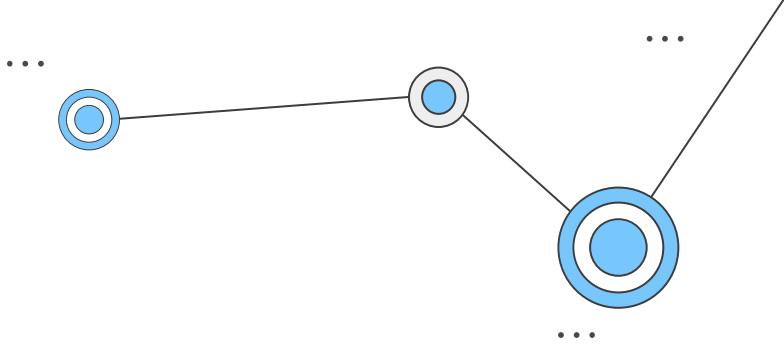
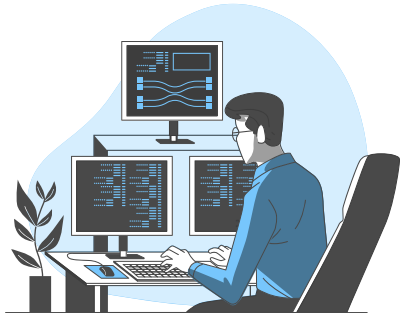
JS



Khóa học Backend

Bài 10: Javascript cơ bản (Tiết 6)



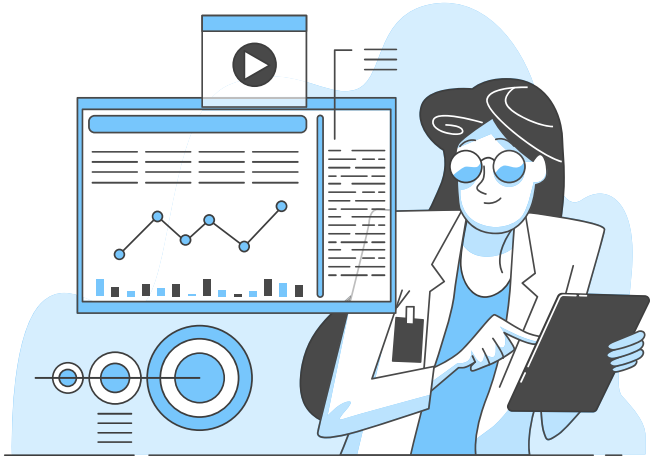
Nội dung

01 DOM Events

02 DOM Navigation

03 DOM Events Listener

04 DOM Nodes



01

DOM Events

01. DOM Events

- **Dom Event** (sự kiện) là một **tác động** nào đó lên **thẻ HTML**.
- Để **bắt được sự kiện** và **thực thi một chương trình**.
- Cú pháp:

```
element.eventname = function () {
  // Code
}
```

- Trong đó:
  - **element**: là phần tử muốn bắt sự kiện.
  - **eventname**: Tên sự kiện.

01. DOM Events

- **Danh sách** một số DOM Events phổ biến:

Sự kiện	Mô tả
onload	Khi trình duyệt đã load xong mọi thứ (image, js, css) thì những đoạn code nằm bên trong đó mới được chạy.
onblur	Kích hoạt khi một phần tử mất trọng tâm (không được focus vào nữa).
onfocus	Kích hoạt khi một phần tử hiển thị đúng trọng tâm (được focus vào).
onkeydown	Kích hoạt khi một phím được nhấn.
onkeyup	Kích hoạt khi một phím được nhả ra.
onclick	Kích hoạt trên con chuột vừa nhấn vào phần tử.
onchange	Kích hoạt khi giá trị được thay đổi khác đi so với giá trị trước đó.

02

DOM Events Listener

02. DOM Events Listener

- **DOM Events Listener** giống Dom Event, nhưng khác ở chỗ:
  - **Một element** có thể **gọi được nhiều sự kiện**.
  - Có thể **hủy bỏ lắng nghe sự kiện** bất kỳ (Dom Events chỉ lắng nghe được nhưng không hủy bỏ lắng nghe được).
- Cú pháp:

```
element.addEventListener("eventname", function (e) {
  // Code
});
```

- Trong đó:
  - **element**: là phần tử muốn bắt sự kiện.
  - **eventname**: Tên sự kiện (**bỏ chữ on**, Ví dụ: onclick → click).

03

DOM Navigation

03. DOM Navigation

- Thể hiện **mối quan hệ cha – con** của **các thẻ HTML**.

- Các thuộc tính:
  - **parentNode**: Truy cập phần tử cha.
  - **childNodes**: Truy cập vào các phần tử con.
  - **firstElementChild**: Truy cập vào phần tử con đầu tiên.
  - **lastElementChild**: Truy cập vào phần tử con cuối cùng.
  - **nextElementSibling**: Truy cập vào phần tử kế tiếp.
  - **previousElementSibling**: Truy cập vào phần tử trước đó.
  - **nodeName**: Lấy ra tên node.

04

DOM Nodes

4.1. document.createElement()

- Tạo ra một phần tử HTML.
- Cú pháp:

```
var tenBien = document.createElement("tagName");
```

- Trong đó:
  - **tagName**: là tên thẻ (Ví dụ: p, h1, div,...).

4.2. document.createTextNode()

- Tạo ra một chuỗi văn bản.
- Cú pháp:

```
var tenBien = document.createTextNode("Nội dung…");
```

4.3. appendChild()

- Dùng để thêm vào vị trí cuối cùng của một thẻ HTML khác.
- Cú pháp:

```
element_parent.appendChild(node_insert);
```

- Trong đó:
  - **element_parent**: là phần tử cha.
  - **node_insert**: là node bạn muốn thêm vào.

4.4. insertBefore()

- Dùng để **thêm một Node** vào **đằng trước một node** con nào đó.
- Cú pháp:

```
element_parent.insertBefore(node_insert, node_child);
```

- Trong đó:
  - **element_parent**: là phần tử cha.
  - **node_insert**: là node bạn muốn thêm vào.
  - **node_child**: là node con mà bạn muốn thêm node_insert vào đằng trước nó.

4.5. removeChild()

- Dùng để xóa một node con ra khỏi node cha.
- Cú pháp:

```
element_parent.removeChild(node_remove);
```

- Trong đó:
  - **element_parent**: là phần tử cha.
  - **node_remove**: là node cần xóa.

4.6. replaceChild()

- Dùng để thay thế một node con bằng một node mới.
- Cú pháp:

```
element_parent.replaceChild(node_insert, node_remove);
```

- Trong đó:
  - **element_parent**: là phần tử cha.
  - **node_insert**: là node bạn muốn thay thế.
  - **node_remove**: là node cũ muốn bỏ đi.

Bài tập

- **Link bài tập**: https://dacavn.notion.site/B-i-t-p-b-i-10-Javascript-c-b-n-Ti-t-6-d90cc1b024df450c924c8acbf303a9eb?pvs=4