

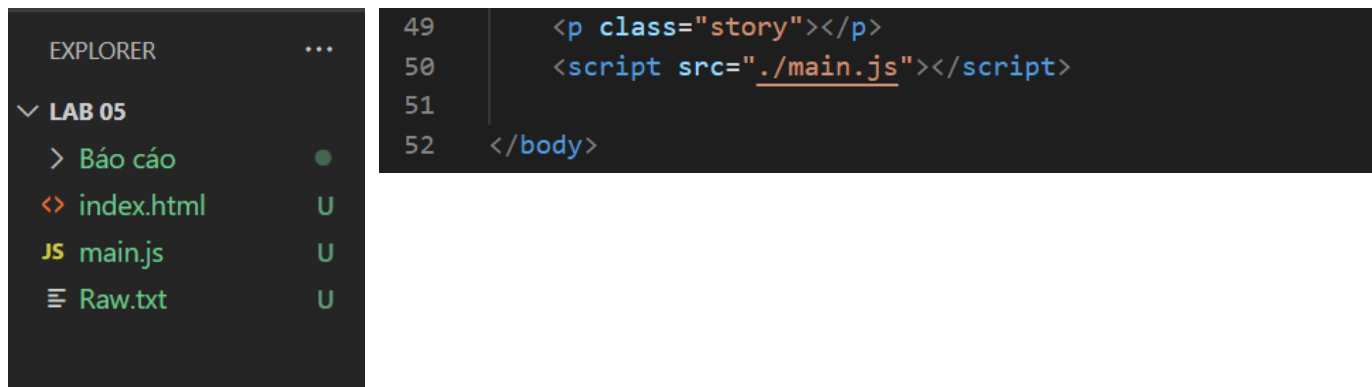
# BÁO CÁO BÀI THỰC HÀNH 05

## Bài 01

### Nội dung thực hiện:

#### 4.1. Thiết lập căn bản:

- Tạo một tập tin tên main.js, nằm cùng thư mục với tập tin index.html.
- Áp dụng nội dung tập tin main.js ở trên bằng cách chèn phần tử `<script>` có tham chiếu tới nội dung tập tin main.js. Và đặt phần tử này trước thẻ đóng `</body>` của nội dung tập tin index.html.



- Để kiểm tra xem nội dung JavaScript đã tác động lên nội dung HTML chưa, ta có thể dùng ví dụ sau để thay đổi màu sắc của trang web sang màu đỏ. Nếu thấy màu nền chuyển đỏ thì chứng tỏ tập tin main.js đã được tham chiếu đúng cách:

**`document.querySelector('html').style.backgroundColor = 'red';`**


### Kết quả thực hiện:



#### 4.2. Khởi tạo biến và phương thức:

- Trong dữ liệu thô bên trên, sao chép tất cả mã nguồn bên dưới mục "1. COMPLETE VARIABLE AND FUNCTION DEFINITIONS" vào trong nội dung tập tin main.js.

Nó tạo ra 3 biến lưu nội dung tham chiếu tới 3 phần tương ứng là “Enter custom name” (customName), nút “Generate random story” (randomize), và phần tử <p> cuối phần thân của HTML là nơi câu chuyện hiển thị lên (story). Ngoài ra, bạn còn có một phương thức randomValueFromArray(), được dùng để trả về một giá trị ngẫu nhiên từ trong một mảng lưu trữ các giá trị.

```
JS main.js >  result
1 // 20521150 - Phạm Quốc Cường - Lab 05
2 const customName = document.getElementById('customname');
3 const randomize = document.querySelector('.randomize');
4 const story = document.querySelector('.story');
5
6 function randomValueFromArray(array){
7     const random = Math.floor(Math.random()*array.length);
8     return array[random];
9 }
```

- Trong mục “2. RAW TEXT STRINGS” bao gồm chuỗi văn bản được xem như đầu vào của ứng dụng web này. Chúng ta chứa các dữ liệu thô này bên trong các biến của tập tin main.js:

- + Biến storyText để lưu đoạn văn bản đầu tiên.
- + Mảng insertX để lưu tập hợp 3 đoạn văn bản tiếp theo.
- + Mảng insertY để lưu tập hợp 3 đoạn văn bản tiếp theo.
- + Mảng insertZ để lưu tập hợp 3 đoạn văn bản cuối cùng.

```
11 const storyText = 'It was 94 fahrenheit outside, so :insertx: went for a walk. When they got to :inserty:, they
    stared in horror for a few moments, then :insertz:. Bob saw the whole thing, but was not surprised - :insertx:
    weighs 300 pounds, and it was a hot day.';
12 const insertX = ['Willy the Goblin', 'Big Daddy', 'Father Christmas'];
13 const insertY = ['the soup kitchen', 'Disneyland', 'the White House'];
14 const insertZ = ['spontaneously combusted', 'melted into a puddle on the sidewalk', 'turned into a slug and crawled
    away'];
15
```

#### 4.4. Hoàn chỉnh phương thức result():

- Tạo biến `newStory`, và gán nó bằng giá trị của `storyText`. Điều này là cần thiết khi ta tạo ra một câu chuyện ngẫu nhiên mới khi nút tạo được nhấn. Nếu ta thay trực tiếp ở biến `storyText`, thì ta chỉ tạo ra câu chuyện mới được 1 lần.

```
let newStory = storyText;
```

- Tạo 3 biến `xItem`, `yItem`, `zItem`, và gán nó bằng giá trị trả về của phương thức `randomValueFromArray()` với 3 mảng mà ta đã có (kết quả cho mỗi biến là một giá trị ngẫu nhiên từ 3 mảng dữ liệu). Ví dụ, bạn có thể gọi hàm và lấy giá trị từ mảng `insertX` bằng hàm `randomValueFromArray(insertX)`.

```
const xItem = randomValueFromArray(insertX);
const yItem = randomValueFromArray(insertY);
const zItem = randomValueFromArray(insertZ);

newStory = newStory.replaceAll(':insertx:', xItem);
newStory = newStory.replaceAll(':inserty:', yItem);
newStory = newStory.replaceAll(':insertz:', zItem);
```

- Tiếp theo, chúng ta muốn thay thế 3 chỗ dữ liệu trong `newStory` - `:insertx:`, `:inserty:`, `:insertz:` - với 3 giá trị đang lưu trong 3 biến `xItem`, `yItem`, `zItem`. Có một phương thức với chuỗi giúp chúng ta làm việc này.

=> **Sử dụng phương thức `replace` / `replaceAll`** để giải quyết việc thay thế.

- Trong khối lệnh `if`, thêm một phương thức để thay thế tên 'Bob' tìm thấy trong `newStory` bằng nội dung trong biến `name`. Đoạn mã nguồn này có thể được diễn giải: "Nếu có một giá trị nhập vào `customName`, hãy thay thế Bob với tên vừa được nhập".

```
if (customName.value !== '') {
  const name = customName.value;
```

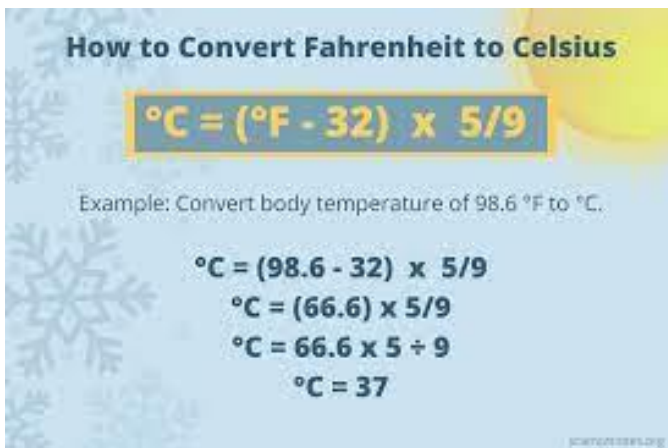
```
newStory = newStory.replaceAll('Bob', name);
}
```

- Trong khối if thứ 2, chúng ta kiểm tra xem là nút radio uk có được chọn hay không. Nếu có, chúng ta chuyển đơn vị cân nặng và nhiệt độ trong câu chuyện từ Pounds và Fahrenheit (độ F) sang stones và centigrade. Các bước như sau:

```
if (document.getElementById("uk").checked) { ... }
```

- + Tìm công thức chuyển đổi pounds sang stone, Fahrenheit sang centigrade.

1 pound =  
0.0714285714 stone



- + Trong dòng mã nguồn định nghĩa biến weight, thay 300 bằng phép toán chuyển đổi 300 pounds sang stones. Cộng chuỗi ' stone' vào sau phương thức Math.round().

```
const weight = `${Math.round(300*0.0714285714)} stone`;
```

- + Trong dòng mã nguồn định nghĩa biến temperature, thay 94 bằng phép toán chuyển đổi 94 Fahrenheit sang centigrade. Cộng chuỗi ' centigrade' vào sau phương thức Math.round().

```
const temperature = `${Math.round((94-32) * 5 / 9)} centigrade`;
```

- + Ngay dòng bên dưới 2 biến trên, thêm 2 dòng mã nguồn với mục đích thay thế chuỗi '94 fahrenheit' với giá trị của biến temperature, và '300 pounds' với giá trị của biến weight.

```
newStory = newStory.replaceAll('94 fahrenheit', temperature);  
newStory = newStory.replaceAll('300 pounds', weight);
```

- Hai dòng cuối cùng của phương thức này, gán thuộc tính textContent của biến story (tham chiếu tới đoạn văn) bằng newStory.

```
story.textContent = newStory;  
story.style.visibility = 'visible';
```

**Nội dung toàn bộ hàm result:**

```

18  function result() {
19      let newStory = storyText;
20
21      const xItem = randomValueFromArray(insertX);
22      const yItem = randomValueFromArray(insertY);
23      const zItem = randomValueFromArray(insertZ);
24
25      newStory = newStory.replaceAll(':insertx:',xItem);
26      newStory = newStory.replaceAll(':inserty:',yItem);
27      newStory = newStory.replaceAll(':insertz:',zItem);
28
29      if (customName.value !== '') {
30          const name = customName.value;
31          newStory = newStory.replaceAll('Bob', name);
32      }
33
34      if (document.getElementById("uk").checked) {
35          const weight = `${Math.round(300*0.0714285714)} stone`;
36          const temperature = `${Math.round((94-32) * 5 / 9)} centigrade`;
37          newStory = newStory.replaceAll('94 fahrenheit', temperature);
38          newStory = newStory.replaceAll('300 pounds', weight);
39      }
40
41      story.textContent = newStory;
42      story.style.visibility = 'visible';
43  }

```

**Kết quả sau thực hiện:**

- Không điền custom name

---

Enter custom name:

US ☒ UK ☐

Generate random story

It was 94 fahrenheit outside, so Father Christmas went for a walk. When they got to the White House, they stared in horror for a few moments, then turned into a slug and crawled away. Bob saw the whole thing, but was not surprised — Father Christmas weighs 300 pounds, and it was a hot day.

- Điền custom name

Enter custom name:

US ☒ UK ☐

Generate random story

It was 94 fahrenheit outside, so Father Christmas went for a walk. When they got to the soup kitchen, they stared in horror for a few moments, then spontaneously combusted. Cường saw the whole thing, but was not surprised — Father Christmas weighs 300 pounds, and it was a hot day.

- Chuyển qua hệ số của UK

Enter custom name:

US ☐ UK ☒

Generate random story

It was 34 centigrade outside, so Willy the Goblin went for a walk. When they got to Disneyland, they stared in horror for a few moments, then turned into a slug and crawled away. Cường saw the whole thing, but was not surprised — Willy the Goblin weighs 21 stone, and it was a hot day.

## Bài 02

### 4.1. Lặp qua các hình ảnh:

Đoạn mã nguồn được cung cấp có dòng code giúp tham chiếu phần tử thumb-bar <div> là hằng số (const) thumbBar, hãy tạo một phần tử <img> mới, thiết lập thuộc tính (attribute) src thành một nơi chờ dữ liệu với giá trị 'xxx', và gắn cuối (append) phần tử <img> mới này vào trong thumbBar.

```
for (let i = 1; i <= 5; i++) {  
  const newImage = document.createElement("img");  
  newImage.setAttribute("src", "images/pic" + i + ".jpg");  
  thumbBar.appendChild(newImage);  
}
```

### 4.2. Áp dụng sự kiện onclick vào mỗi hình ảnh nhỏ (thumbnail image):

- Trong vòng lặp, bạn cần thêm một sự kiện onclick vào newImage hiện hành (hình ảnh đang được ấn) - sự kiện này phải lấy được giá trị của thuộc tính src của hình ảnh hiện hành. Thiết lập giá trị thuộc tính src của displayed-img <img> bằng giá trị của thuộc tính



src được truyền vào từ tham số. (Cách thay thế, bạn có thể thêm 1 lắng nghe sự kiện tại thumb bar là được).

```
newImage.onclick = function (e) {  
    displayedImage.src = e.target.src;  
};
```

Cấu trúc nội dung vòng lặp:

```
for (let i = 1; i <= 5; i++) {  
    const newImage = document.createElement("img");  
    newImage.setAttribute("src", "images/pic" + i + ".jpg");  
    thumbBar.appendChild(newImage);  
    newImage.onclick = function (e) {  
        displayedImage.src = e.target.src;  
    };  
}
```

4.3. Viết đoạn mã nguồn cho nút (button) quản lý việc chuyển sáng/tối cho hình ảnh:

Bạn cần thêm sự kiện onclick cho <button> đã được định nghĩa trong mã nguồn với một hằng số tên là btn.

```
btn.onclick = function () { ... }
```

Kết quả thực hiện:



- Kiểm tra tên lớp hiện tại đang được thiết lập trong <button> - bạn có thể làm thông qua việc sử dụng phương thức `getAttribute()`.

```
const btnClass = btn.getAttribute("class");
```

- Nếu tên lớp là “dark”: đổi lớp của <button> sang “light” (sử dụng `setAttribute()`), nội dung văn bản của nó sang “Lighten”, và background-color của overlay <div> sang “`rgba(0,0,0,0.5)`”.

```
if (btnClass === "dark") {  
    btn.setAttribute("class", "light");  
    btn.textContent = "Lighten";  
    overlay.style.backgroundColor = "rgba(0,0,0,0.5)";  
}
```

- Nếu tên lớp không phải “dark”: đổi lớp của <button> sang “dark”, nội dung văn bản của nó sang “Darken”, và background-color của overlay <div> sang “`rgba(0,0,0,0)`”.

```
else {  
    btn.setAttribute("class", "dark");  
    btn.textContent = "Darken";  
    overlay.style.backgroundColor = "rgba(0,0,0,0)";  
}
```

### Nội dung toàn bộ file main.js

```
// 20521150 - Phạm Quốc Cường  
/* Looping through images */  
  
for (let i = 1; i <= 5; i++) {  
    const newImage = document.createElement("img");  
    newImage.setAttribute("src", "images/pic" + i + ".jpg");  
    thumbBar.appendChild(newImage);  
    newImage.onclick = function (e) {  
        displayedImage.src = e.target.src;  
    };  
}
```

```

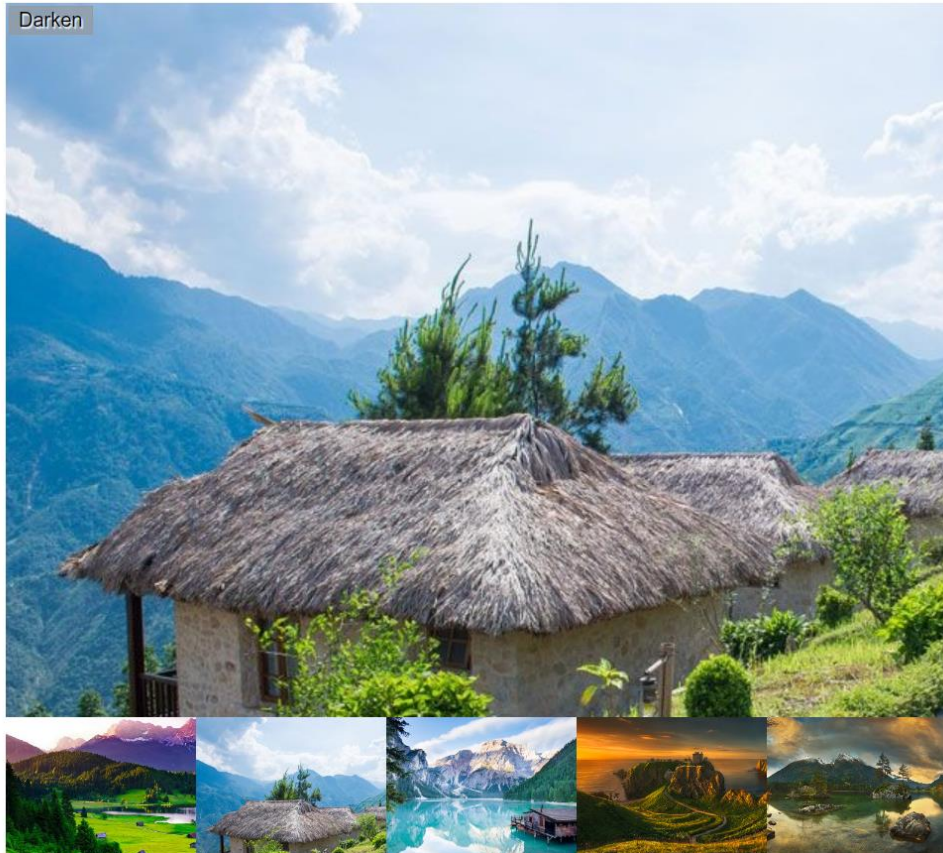
/* Wiring up the Darken/Lighten button */

btn.onclick = function () {
  const btnClass = btn.getAttribute("class");
  if (btnClass === "dark") {
    btn.setAttribute("class", "light");
    btn.textContent = "Lighten";
    overlay.style.backgroundColor = "rgba(0,0,0,0.5)";
  } else {
    btn.setAttribute("class", "dark");
    btn.textContent = "Darken";
    overlay.style.backgroundColor = "rgba(0,0,0,0)";
  }
};

```

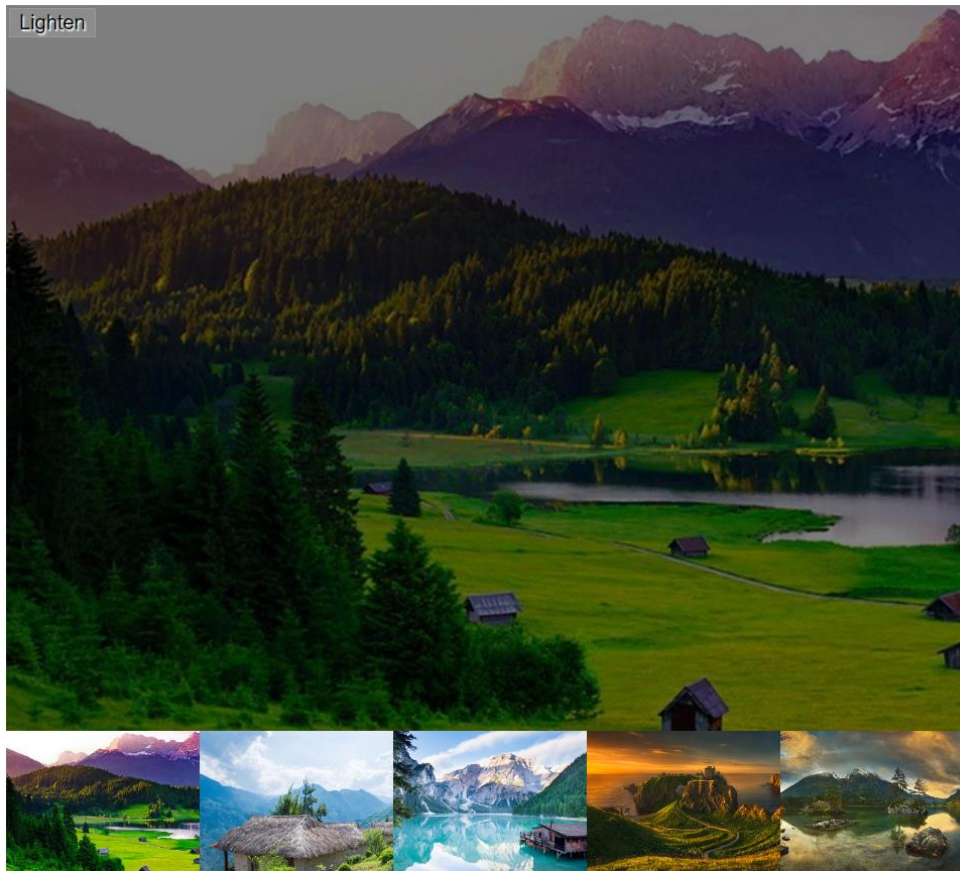
### Kết quả sau thực hiện:

- Hiển thị danh sách các file ảnh





- Chuyển màu sang tối / sáng



HẾT

## PHỤ LỤC

1. Github: [pzcuong/IE104-Internet-and-web-technology-: IE104 - Internet và công nghệ Web \(github.com\)](https://github.com/pzcuong/IE104-Internet-and-web-technology-)