

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 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 – CSS3 & Responsive Web Design

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, Media Queries, Pseudo-Classes,...

2. **Áp dụng Responsive Web Design**

- **Meta Viewport:** thiết lập cho trình duyệt hiển thị tương xứng 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. Kiểm thử giao diện Responsive của trang web

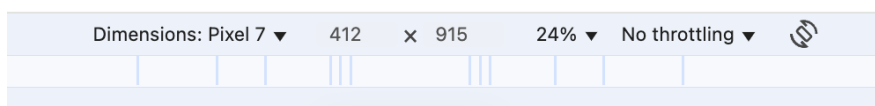
Mở trình duyệt Chrome, vào trang web muốn kiểm tra

Mở công cụ Inspector

Chọn vào thanh công cụ thiết bị (Toggle Device Toolbar) - phím tắt Ctrl + Shift + M như hình



Tại thanh công cụ thiết bị (Device Toolbar), ta có thể chọn thiết bị để giả lập, hoặc tùy chọn thay đổi chiều rộng / chiều cao của viewport



4. CSS Bootstrap Framework

- **Bootstrap** là CSS framework phổ biến, ra mắt lần đầu năm 2011, giúp xây dựng trang web hỗ trợ responsive nhanh chóng. Version 5 mới nhất đã loại bỏ jQuery, chuyển sang Vanilla JavaScript, và cải thiện mạnh mẽ hệ thống lưới (grid system). Tính năng nổi bật của nó là thư viện các thành phần giao diện có sẵn, đảm bảo tính đồng nhất và tiết kiệm thời gian đáng kể cho lập trình viên.
- Các lý do nên sử dụng Bootstrap
 - Tiết kiệm thời gian.
 - Dễ sử dụng.
 - Tùy chỉnh dễ dàng.
 - Responsive.
 - Tương thích với nhiều trình duyệt.
 - Nguồn mở.
- Sinh viên tham khảo CSS Bootstrap Framework và cách sử dụng tại: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
 - **Cách cài đặt Bootstrap**
 - Tải trực tiếp file thư viện
 - Sử dụng CDN

Grid options

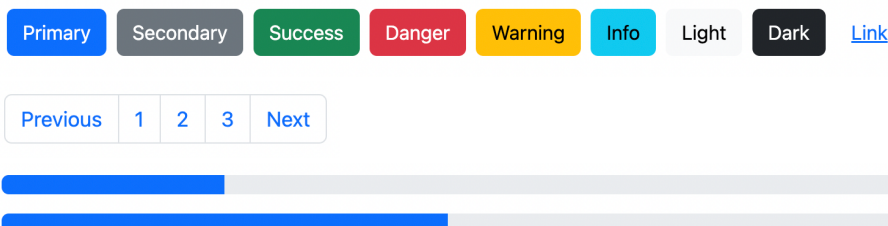
Bootstrap's grid system can adapt across all six default breakpoints, and any breakpoints you customize. The six default grid tiers are as follows:

- Extra small (xs)
- Small (sm)
- Medium (md)
- Large (lg)
- Extra large (xl)
- Extra extra large (xxl)

As noted above, each of these breakpoints have their own container, unique class prefix, and modifiers. Here's how the grid changes across these breakpoints:

	xs <576px	sm ≥576px	md ≥768px	lg ≥992px	xl ≥1200px	xxl ≥1400px
Container <i>max-width</i>	None (auto)	540px	720px	960px	1140px	1320px
Class prefix	.col-	.col-sm-	.col-md-	.col-lg-	.col-xl-	.col-xxl-
# of columns	12					
Gutter width	1.5rem (.75rem on left and right)					
Custom gutters	Yes					
Nestable	Yes					
Column ordering	Yes					

Form và các Components khác



- Tận dụng các thuộc tính CSS3 đã biết để tạo hiệu ứng cho các phần tử của trang web.

5. Các thư viện/framework CSS/CSS3 khác:

- Tham khảo:
<https://github.com/uuhub/awesome-css>
- Cách chọn màu sắc có thể tham khảo các công cụ online:
<https://paletton.com>
<https://www.colorcombos.com/>
<https://flatuicolors.com/>

❖ Bài tập phần 1

Bài 1. (thư mục phan1_bai1) Chính sửa lại bài tập sở Lab 01 (bài 6, phần 2), áp dụng CSS media query để tạo ra:

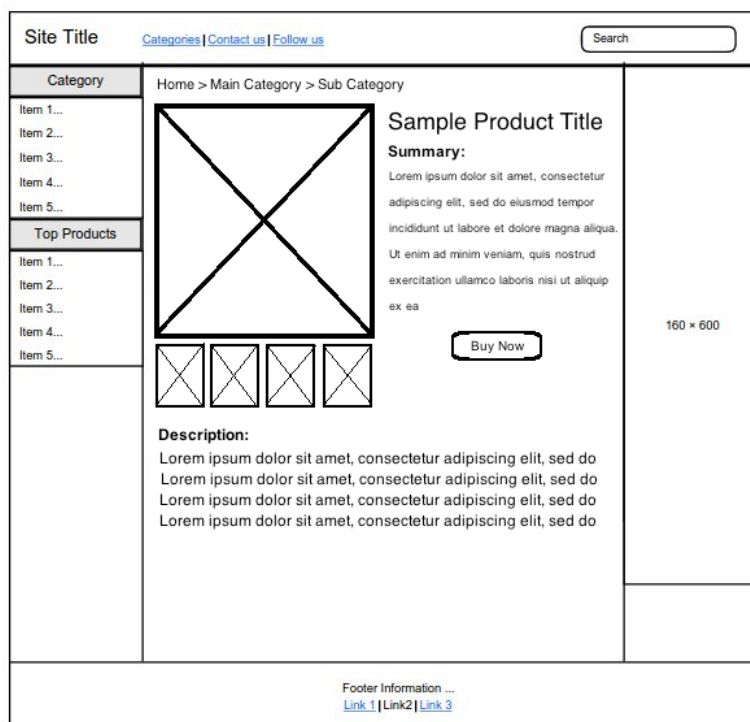
Giao diện web đáp ứng trên màn hình nhỏ (điện thoại di động) như hình **page_screenshot_mobile.png**, với menu của trang web được thiết kế như hình **page_screenshot_mobile_side_bar.png**

Giao diện web đáp ứng trên màn hình cỡ vừa (máy tính bảng, SV tự thiết kế sau cho phù hợp, giúp tối ưu hóa trải nghiệm người dùng).

Trợ giúp:

- Tham khảo các mẫu menu đơn giản:
 - https://www.w3schools.com/howto/tryit.asp?filename=tryhow_js_topnav
 - File **side_bar_mobile.html** trong thư mục **refs**

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 các hình bên dưới một trong các CSS framework đã được giới thiệu.



➤ **Yêu cầu:**

- Tất cả mã HTML5/CSS3 phải được kiểm tra và sửa lỗi ở công cụ trên trang web: <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 tự chọn CSS 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.
- Nộp bài kèm theo các file liên quan (thư viện, hình ảnh,...) nếu có sử dụng

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
 - 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
 - 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
 - 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 xây dựng một ứng dụng web đơn giản cho phép người dùng tương tác để quản lý danh sách công việc (To-Do List Manager). Sinh viên phải sử dụng JavaScript để thao tác linh hoạt với DOM (Document Object Model), biến giao diện tĩnh thành một công cụ tương tác.

Yêu cầu:

Thiết kế giao diện cho chương trình sử dụng một Framework CSS tự chọn. Ứng dụng phải thực hiện ba chức năng chính, mỗi chức năng yêu cầu sử dụng các phương thức DOM nâng cao:

1. Thêm Công Việc Mới:

- Sử dụng **truy vấn DOM** để lấy nội dung từ ô nhập liệu và bắt **sự kiện** (nhấp chuột/nhấn Enter).
- Sử dụng **document.createElement()** để tạo ra các phần tử HTML mới (, <button>).
- Sử dụng **parentElement.appendChild()** để chèn các phần tử mới tạo vào danh sách (thẻ) trên giao diện.

2. Đánh Dấu Hoàn Thành:

- Gán **sự kiện** nhấp chuột cho mỗi mục việc () được tạo ra.
- Sử dụng **element.classList.toggle('completed')** để luân phiên thêm hoặc xóa một class CSS đặc biệt, từ đó thay đổi kiểu hiển thị (ví dụ: gạch ngang chữ) cho biết công việc đã hoàn thành.

3. Xóa Công Việc:

- Gán **sự kiện** nhấp chuột cho nút "Xóa" riêng biệt trong mỗi mục việc.
- Sử dụng **event.stopPropagation()** để ngăn sự kiện xóa kích hoạt chức năng đánh dấu hoàn thành.
- Sử dụng phương thức **element.remove()** để xóa vĩnh viễn mục việc đó khỏi cấu trúc DOM và giao diện.

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.

Các bước thực hiện:

- Thiết kế giao diện cho chương trình sử dụng một Framework CSS tự chọn
- Hiện thực tính năng cho chương trình bằng ngôn ngữ javascript (áp dụng sự kiện trên các nút nhấn, lấy giá trị từ form, kiểm tra dữ liệu đầu vào theo yêu cầu, xử lý và xuất dữ liệu)

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 checkbox xác nhận người dùng đã đồng ý với các điều khoản của trang web.
- 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.**
- Hiển thị thông báo lỗi trong trường hợp người dùng nhập dữ liệu không đúng quy định.
- Hiển thị thông báo “Complete!” trong trường hợp tất cả dữ liệu người dùng nhập đều đúng quy định.

Các bước thực hiện:

- Thiết kế giao diện cho chương trình sử dụng một Framework CSS tự chọn
- Hiện thực tính năng cho chương trình bằng ngôn ngữ javascript (áp dụng sự kiện trên các nút nhấn, lấy giá trị từ form, kiểm tra dữ liệu đầu vào theo yêu cầu, xử lý và xuất dữ liệu)

III. Cách thức nộp bài

- Bài Lab 02 bao gồm 2 phần:
 - **Phần làm và nộp tại lớp:** sinh viên làm và nộp tại lớp tại mục “Bài nộp tại lớp (Lab 01)”:
 - Các bài tập bao gồm:
 - **Phần 1: bài 1**
 - **Phần 2: bài 1**
 - Tạo thư mục chứa các thư mục bài làm (phan1_bai1), đặt tên thư mục theo MSSV, sau đó nén thư mục trên vào một file **zip** (<mssv>.zip) và nộp.
 - **Phần bài nộp sau:**
 - Các bài tập còn lại, bao gồm:
 - **Phần 1: bài 2**
 - **Phần 2: bài 2, bài 3**
 - Tạo thư mục chứa các thư mục bài làm của các phần còn lại (phan1_bai2, phan2_bai1, phan2_bai2, phan2_bai3), đặt tên thư mục theo MSSV, sau đó nén thư mục trên vào một file **zip** (<mssv>.zip) và nộp ở LMS: <https://lms.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 bất thường 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 LMS, không nhận bài nộp qua email hay các hình thức khác.**



--HẾT--