

BÀI THỰC HÀNH TUẦN 3: CSS

Mô tả

Mục đích của bài thực hành

- Cung cấp cho sinh viên những kiến thức cơ bản về CSS
- Cách định dạng trang web với CSS

Nội dung bài thực hành

- Phần 1 – CSS – Cascading Style Sheets
- Phần 2 – Định dạng trang web với CSS
- Bài tập

Yêu cầu

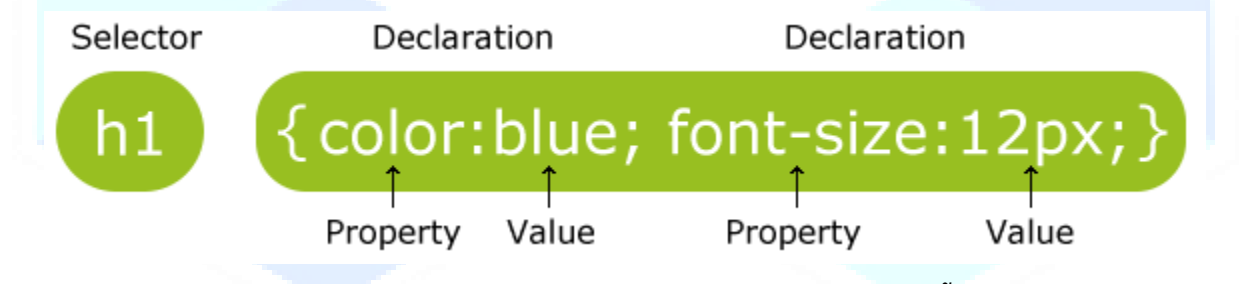
Yêu cầu phải thực hiện trước và sau khi làm bài thực hành

- Đọc phần bài giảng và các phần liên quan trong tài liệu tham khảo của môn học
- Làm bài tập

Phần 1 – CSS – Cascading Style Sheets

1. Các khái niệm cơ bản

CSS được dùng để định nghĩa kiểu cho các elements của HTML (hoặc XHTML). Với CSS, bạn sẽ tiết kiệm được thời gian và công sức mỗi khi cần định nghĩa lại kiểu cho một element. Cú pháp để định nghĩa CSS như sau [1]



Trong đó, selector là HTML element mà bạn muốn áp dụng kiểu. Mỗi kiểu được tạo nên từ nhiều declaration, mỗi declaration gồm 1 property và 1 value tương ứng. Property chính là thuộc tính của kiểu mà bạn muốn thay đổi.

Có 2 loại selector cơ bản được sử dụng trong CSS: id và class.

Id selector được dùng để định nghĩa kiểu cho một element duy nhất, element này sẽ có *id* attribute trùng với tên id của selector, như ví dụ bên dưới.

```
#para1
```

```
{  
    text-align:center;  
    color:red;  
}
```

Lưu ý, không cần có khoảng trắng giữa phần property và value. Đoạn CSS trên sẽ được áp dụng vào HTML như sau

```
<p id="para1">Hello World!</p>
```

Lúc này chữ Hello World! sẽ có màu đỏ và được canh giữa. Lưu ý, tên của id selector (tính từ sau dấu #) không được bắt đầu bằng số vì trình duyệt Firefox sẽ không thể hiển thị được.

Class selector được dùng để định nghĩa kiểu cho một nhóm các elements thông qua *class* attribute của các elements này. Ví dụ như sau

```
.center  
{  
    text-align:center;  
}
```

Đoạn CSS trên sẽ được áp dụng vào HTML như sau

```
<p class="center">Center paragraph</p>
```

Lúc này các chữ trong paragraph sẽ được canh giữa. Lưu ý, tên của class selector (tính từ sau dấu chấm (.)) không được bắt đầu bằng số vì cách định nghĩa này chỉ hiển thị được trên Internet Explorer.

Có 3 cách để chèn đoạn CSS vào HTML:

- Dùng file CSS bên ngoài
- Định nghĩa CSS ngay trong file HTML
- Định nghĩa trong từng dòng HTML

Với cách đầu tiên, giả sử bạn có một file CSS có tên lab1.css; lưu ý file này chỉ chứa các định nghĩa CSS, bạn có thể dùng bất cứ chương trình soạn thảo văn bản nào để tạo file này, với phần đuôi mở rộng .css. File HTML sẽ tham khảo đến file CSS này như sau

```
<head>  
    <link rel="stylesheet" type="text/css" href="lab1.css" />  
</head>
```

Với cách thứ 2, bạn có thể định nghĩa CSS trong file HTML như sau

```
<head>  
    <style type="text/css">  
        hr {color:sienna;}  
        p {margin-left:20px;}  
        body {background-image:url("Images/cutecat.gif");}  
    </style>  
</head>
```

Với cách định nghĩa kiểu ngay trong tag của element, bạn sẽ không tận dụng được ưu điểm dùng lại của CSS, như ví dụ sau

```
<p style="color:red;margin-left:20px">This is a paragraph.</p>
```

Vấn đề đặt ra là nếu bạn dùng 2 trong 3 (hoặc cả 3) cách trên trong 1 file HTML thì kiểu nào sẽ được áp dụng? Sinh viên tự tìm hiểu phần này.

CSS cho phép bạn có thể định dạng màu sắc, font chữ, màu nền... cho các HTML elements, sinh viên sẽ tìm hiểu các định dạng này trong quá trình thực hành.

2. CSS Styling

CSS có thể được dùng để định nghĩa kiểu nền, kiểu chữ hay table...

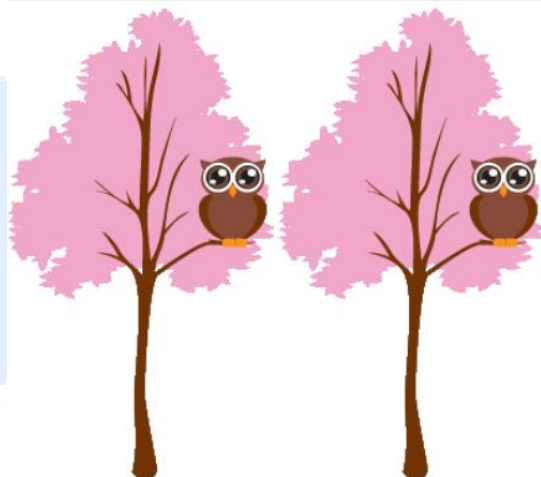
2.1. Background

Các thuộc tính CSS dùng định nghĩa kiểu cho background:

- **Background-color**: xác định màu nền (sinh viên có thể dùng các phần mềm color picker để tiết kiệm thời gian khi chọn màu nền)
- **Background-image**: cài đặt hình ảnh làm nền (hoặc chỉ cần khai báo đơn giản là "background:url(link image)"). Mặc định, hình ảnh sẽ được lặp để phủ đầy element đang được định dạng.
- **Background-repeat**: xác định hình ảnh làm nền sẽ được lặp theo chiều ngang (repeat-x), theo chiều dọc (repeat-y), hoặc không lặp (no-repeat)

Ví dụ:

```
body  
{  
  background-image:url('img_tree.png');  
  background-repeat:repeat-x;  
}
```



- **Background-position**: xác định vị trí của hình nền so với không gian của element chứa nó. Giá trị mặc định là trên cùng góc trái (left top). Bạn cũng có thể xác định giá trị này bằng % hoặc giá trị px

Ví dụ:

```
background-position:left top;  
background-position:25% 75%;  
background-position:10px 200px;
```

- **Background-attachment**: xác định hình nền sẽ được giữ cố định (fixed), cuộn cùng với nội dung của element chứa nó (local) hay cuộn cùng với nội dung của trang (scroll, giá trị mặc định).

Ngoài ra, để rút ngắn định nghĩa kiểu cho background, bạn có thể viết ngắn gọn như sau

```
body {background:#ffffff url('img_tree.png') no-repeat right top;}
```

Với cách viết này, các thuộc tính sẽ được sắp xếp theo thứ tự sau:

- background-color
- background-image
- background-repeat
- background-attachment
- background-position

Ví dụ

```
background:yellow url('img_tree.png') no-repeat top left;
```



2.2. Text

CSS cung cấp các thuộc tính giúp ta định dạng màu chữ, kiểu chữ, font chữ... và nhiều kiểu định dạng khác.

- **Color**: định nghĩa màu chữ
- **Text-align**: canh chữ theo chiều ngang (left, right, center, justify)
- **Text-decoration**: dùng để thiết lập hay bỏ phần trang trí cho chữ

Ví dụ:

```
h1 {text-decoration:overline;}  
h2 {text-decoration:line-through;}  
h3 {text-decoration:underline;}
```

This is heading 1

~~This is heading 2~~

This is heading 3

- Text-transformation: xác định các chữ được viết thường (lowercase), viết hoa (uppercase) hoặc viết hoa chữ đầu tiên của từ (captialize)
- Text-indent: xác định indent cho dòng đầu tiên của đoạn text
- Text-shadow: tạo bóng cho chữ

Ví dụ:

```
h1 {text-shadow:2px 2px #FF0000;}
```

Text-shadow effect

- White-space: xác định cách ngắt dòng trong một đoạn text (sinh viên tự tìm hiểu các giá trị của thuộc tính này)
- Font-family: xác định kiểu font chữ. Bạn có thể xác định cùng lúc nhiều kiểu font, trình duyệt sẽ tự động bỏ qua kiểu font không hỗ trợ và chọn kiểu được khai báo ngay sau đó. Do đó, kiểu font cuối cùng bạn nên chọn kiểu generic. Xem thêm trong tài liệu tham khảo.
- Font-style: xác định text được in nghiêng hay không
- Font-size: kích thước text
- Font-weight: thường dùng để định dạng chữ in đậm

2.3. Links

Ta có thể áp dụng các thuộc tính CSS để định dạng text phía trên cho link. Ngoài ra, ta còn có thể định dạng dựa trên trạng thái của link, ví dụ:

- a:link: trường hợp thường, link chưa được click
- a:visited: link đã được click
- a:hover: người dùng đang trỏ chuột vào link
- a:active: người dùng chọn link (sinh viên có thể xem trạng thái này bằng cách click và giữ chuột trái trên link)

Ví dụ bên dưới định dạng các kiểu khác nhau cho từng trạng thái của link

```

<style>
a.one:link {color:#ff0000;}
a.one:visited {color:#0000ff;}
a.one:hover {color:#ffcc00;}

a.two:link {color:#ff0000;}
a.two:visited {color:#0000ff;}
a.two:hover {font-size:150%;}

a.three:link {color:#ff0000;}
a.three:visited {color:#0000ff;}
a.three:hover {background:#66ff66;}

a.four:link {color:#ff0000;}
a.four:visited {color:#0000ff;}
a.four:hover {font-family:monospace;}

a.five:link {color:#ff0000;text-decoration:none;}
a.five:visited {color:#0000ff;text-decoration:none;}
a.five:hover {text-decoration:underline;}
</style>

```

Trong ví dụ này, khi người dùng trỏ chuột, link1 sẽ chuyển màu, link2 sẽ thay đổi kích thước, link3 sẽ đổi màu nền, link4 đổi font chữ và link5 sẽ được thêm dấu gạch dưới.

2.4. Lists

Ta có thể dùng CSS để định dạng kiểu cho các lists, ví dụ như sau

```

<style>
ul.a {list-style-type:circle;}
ul.b {list-style-type:square;}
ol.c {list-style-type:upper-roman;}
ol.d {list-style-type:lower-alpha;}
</style>
</head>

```

- o Coffee
 - o Tea
 - o Coca Cola
- Coffee
 - Tea
 - Coca Cola
- I. Coffee
 - II. Tea
 - III. Coca Cola
- a. Coffee
 - b. Tea
 - c. Coca Cola

Ngoài ra, ta cũng có thể cài đặt các "marker" (ký tự xuất hiện đầu mỗi dòng trong list) dưới dạng ảnh như ví dụ sau:

```
ul {list-style-image:url('sqpurple.gif');}
```

- Coffee
- Tea
- Coca Cola

Tương tự như định dạng background, ta cũng có cách viết ngắn gọn cho list

```
ul{list-style: square url("sqpurple.gif");}
```

2.5. Table

CSS hỗ trợ định dạng kiểu cho các element chứa trong table (tương tự như những phần trên) và bản thân table (định dạng kiểu khung, đường viền, màu nền...). Ví dụ:

```
table,th,td
{
border:1px dashed grey;
border-collapse:collapse;
background-color:yellow;
color:red;
text-align:center;
vertical-align:top;
width:50%;
height:30px;
}
```

Firstname	Lastname
An	Nguyen Van
Binh	Le Van

Trong ví dụ trên, cạnh table được viền có độ dày 1px, kiểu dash, màu grey; đường viền giữa các ô không có khoảng trống (collapse); màu nền table là màu vàng, chữ đỏ; các text được canh giữa theo chiều ngang và canh trên theo chiều dọc; chiều rộng table là 50% độ rộng của element chứa nó, chiều cao là 30px. Sinh viên sẽ tìm hiểu các thuộc tính CSS khác trong lúc thực hành.

3. CSS Advanced

3.1. Grouping/Nesting

Như đã trình bày ở phần 1, để áp dụng CSS cho một element nào đó, ta sử dụng id và class đã được định nghĩa trước cho element này. Tuy nhiên, ta cũng có thể áp dụng CSS cho một nhóm các element như sau

```
h1,h2,p
{
color:green;
}
```

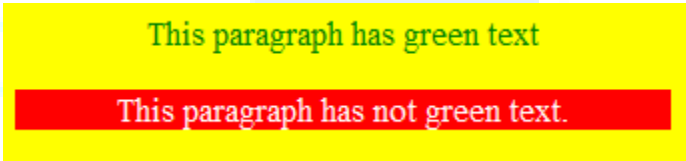
Trong ví dụ trên, bất kỳ text nào được định dạng là h1, h2 hoặc p đều được hiển thị chữ màu xanh. Ta cũng áp dụng cách định nghĩa này cho các element HTML5 như bài thực hành số 2

```
header {
margin:0 auto;
width:913px;
}
figure {
margin:0 auto;
width:913px;
padding: 5px 0px;
}
nav {
margin:0 auto;
background-color: #3195D0;
}
```

Ngoài ra, CSS còn giúp định dạng kiểu cho các element lồng nhau như sau

```
body{
width:50%;
background:yellow;
}
p
{
color:green;
text-align:center;
}
.marked
{
background-color:red;
}
.marked p
{
color:white;
}
```

```
<body>
<p>This paragraph has green text</p>
<div class="marked">
<p>This paragraph has not green text.</p>
</div>
```



3.2. Display

Các thuộc tính display và visibility của CSS thường được dùng để định dạng cách hiển thị của các element. Ví dụ, để ẩn một element, ta có thể sử dụng một trong 2 cách sau


```
h1.hidden {visibility:hidden;}
h1.hidden {display:none;}
```

Ngoài ra, bạn cũng có thể định dạng các elements được hiển thị theo dạng khối (block) hay dạng dòng (inline) như ví dụ sau

```
<style>
li{display:inline;}
li.block{display:block;}
</style>
```

[HTML](#) [CSS](#) [JavaScript](#) [XML](#)

[HTML](#)
[CSS](#)
[JavaScript](#)
[XML](#)

3.3. Position

Thuộc tính position của CSS giúp định dạng vị trí cho các elements. Có 4 cách như sau

- Static: đây là cách mặc định, element sẽ được định dạng theo dòng chảy bình thường của trang
- Fixed: element được định dạng với vị trí cố định trong cửa sổ trình duyệt. Tức là element sẽ giữ nguyên vị trí này kể cả khi scroll cửa sổ trình duyệt.
- Relative: element có vị trí tương đối so với vị trí bình thường của nó. Ví dụ

```
<style>
h3.pos_left
[
position:relative;
left:-30px;
]
h3.pos_right
[
position:relative;
left:30px;
]
</style>
```

Heading with no position

Heading is moved left according to its normal position

Heading is moved right according to its normal position

Lưu ý rằng, mặc dù nội dung của element được dịch chuyển so với vị trí bình thường của nó, nhưng khoảng trống tạo ra (ở vị trí bình thường) của element này vẫn được trình duyệt giữ nguyên. Ví dụ

This heading is moved upwards according to its normal position
This is a heading with no position

Right after the above heading in HTML code

- Absolute: element sẽ được xác định vị trí so với element chứa nó

Ngoài ra, khi các element được định dạng vị trí chồng lên nhau (overlap), ta có thể xác định element nào được đặt trên bằng thuộc tính `z-index`, các element sẽ được đặt theo `z-index` tăng dần (`z-index` cao được đặt phía trên), hoặc theo thứ tự trong HTML code (nếu `z-index` không được định nghĩa).

3.4. Navigation Bar

CSS hỗ trợ bạn tạo các menu (đứng hoặc ngang) bằng cách định dạng cho list như sau

```
<style>
ul
{
  list-style-type:none;
  margin:0;
  padding:0;
}
</style>
</head>
<body>
<ul>
<li><a href="#home">Home</a></li>
<li><a href="#news">News</a></li>
<li><a href="#contact">Contact</a></li>
<li><a href="#about">About</a></li>
</ul>
```

[Home](#)
[News](#)
[Contact](#)
[About](#)

Với menu ngang, bạn có thể dùng thuộc tính `display:inline` hoặc `float` của CSS, tuy nhiên kết quả của 2 cách sử dụng này có khác nhau. Sinh viên tự tìm hiểu điểm khác biệt này.

```

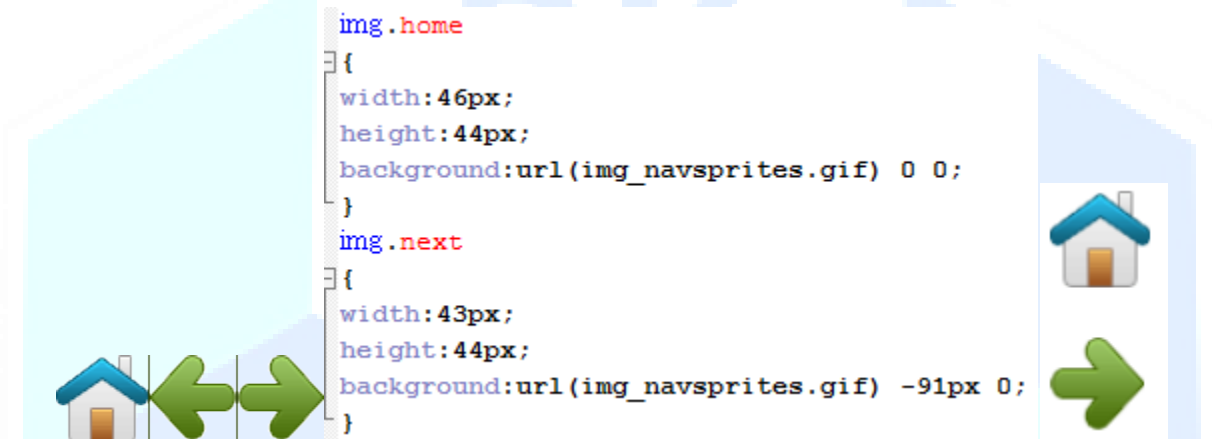
ul
{
  list-style-type:none;
  margin:0;
  padding:0;
  overflow:hidden;
}
li
{
  float:left;
}
a
{
  display:block;
  width:60px;
  background-color:#dddddd;
}

```

[Home](#) [News](#) [Contact](#) [About](#)

3.5. Image Sprites

Một image sprite là một tập hợp nhiều image trong một image duy nhất. Khi trang web sử dụng nhiều image sẽ làm giảm tốc độ load, do đó CSS cho phép ta hiển thị từng phần của một image, giúp giảm thời gian load. Ví dụ như sau



Ngoài ra, CSS còn hỗ trợ định dạng cho các media hay form element, sinh viên tự tìm hiểu phần này.

Phần 2 – Định dạng trang web với CSS

Ở phần này, các bạn sẽ tạo và định dạng một trang web đơn giản bằng HTML và CSS. Đầu tiên, tạo phần khung cho trang web tương tự như ví dụ trong bài HTML Layout như sau

```

<!DOCTYPE html>
<html>
  <body>
    <div id="container">
      <div id="header">

```

```
<h1 style="margin-bottom:0;">Main Title of Web Page</h1>
</div>
<div id="menu">
  <b>Menu</b><br /> HTML<br /> CSS
</div>
<div id="content">Content goes here</div>
<div id="footer">Copyright &copy; W3Schools.com</div>
</div>
</body>
</html>
```

Sau đó ta chèn vào một đoạn CSS trong `<head>` như sau

```
<head>
  <style type="text/css">
    #container
    {
      width:500px;
    }
    #header
    {
      background-color:#FFA500;
    }
    #menu
    {
      background-color:#FFD700;
      height:200px;
      width:100px;
      float:left;
    }
    #content
    {
      background-color:#EEEEEE;
      height:200px;
      width:400px;
      float:left;
    }
    #footer
    {
      background-color:#FFA500;
      clear:both;
      text-align:center;
    }
  </style>
</head>
```

Trang web được hiển thị như sau



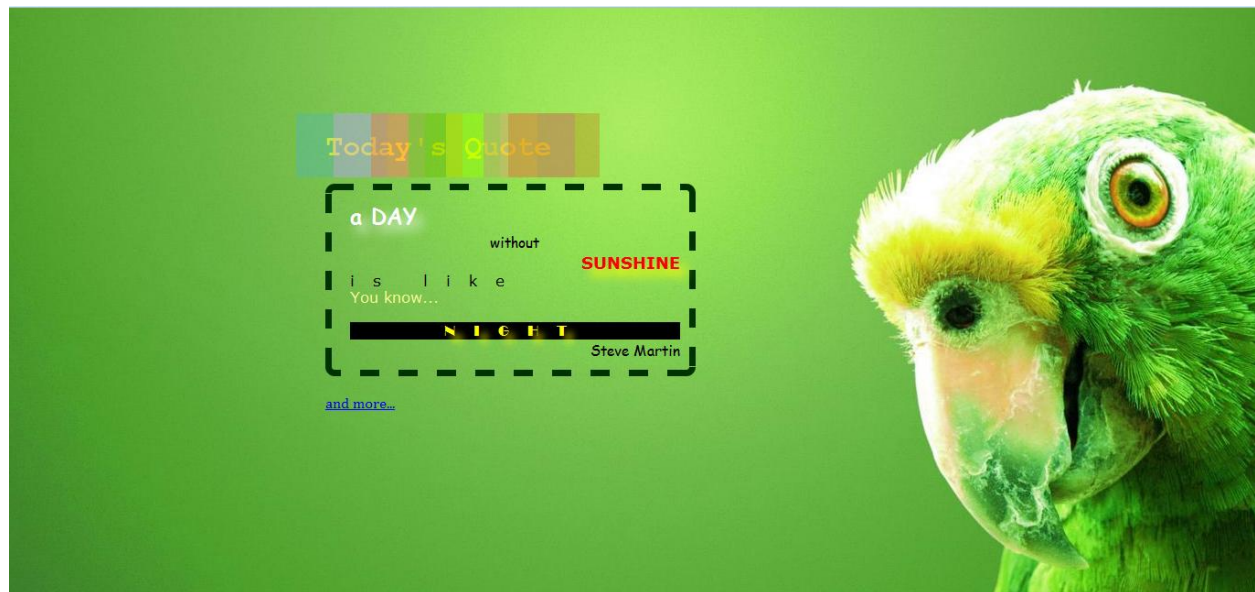
Sinh viên thực hành thay đổi giá trị các thuộc tính để thấy sự thay đổi của trang web.

Bài tập

1. Dùng CSS để định dạng một table như hình sau. Lưu ý, "Sign Up" được định dạng là các buttons; khi trỏ chuột vào Sign Up, cursor sẽ được đổi sang hình bàn tay

Beginner	Advanced	Ultimate
\$ 2 . ⁹⁹ Monthly	\$ 8 . ⁹⁹ Monthly	\$ 10 . ⁹⁹ Monthly
10GB Of Storage	20GB Of Storage	30GB Of Storage
01 Domain	02 Domains	03 Domains
No Daily Backup	No Daily Backup	Weekly Backup
Sign Up	Sign Up	Sign Up

2. Làm lại bài tập 4 của lab 1, sử dụng CSS:
 - Trang web đã sử dụng các tags: <div>, <h1>, <a>
 - Link background: <http://downloads.xdesktopwallpapers.com/wp-content/uploads/2012/07/Green%20Parrot%20On%20Green%20Background.jpg>
 - Link "and more...": http://www.brainyquote.com/quotes/topics/topic_funny.html
 - Các thuộc tính CSS tham khảo: position, opacity, border-style, text-shadow, letter-spacing
 - Các font-family tham khảo: verdana, cursive, Comic Sans MS, Broadway



3. Tạo một album ảnh như hình sau (sinh viên có thể dùng file **img.rar** hoặc chèn hình tùy ý)

Your Gallery



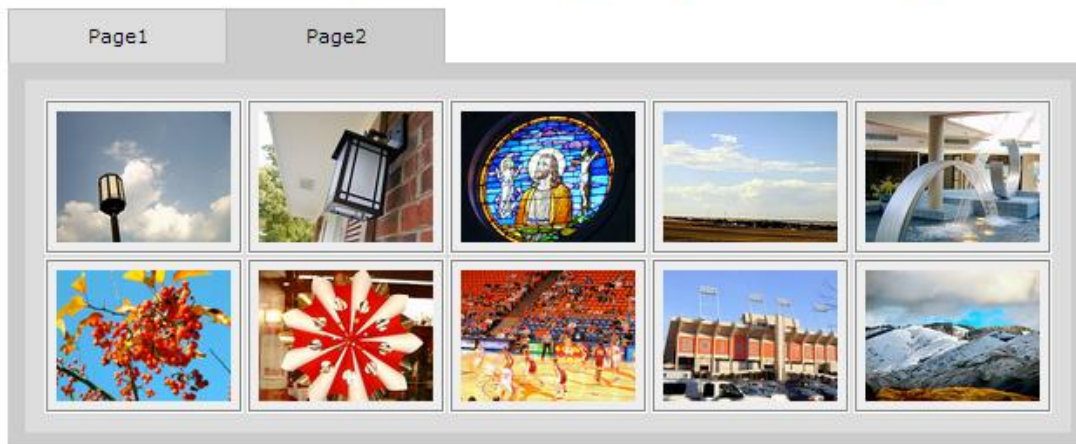
Khi trỏ chuột vào một ảnh, ta có kết quả sau



Gợi ý:

- Bạn có thể tạo một list rồi đặt các images trong từng phần tử của list
 - Định dạng list với list-style:none để các images được nằm ngang
 - Sử dụng trạng thái hover
 - Ngoài ra, bạn có thể chọn cách dùng table. Tuy nhiên, cách này thiếu sự linh hoạt khi số lượng image thay đổi
4. Tương tự như bài 3, tạo một album có 2 trang như hình sau. Lưu ý, trang 2 đang được activated

Multi Page Photograph Gallery

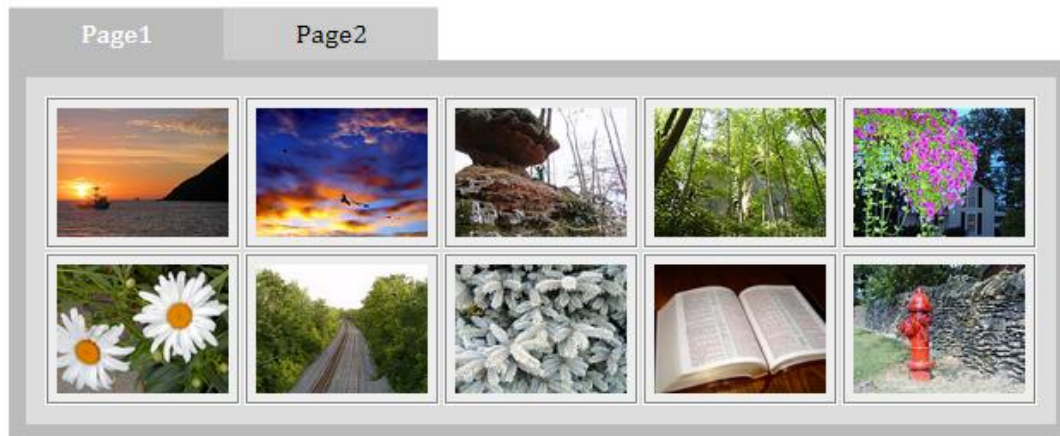


Khi trỏ chuột vào một ảnh



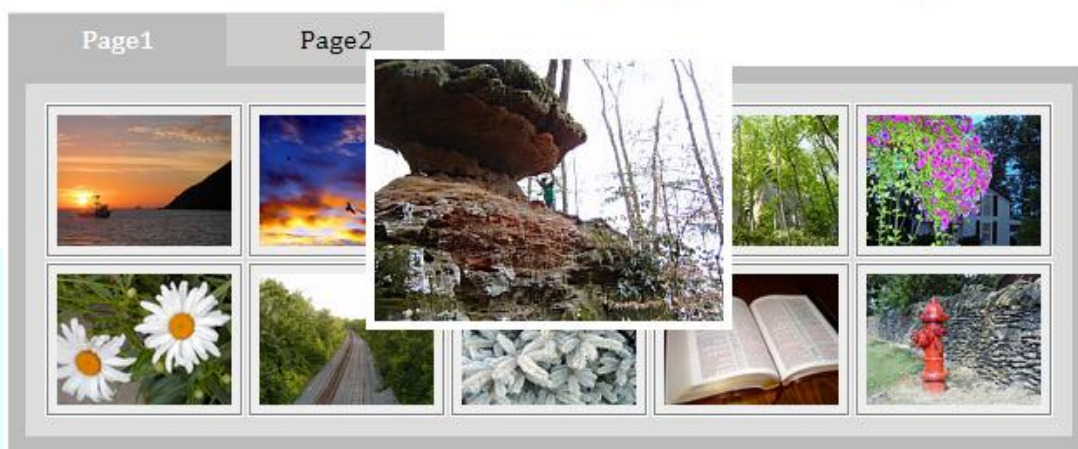
Khi trỏ chuột vào tab Page1

Multi Page Photograph Gallery



Tiếp đó, di chuyển chuột đến một image trong tab Page1

Multi Page Photograph Gallery



Gợi ý:

- Phần image làm tương tự bài 3
 - Để tạo các tab, bạn có thể thao khảo cách sau:
 - o Tạo một list gồm 2 phần tử tương ứng với 2 tab
 - o Mỗi phần tử của list trên sẽ chứa một list khác, list con này bao gồm tab và khung chứa hình ảnh trong một tab (tương tự bài 3)
 - o Định dạng khung chứa hình ảnh của tab 2 cho đúng vị trí canh trái ngoài cùng
 - o Khi trỏ chuột vào tab 1, hiển thị khung chứa hình ảnh của tab này ở cùng vị trí khung của tab 2, đồng thời thay đổi màu nền và màu khung tương ứng
5. Sử dụng file **menu.rar**, tạo một menu như hình sau



Khi trỏ chuột vào Home



Khi trỏ chuột vào About



Khi trỏ chuột vào Rss



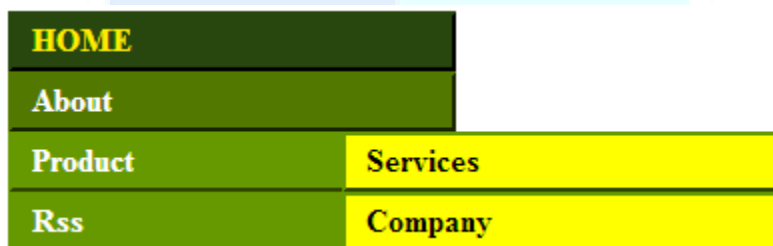
Gợi ý:

- Tạo menu từ 1 list các phần tử Home, About và RSS
- Đặt hình nền cho list này là hình nền của menu

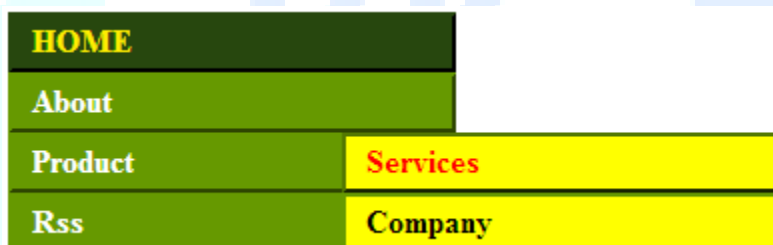
- Đặt hình nền cho các link tương ứng với vị trí của Home, About, RSS
 - Khi trỏ chuột vào link, hiển thị các hình ảnh kèm theo, ví dụ khi trỏ chuột vào Home, hiển thị hình home-over.gif và phần dưới của hình home.gif (bằng cách set background-position)
6. Tạo một menu đứng như hình vẽ. Tab HOME đang được activated



Khi trỏ chuột vào About



Khi trỏ chuột vào Services hoặc Company



Khi click vào 1 tab cha bất kỳ (About, Product, Rss), ví dụ Product



Tuy nhiên, tab được activated sẽ giữ cùng một định dạng khi hover, focus hay click

7. Chỉ dùng CSS và HTML, tạo trang web như sau. Sinh viên có thể sử dụng file ảnh **cat.rar**

Catch me if you can!



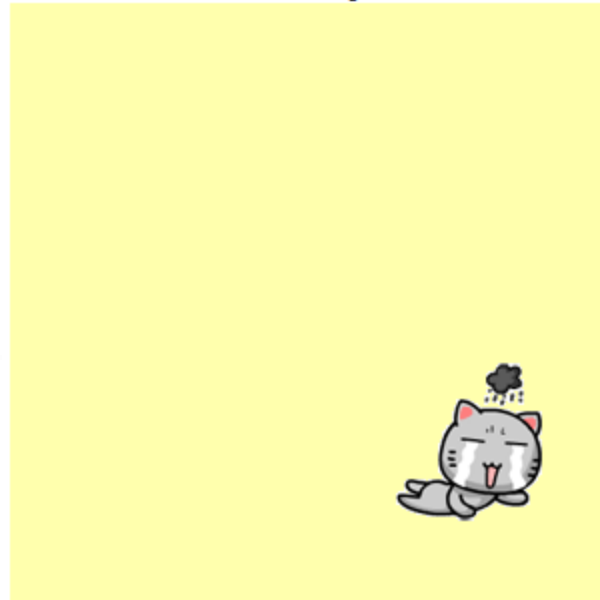
Khi trỏ chuột vào image

Catch me if you can!



Khi click và giữ chuột

Catch me if you can!



Tài liệu tham khảo

- [1] W3Schools CSS, http://www.w3schools.com/css/css_syntax.asp (2013)
- [2] 20 CSS examples, <http://develop-a-website.com/20-amazing-css-examples> (2013)
- [3] 30 CSS examples, http://sixrevisions.com/css/30_css_techniques_examples/ (2013)
- [4] CSS hoverbox, <http://host.sonspring.com/hoverbox/> (2013)
- [5] CSS menu, <http://webdesignerwall.com/tutorials/advanced-css-menu> (2013)
- [6] CSS dropdown menu (2013)
<http://line25.com/tutorials/how-to-create-a-pure-css-dropdown-menu>