

[Nhấn vào đây để tải layout bài tập](#)

## 1 Xây dựng ứng dụng đổi màu ngôi nhà

Link demo: <https://change-color-house.vercel.app/>

### 1. Load danh sách ô màu lên UI

Khi trang web hiện lên, hàm load các ô màu sẽ được gọi để thêm các thẻ button cho `colorContainer`

- Mảng màu:

```
const colorList = ["pallet", "viridian", "pewter", "cerulean", "vermillion", "lavender", "celadon", "saffron", "fuschia", "cinnabar"];
```

- Sử dụng arrow function tạo hàm load danh sách các ô màu:

html sau khi load các ô màu:

```
</p>
<div id="colorContainer" class="color-button-container"> flex
  <button class="color-button pallet active">...</button> == $0
  <button class="color-button viridian">...</button>
  <button class="color-button pewter">...</button>
  <button class="color-button cerulean">...</button>
  <button class="color-button vermillion">...</button>
  <button class="color-button lavender">...</button>
  <button class="color-button celadon">...</button>
  <button class="color-button saffron">...</button>
  <button class="color-button fuschia">...</button>
  <button class="color-button cinnabar">...</button>
</div>
```

### 2. Đổi màu ngôi nhà

- Sử dụng arrow function tạo hàm đổi màu ngôi nhà
- Khi click vào ô màu, copy class color của ô màu và gắn vào thẻ id `house`

```
<div class="container">
  <div id="house" class="house viridian">...</div> == $0
</div>
<div class="color-controls">
  <p>
    Select a colour
  </p>
  <div id="colorContainer" class="color-button-container"> flex
    <button class="color-button pallet">...</button>
    <button class="color-button viridian active">...</button>
    <button class="color-button pewter">...</button>
```

## 2 Tính điểm trung bình

Link demo: <https://calculate-average-es-6.vercel.app/>

Viết chương trình tính điểm trung bình cho cả 2 khối lớp 1 và 2. Cả 2 khối lớp dùng chung 1 hàm tính điểm trung bình

- Sử dụng arrow function xây dựng hàm tính điểm trung bình
- Dùng Rest Parameter cho tham số của hàm tính điểm trung bình
- Hiện điểm trung bình lên UI như hình bên dưới

The screenshot displays a web application with two columns for calculating average scores. The left column is for 'Khối lớp 1' (Class 1) and the right column is for 'Khối lớp 2' (Class 2). Each column contains three input fields for subjects and their scores, a 'Tính' (Calculate) button, and a large box showing the final average score.

Khối lớp 1	Khối lớp 2
Toán: 8	Văn: 5
Lý: 8	Sử: 5
Hóa: 8	Địa: 5
English: 5	
<b>Tính</b>	<b>Tính</b>
8.00	5.00

## 3 Xây dựng hiệu ứng Jump text khi hover

Link demo: <https://jump-text.vercel.app/>

Khi hover vào từng chữ, chữ sẽ được phóng to.

HTML khi chưa có hiệu ứng:

```
<h2 class="heading">Hover Me!</h2>
```

Hover Me!

HTML sau khi xây dựng hiệu ứng

```
<h2 class="heading">
  <span>H</span>
  <span>o</span>
  <span>v</span>
  <span>e</span>
  <span>r</span>
  <span>M</span> == $0
  <span>e</span>
  <span>!</span>
</h2>
```

HoverMe!

- Sử dụng arrow function và xây dựng hàm thêm hiệu ứng jump text
- Lấy nội dung text "Hover Me!" của class `heading`
- Sử dụng Spread Operator để tách từng ký tự của text "Hover Me!" => `["H","o","v","e","r","M","e","!"]`
- Tạo các thẻ span chứa các ký tự chữ sau khi đã tách được và thêm các thẻ span đó vào thẻ `heading`
- Code mẫu sử dụng Spread Operator để tách từng ký tự

```
let str = "foo"
let chars = [ ...str ]
console.log(chars);
//kết quả: (3) ["f", "o", "o"]
```