

Exercise

Một số key notes:

Kế thừa (extends)

- Kế thừa giúp bạn tái sử dụng các thuộc tính của phần tử cha.
- Hàm tạo (**constructor**) là hàm sẽ chạy khi bạn khởi tạo một object.

Ví dụ:

Hàm tạo:

```
export class SimpleClass {  
  constructor() {  
    console.log("Hello Playwright");  
  }  
}
```

Sử dụng:

```
const instance = new SimpleClass();
```

Khi khởi tạo một đối tượng mới từ class **SimpleClass**, hàm **console.log** sẽ tự được chạy.

- Từ khóa **super()** dùng để gọi hàm tạo của hàm cha. Khi kế thừa, bạn luôn cần gọi **super()** trong hàm cha.

Export

- Từ khóa **export** giúp chúng ta có thể xuất 1 biến, 1 hằng số ở 1 file và nhập (import) dùng ở file khác.

Ví dụ:

Tại file **login-page.ts**

```
export class LoginPage {  
  // Some code...  
}
```

Tại file **test.spec.ts**

```
import { LoginPage } from './page/login-page';
```

```
// Some code...
```

Lưu ý:

- Cấu trúc thư mục:

```
|— test.spec.ts
|— page
|   |— login-page.ts
```

- `from './page/login-page'`; là đường dẫn để đi tới `login-page` cần import. Do `login-page` này nằm trong thư mục `page` nên cần định nghĩa vào trong import.
- Ta có thể viết `from './page/login-page.ts'`; hoặc bỏ đuôi `.ts` trong phần import: `from './page/login-page'`; , vì Javascript sẽ tự động thêm đuôi `.ts` khi tìm kiếm
- Để import file nằm ở thư mục bên ngoài, ta dùng `..` để đi ra folder cha của folder hiện tại.

VD: `'../../login-page.ts'`

POM

- POM = Page Object Model
- POM giúp code tổ chức gọn gàng hơn, dễ bảo trì hơn
- POM gồm 2 thành phần chính:
 - Các thuộc tính (property): đại diện cho các phần tử trên trang.
 - Các phương thức (method): đại diện cho các hành động trên trang.
- Hàm tạo của POM thường có thuộc tính `page`. Ta sẽ dùng `page` này để tương tác với trang web.
 - Thuộc tính `page` này nếu đã xuất hiện trong POM cha rồi thì bạn không cần định nghĩa ở POM con nữa.

POM

Kiến thức bổ sung: Refactoring

Refactoring là viết lại source code một cách khoa học hơn mà vẫn giữ được tính đúng đắn và giá trị về chức năng của source code đó.

Tại sao phải refactoring code

Refactoring không hề làm hệ thống chạy nhanh hơn, bảo mật hơn tuy nhiên nó sẽ giúp source code dễ tiếp cận, dễ đọc, dễ hiểu từ đó giúp ích rất nhiều cho quá trình bảo trì, mở rộng hệ thống.

Khi nào thì thực hiện refactoring

Bất cứ khi nào bạn muốn đoạn code của mình "tốt hơn" thì đều có thể thực hiện refactoring. Tuy nhiên một số giai đoạn dưới đây được cho là thích hợp hơn để làm refactoring.

- Khi thêm chức năng mới vào source cũ
- Khi tiến hành review code
- Khi cần handover (bàn giao) lại

Một số dấu hiệu cần refactoring (smell)

- Smell: Method, functions quá dài
- Smell: quá nhiều parameters trong method, functions
- Smell: Class quá lớn

Bài tập

Hãy sử dụng POM và hooks, hoàn thiện bài tập sau:

1. Tạo file `test1.spec.ts`. Truy cập trang <https://material.playwrightvn.com/>, click vào "Bài học 1: Register Page"
 - a. Nhập đầy đủ các thông tin, click button register
 - b. Kiểm tra nội dung đã đăng kí ở bảng là đúng
2. Tạo file `test2.spec.ts`. Truy cập trang <https://material.playwrightvn.com/>, click vào "Bài học 2: Product page":
 - a. Hãy thêm sản phẩm để giỏ hàng có số lượng sản phẩm như sau:
 - i. Sản phẩm 1: 2 sản phẩm
 - ii. Sản phẩm 2: 3 sản phẩm
 - iii. Sản phẩm 3: 1 sản phẩm
 - b. Kiểm tra số lượng sản phẩm tại giỏ hàng đúng như đã thêm.
 - c. Kiểm tra tổng tiền tại giỏ hàng đúng.
3. Tạo file `test3.spec.ts`. Truy cập trang <https://material.playwrightvn.com/>, click vào "Bài học 3: Todo page".
 - a. Thêm mới 100 todo item có nội dung "Todo <i>"
 - b. Xoá các todo có số lẻ
 - c. Kiểm tra todo có số thứ tự 90 nằm trong viewport.
 - d. Kiểm tra todo có số thứ tự 21 bị ẩn (không nằm trong DOM)
4. Tạo file `test4.spec.ts`. Truy cập trang <https://material.playwrightvn.com/>, click vào "Bài học 4: Personal notes".
 - a. Thêm mới 10 note có nội dung là tiêu đề và một phần ngắn (khoảng 3 dòng) tại báo <https://vnexpress.net/khoa-hoc>,
 - b. Thực hiện search theo keyword bất kì.
 - c. Kiểm tra tất cả các bài báo search được đều chứa keyword đã tìm kiếm.

