

# Môn học: Lập trình Web

## LAB 02

### CSS3 & JAVASCRIPT

#### I. Mục tiêu

- Sử dụng được các thuộc tính CSS3 thông dụng thường gặp để tạo hiệu ứng cho các phần tử trong tài liệu HTML.
- Hiểu biết về phương pháp thiết kế trang web hỗ trợ hiển thị tốt trên nhiều loại thiết bị (Responsive Web Design).
- Ứng dụng CSS Framework (Bootstrap) trong thiết kế giao diện trang web.
- Hiểu biết về ngôn ngữ lập trình Javascript và cách sử dụng mã Javascript trong trang web.
- Hiểu biết về mô hình đối tượng tài liệu DOM (Document Object Model) và các phương thức cơ bản được hỗ trợ trong Javascript dùng để tương tác (truy xuất và chỉnh sửa) các phần tử trong DOM.
- Ứng dụng Javascript để tạo sự tương tác cho các phần tử trong trang web.

#### II. Các bước thực hiện

**Chú ý:** các file liên quan đến bài Lab và tài liệu tham khảo được đặt trong thư mục “refs”

##### A. Phần 1

##### 1. **Tìm hiểu CSS3 và các thuộc tính CSS3 thông dụng**

- CSS3 là phiên bản nâng cấp từ phiên bản CSS2.1, CSS3 hỗ trợ nhiều hiệu ứng giao diện mới cho các phần tử của tài liệu HTML như shadows, gradients, transitions or animations,...

##### 2. **Responsive Web Design**

- **Meta Viewport:** thiết lập cho trình duyệt hiển thị tương ứng với kích thước màn hình thiết bị. Đặt thẻ meta bên dưới trong phần **head** của tài liệu HTML.

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

- **CSS Media Query:** khai báo CSS3 cho từng mức chiều rộng của thiết bị  
**Tham khảo:**

```
/*===== Mobile First Method =====*/
/* Custom, iPhone Retina */
@media only screen and (min-width : 320px) {
    ...
}
/* Extra Small Devices, Phones */
@media only screen and (min-width : 480px) {
    ...
}
/* Small Devices, Tablets */
@media only screen and (min-width : 768px) {
    ...
}
```

```
/* Medium Devices, Desktops */
@media only screen and (min-width : 992px) {
  ...
}
/* Large Devices, Wide Screens */
@media only screen and (min-width : 1200px) {
  ...
}

/*===== Non-Mobile First Method =====*/
/* Large Devices, Wide Screens */
@media only screen and (max-width : 1200px) {
  ...
}
/* Medium Devices, Desktops */
@media only screen and (max-width : 992px) {
  ...
}
/* Small Devices, Tablets */
@media only screen and (max-width : 768px) {
  ...
}
/* Extra Small Devices, Phones */
@media only screen and (max-width : 480px) {
  ...
}
/* Custom, iPhone Retina */
@media only screen and (max-width : 320px) {
  ...
}
```

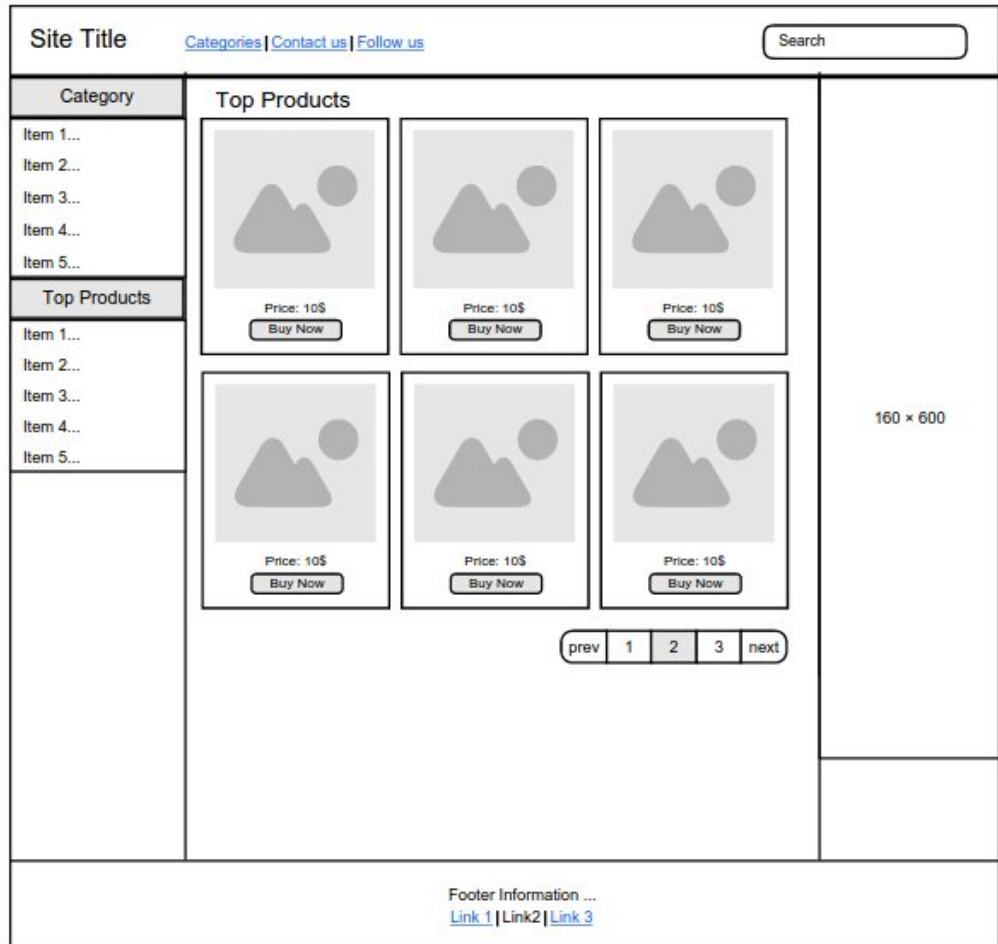
### 3. **CSS Bootstrap Framework**

- **Bootstrap** là một CSS Framework (thư viện bao gồm các mẫu HTML, các lớp CSS, hàm javascript được khai báo sẵn) hỗ trợ thiết kế trang web hiển thị tốt trên nhiều loại thiết bị khác nhau (responsive web design).
- Sinh viên tham khảo CSS Bootstrap Framework và cách sử dụng tại:  
<https://getbootstrap.com/css/>
- Tận dụng các thuộc tính CSS3 đã biết để tạo hiệu ứng cho các phần tử của trang web.
- Tham khảo các loại thư viện/framework CSS:  
<https://github.com/uhub/awesome-css>
- Cách chọn màu sắc có thể tham khảo tại:  
<https://paletton.com>  
<https://www.colorcombos.com/>

## ❖ Bài tập phần 1

- **Làm tại lớp bài trắc nghiệm ôn tập (ở trang lý thuyết của môn học trên Bkel): Bài kiểm tra số 5**

**Bài 1. (thư mục phan1\_bai1) - Làm và nộp tại lớp -** Dùng ngôn ngữ HTML5 và CSS3 để tạo ra một trang web có bố cục như hình bên dưới.



➤ **Các bước thực hiện:**

**Bước 1:**

Tạo layout tổng quát cho trang web, xác định số hàng, số cột trong layout:

- Layout trang web như hình có bao nhiêu hàng?
- Trong mỗi hàng có bao nhiêu cột?

**Bước 2:**

Viết mã HTML5 theo cấu trúc dựa trên các quy tắc cơ bản:

- Từ ngoài vào trong: tạo các block tổng quát, sau đó tạo các block con bên trong để chia nhỏ.
- Từ trái sang phải, xác định các cột (nếu có) trong hàng và sử dụng một trong các cách để chia cột dữ liệu bằng CSS:
  - float/clear
  - inline-block
  - flexbox
  - grid

**Bước 3:**

Tạo các phần tử (văn bản, hình ảnh, danh sách, liên kết, bảng, phần tử form,...) tương ứng trong các block ở vị trí tương ứng.

**Bước 4:**

Áp dụng hiệu ứng CSS3 cho các phần tử đã thiết kế:

- Responsive web design cho các phần tử
- Chọn tông màu cho trang web và các phần tử
- Áp dụng hiệu ứng CSS3 đã biết cho các phần tử

**Bước 5:**

Kiểm tra lại giao diện của trang web đã thiết kế trên các loại trình duyệt web khác nhau

Kiểm tra và sửa lỗi cú pháp cho mã HTML5/CSS3 ở công cụ online:

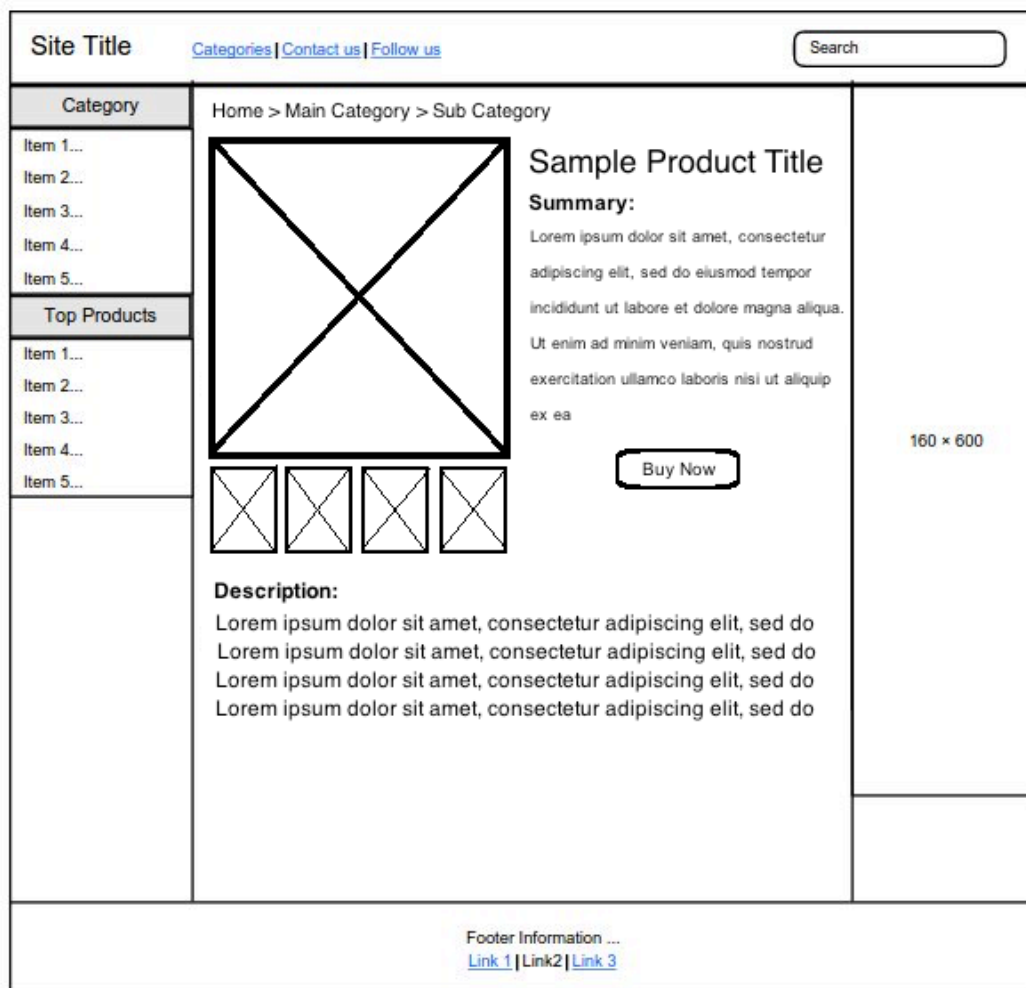
<http://validator.w3.org>

**Chú ý:** áp dụng các bước trên trong trường hợp tự viết mã HTML5/CSS3. Trong trường hợp sử dụng các CSS framework, sinh viên tham khảo và sử dụng các CSS class được định nghĩa trong framework tương ứng cho từng bước.

➤ **Yêu cầu:**

- Tất cả mã HTML5/CSS3 phải được kiểm tra và sửa lỗi ở: <http://validator.w3.org>
- Trang web phải hỗ trợ hiển thị giống nhau trên các trình duyệt phổ biến hiện nay (Firefox, Chrome, IE, Safari,...).
- **Responsive Web Design:** Trang web hỗ trợ hiển thị tốt trên nhiều loại thiết bị khác nhau (máy tính, máy tính bảng, điện thoại,...).
- Sinh viên có thể sử dụng CSS Bootstrap Framework.
- Sinh viên hiện thực mã nguồn vào 2 files:
  1. **index.html:** chứa mã HTML5
  2. **style.css:** chứa mã CSS3 dùng để định dạng các phần tử trong tài liệu HTML5.

**Bài 2. (thư mục phan1\_bai2)** Dùng ngôn ngữ HTML5 và hiệu ứng và CSS3 để tạo ra một trang web có bố cục như hình bên dưới.



➤ **Các bước thực hiện:**  
Tương tự bài 1

➤ **Yêu cầu:**

- Tất cả mã HTML5/CSS3 phải được kiểm tra và sửa lỗi ở: <http://validator.w3.org>
- Trang web phải hỗ trợ hiển thị giống nhau trên các trình duyệt phổ biến hiện nay (Firefox, Chrome, IE, Safari,...).
- **Responsive Web Design:** Trang web hỗ trợ hiển thị tốt trên nhiều loại thiết bị khác nhau (máy tính, máy tính bảng, điện thoại,...).
- Sinh viên có thể sử dụng CSS Bootstrap Framework.
- Sinh viên hiện thực mã nguồn vào 2 files:
  1. **index.html:** chứa mã HTML5
  2. **style.css:** chứa mã CSS3 dùng để định dạng các phần tử trong tài liệu HTML5.

## B. Phần 2

### 1. **Tìm hiểu ngôn ngữ lập trình Javascript**

- Javascript là một ngôn ngữ lập trình kịch bản được dùng để tạo ra tương tác cho các trang web. Mã Javascript được thực thi trong trình duyệt của người dùng.
- Để nhúng mã Javascript vào tài liệu HTML, chúng ta có 2 cách:

**Cách 1:** viết mã Javascript trực tiếp vào tài liệu HTML

```
<script type="text/javascript">  
    //Mã Javascript  
</script>
```

**Cách 2:** dùng một tập tin chứa mã Javascript bên ngoài

```
<script src="demo.js" type="text/javascript"></script>
```

### 2. **Các phương thức, cấu trúc cơ bản của ngôn ngữ Javascript**

- Tìm hiểu cách thức khai báo biến, các cấu trúc, đối tượng và các phương thức cơ bản của đối tượng trong ngôn ngữ Javascript, tham khảo file: **javascript\_tutorial.pdf**
- Thực hành sử dụng các phương thức Javascript cơ bản bằng cách chạy các đoạn mã ví dụ được cho trong file: **basic\_javascript.html**.
- Tham khảo một số đoạn mã Javascript thông dụng ở trang web:  
View source các trang để xem mã ví dụ  
<http://www-k12.atmos.washington.edu/~ovens/javascript/jsindex.html>
- Tìm hiểu các đối tượng cơ bản BOM (Browser Object Model):  
[http://www.w3schools.com/jsref/obj\\_window.asp](http://www.w3schools.com/jsref/obj_window.asp)
  - Window
    - open()
    - close()
    - alert()
    - confirm()
    - prompt()
  - Navigator
  - Screen
  - History
  - Location
- Tìm hiểu các đối tượng cơ bản DOM (Document Object Model):  
[http://www.w3schools.com/jsref/dom\\_obj\\_document.asp](http://www.w3schools.com/jsref/dom_obj_document.asp)
  - DOM Document
  - DOM Elements
  - DOM Attributes
  - DOM Events
  - DOM Style

### 3. Các phương thức cơ bản dùng để truy xuất các phần tử trong DOM

- Tìm hiểu các thuộc tính, phương thức cơ bản dùng để truy xuất, chỉnh sửa các phần tử trong DOM, đọc file: **DOM.html** đính kèm trong bài Lab
- Tìm hiểu cách thức sử dụng các phương thức thông dụng  
[http://www.w3schools.com/jsref/dom\\_obj\\_document.asp](http://www.w3schools.com/jsref/dom_obj_document.asp)
  - document.cookie
  - document.createAttribute()
  - document.createElement()
  - document.createTextNode()
  - document.getElementById()
  - document.getElementsByTagName()
  - document.getElementsByClassName()
  - document.getElementsByName()
  - document.getElementsByTagName()
  - document.querySelector()
  - document.querySelectorAll()
  - document.images
  - document.title
  - document.links
  - document.write()

### ❖ Bài tập phần 2

**Bài 1. (thư mục phan2\_bai1)** Dùng các phương thức Javascript trong DOM (tạo, xóa, sửa các phần tử DOM) để tạo và chỉnh sửa động một HTML table.

**Yêu cầu:**

- Nút tạo mới 1 HTML table (2x2)
- Nút thêm 1 hàng vào HTML table có sẵn
- Nút thêm 1 cột vào HTML table có sẵn
- Nút xóa 1 hàng (1 cột) khỏi HTML table có sẵn (chỉ số hàng/cột cần xóa được nhập vào bởi người dùng thông qua phần tử input text)
- Nút xóa toàn bộ HTML table có sẵn

**Bài 2. (thư mục phan2\_bai2)** Hiện thực một chương trình máy tính cơ bản sử dụng HTML Form và ngôn ngữ Javascript, chương trình hỗ trợ các phép tính cơ bản (cộng, trừ, nhân, chia, lũy thừa), các giá trị được người dùng nhập vào thông qua HTML Form.

**Bài 3. (thư mục phan2\_bai3)** Hiện thực một Form đăng kí thành viên đơn giản, yêu cầu những thông tin dưới đây từ người sử dụng:

- First name (chuỗi từ 2-30 kí tự).
- Last name (chuỗi từ 2-30 kí tự).
- Email (theo định dạng email: <sth>@<sth>.<sth>).
- Password (chuỗi từ 2-30 kí tự, dùng phần tử password trong HTML Form).

- Birthday (ngày, tháng, năm. Dùng phần tử select).
- Gender (nam, nữ, không xác định. Dùng phần tử radio).
- Country (dùng phần tử select, có một số lựa chọn: Vietnam, Australia, United States, India, Other).
- About (phần tử textarea, giới hạn 10000 kí tự).
- Nút Submit và Reset.

**Yêu cầu:**

- Tất cả các dữ liệu nhập từ người dùng đều phải được kiểm tra sử dụng ngôn ngữ **Javascript** theo yêu cầu cho mỗi trường liệt kê ở trên.
- **Không sử dụng các validator có sẵn trong HTML5.**
- Hiện thị thông báo lỗi (dùng hộp thoại **alert**) trong trường hợp người dùng nhập dữ liệu không đúng quy định.
- Hiện thị thông báo “Complete!” bằng hộp thoại **alert** trong trường hợp tất cả dữ liệu người dùng nhập đều đúng quy định.

**III. Cách thức nộp bài**

- Bài Lab 02 bao gồm 2 phần:
  - **Phần làm và nộp tại lớp** (bài 1 của phần 1): sinh viên làm và nộp tại lớp tại mục “Bài nộp tại lớp (Lab 02)”, tạo thư mục chứa thư mục bài làm (phan1\_bai1), đặt tên thư mục theo MSSV, sau đó nén thư mục MSSV vào một file **zip** (<mssv>.zip) và nộp.
  - **Phần bài nộp sau** (các bài tập còn lại): sinh viên tạo thư mục chứa các thư mục bài làm của các phần còn lại, đặt tên thư mục theo MSSV, sau đó nén thư mục MSSV vào một file **zip** (<mssv>.zip) và nộp ở Bkel:  
<http://e-learning.hcmut.edu.vn> trong trang của môn học tương ứng.
- Các bài nộp sai quy định sẽ không được tính điểm.
- Các bài làm giống nhau sẽ bị xem là gian lận và bị 0 điểm,
- Sinh viên theo dõi deadline nộp bài lab và nộp bài đúng hạn để được tính điểm, mọi trường hợp nộp bài trễ hạn sẽ không được giải quyết.
- **Chỉ nhận bài nộp thông qua Bkel, không nhận bài nộp qua email hay các hình thức khác.**

--HẾT--