.0s)
Waiting for the debugger to disconnect...
Debugger attached.
✗ 9 task9.test.js:3:1 > Перевірка завантаження сторінок іншими мовами (30.0s)
Waiting for the debugger to disconnect...

```

## Завдання 10

### Перевірка завантаження зовнішніх ресурсів

- Мета: Переконалися, що всі зовнішні ресурси (зображення, CSS, JavaScript) завантажуються коректно.

- Завдання:

- о Відкрити консоль розробника браузера (DevTools) і перевірити вкладку "Network".

- о Переконалися, що немає помилок завантаження (404, 500).

- о Перевірити, що всі ресурси мають оптимальний час завантаження (без значних затримок, тобто не більше 3 секунд)

```
JS task10.test.js > ...
1  const { test, expect } = require('@playwright/test');
2
3  test('Перевірка завантаження зовнішніх ресурсів', async ({ page }) => {
4      await page.goto('https://kpi.ua');
5      const requests = page.context().on('requestfinished', request => {
6          const response = request.response();
7          if (response) {
8              expect(response.status()).toBeLessThan(400); // Всі ресурси мають статус < 400
9              console.log(`Завантажено: ${request.url()} - ${response.status()}`);
10         }
11     });
12 });
13
```

Debugger attached.

Debugger attached.

Debugger attached.

Debugger attached.

Debugger attached.

Debugger attached.

✓ 1 task10.test.js:3:1 > Перевірка завантаження зовнішніх ресурсів (1.5s)

✓ 2 task4.test.js:2:1 > Перевірка внутрішніх посилань (1.4s)

✓ 3 task3.test.js:2:1 > Тестування роботи меню навігації (6.3s)

✓ 4 task1.test.js:2:1 > Перевірка завантаження головної сторінки (4.8s)

✗ 5 task2.test.js:2:1 > Перевірка HTTP → HTTPS перенаправлення (5.0s)

✓ 6 task5.test.js:2:1 > Тестування адаптивності сторінки (5.7s)

✓ 7 task6.test.js:2:1 > Тестування часу завантаження сторінки (215ms)

✗ 8 task7.test.js:3:1 > Перевірка наявності основних елементів (4.6s)

Час завантаження: 83 мс

✓ 9 task8.test.js:3:1 > Тестування доступності сайту (5.0s)

✗ 10 task9.test.js:3:1 > Перевірка завантаження сторінок іншими мовами (30.0s)

Waiting for the debugger to disconnect...