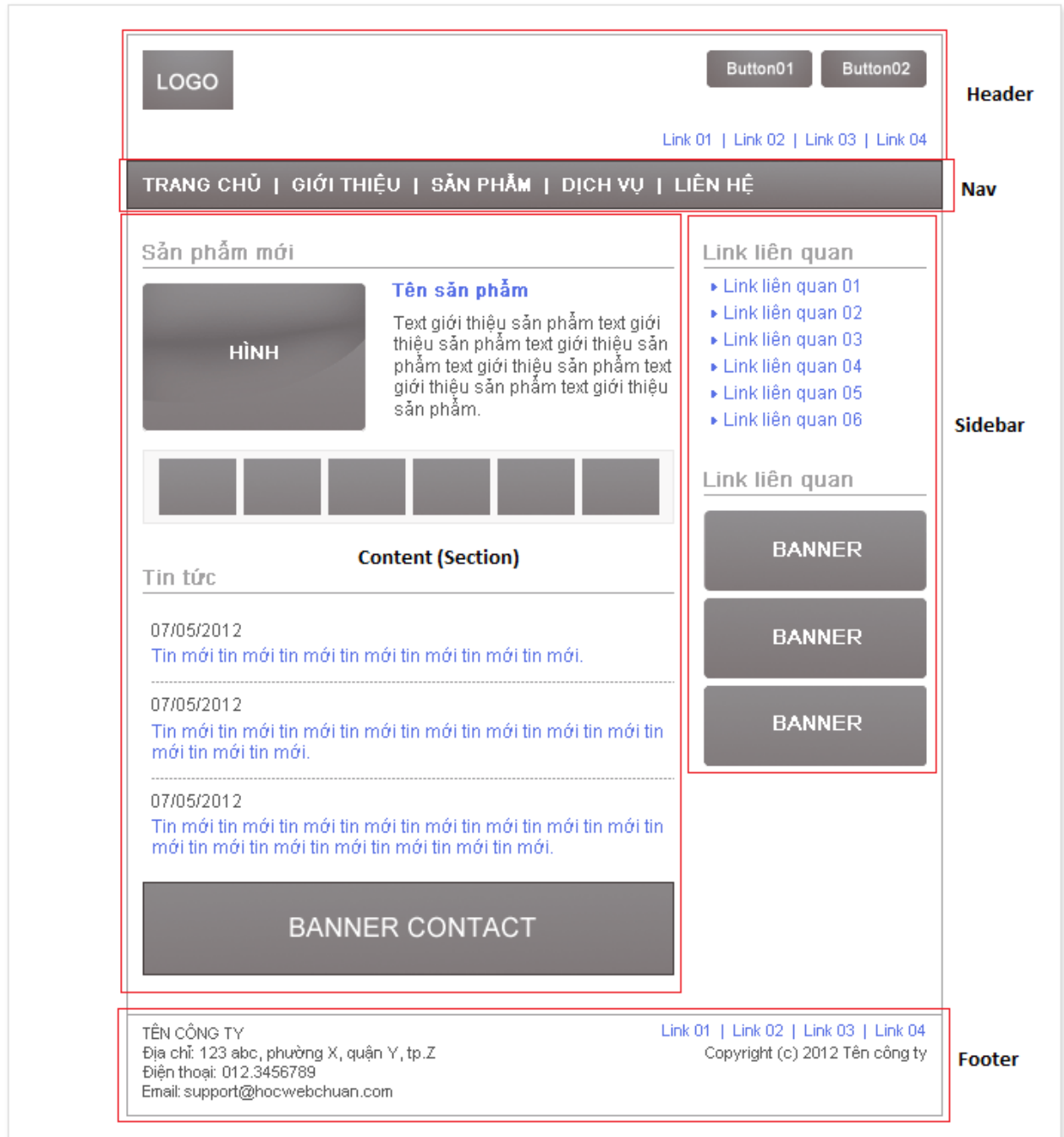


Hướng dẫn bài tập

I. Phân tích giao diện

Dựa vào bản thiết kế, có thể phân tích bố cục trang web thành các phần như sau:



Đối chiếu với mã nguồn HTML, Các thẻ tương ứng với các thành phần bố cục như sau:

- Header: Thẻ `<div id="header">`
- Navigation bar: Thẻ `<div id="gNav">`
- Content: Thẻ `<div id="content">`
- Sidebar: Thẻ `<div id="sidebar">`
- Footer: Thẻ `<div id="footer">`

Khi thiết kế bố cục trang web, thông thường chúng ta sẽ đi theo quy tắc từ trên xuống dưới (Header → Navigation bar → Content → Sidebar → Footer) và quy tắc từ ngoài vào trong (tức là tạo bố cục tổng quát trước rồi mới đi đến bố cục chi tiết bên trong).

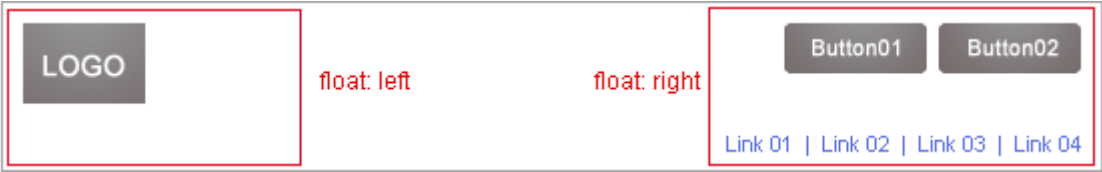
Trước khi bắt đầu viết CSS, cần phải phân tích bố cục cho phần đang định dạng và định hình trước những thuộc tính cần sử dụng.

II. Header

Phần Header của trang web như sau:



Phần Header gồm 1 logo nằm bên trái, 2 button và 4 link nằm bên phải. Tương ứng với 2 phần này là thẻ `<div class="hLeft">` và thẻ `<div class="hRight">`. Để đưa 2 phần này trượt về 2 phía như vậy, chúng ta dùng thuộc tính float.

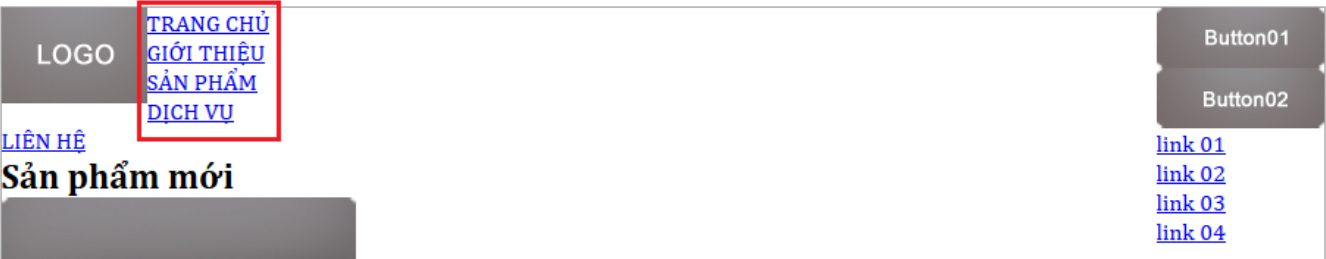


CSS viết như sau:

```
div.hLeft {
    float: left;
}
```

```
div.hRight {
    float: right;
}
```

Tuy nhiên, khi sử dụng float như vậy, khoảng không gian trống sẽ đủ chỗ để thành phần Navigation bar phía dưới chiếm chỗ:

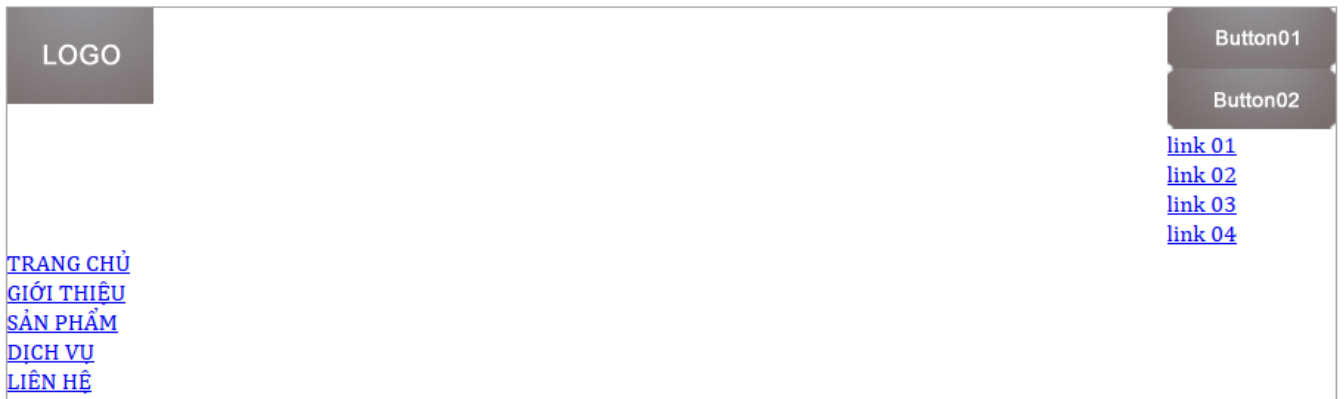


Do đó, chúng ta cần dùng thuộc tính clear cho Navigation bar (div#gNav) để xóa thuộc tính float. Do 2 phần phía trên float cả left và right nên cần phải clear: both.

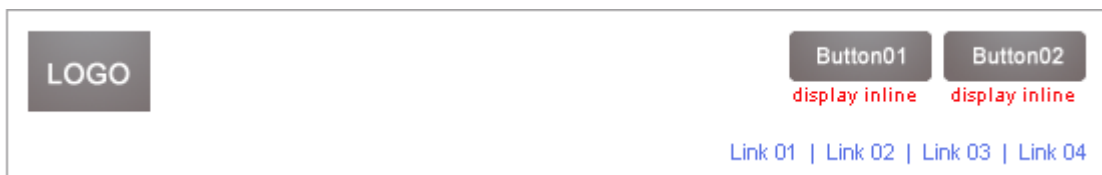
CSS viết như sau:

```
div#gNav {
    clear: both;
}
```

Kết quả:



Phân tích nhóm Button01 và Button02: Mặc dù 2 button này được chèn dưới dạng hình ảnh bằng thẻ là thẻ inline, nhưng lại nằm trong thẻ là thẻ block, do đó chúng hiển thị trên 2 dòng khác nhau. Để chúng hiển thị trên cùng 1 dòng như 1 đối tượng inline, ta chỉnh thuộc tính display: inline cho các thẻ này. Tương tự cho các thẻ của nhóm 4 link phía dưới:



CSS viết như sau:

```
div.hRight ul.hButton li {  
    display: inline;  
}
```

```
div.hRight ul.hNav li {  
    display: inline;  
}
```

Kết quả:



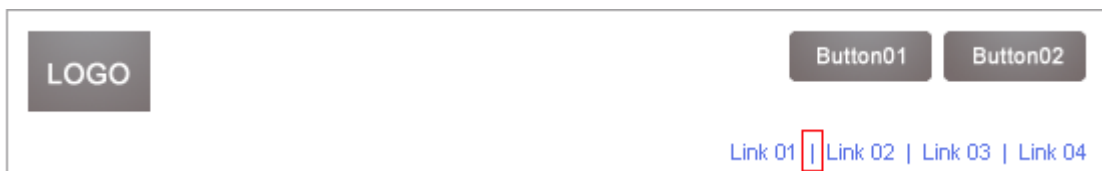
Tuy nhiên, nhóm button và link được canh sát lề phải của trang web, do đó ta dùng thuộc tính text-align cho 2 thẻ chứa 2 danh sách button và link này.

CSS viết như sau:

```
div.hRight ul.hButton {  
    text-align: right;  
}
```

```
div.hRight ul.hNav {  
    text-align: right;  
}
```

Phân tích danh sách 4 link trong Header:



Các đường thẳng nằm giữa các link có thể chèn bằng text vào file HTML, nhưng cách này không khuyến khích vì cần can thiệp vào mã nguồn HTML. Chúng ta có thể sử dụng thuộc tính border-right cho các link này, đồng thời dùng padding để phần nội dung nằm cách viền 1 khoảng.

CSS viết như sau:

```
div.hRight ul.hNav li {  
    .....  
    border-right: 1px solid #333;  
    padding: 0 5px;  
}
```

Ta thấy cuối danh sách link có 1 border thừa:



Do đó, đối với thẻ cuối cùng (li.last), ta sẽ hủy bỏ thuộc tính border-right cũng như padding-right.

CSS viết như sau:

```
div.hRight ul.hNav li.last {  
    border-right: none;  
    padding-right: 0;  
}
```

Kết quả:



Cuối cùng, chúng ta sử dụng các thuộc tính margin và padding để giãn khoảng cách cho giống với bản thiết kế:

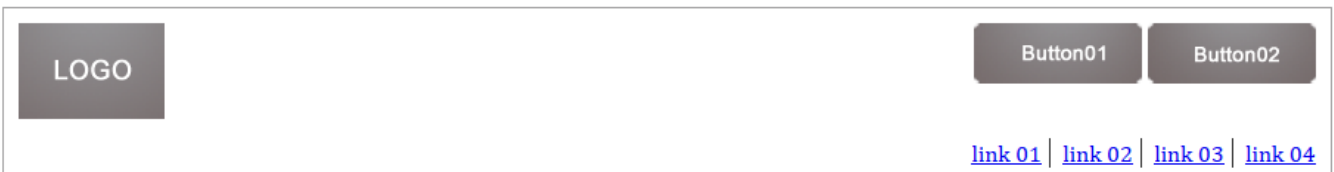
- Header cách viền trang web 1 khoảng: sử dụng thuộc tính padding cho Header.
- Nhóm button giãn cách với nhóm link 1 khoảng: sử dụng thuộc tính margin-bottom cho thẻ chứa nhóm button này (tức là thẻ ul.hButton).
- Nhóm link giãn cách với Navigation bar 1 khoảng: sử dụng thuộc tính margin-bottom cho thẻ chứa nhóm link này (tức là thẻ ul.hNav).

CSS viết như sau:

```
div#header {  
    padding: 10px;  
}  
div.hRight ul.hButton {  
    margin-bottom: 37px;  
    .....  
}
```

```
div.hRight ul.hNav {
    margin-bottom: 10px;
    .....
}
```

Kết quả cuối cùng:



Toàn bộ CSS của phần Header viết như sau:

```
div#header {
    padding: 10px;
}
div.hLeft {
    float: left;
}
div.hRight {
    float: right;
}
div.hRight ul.hButton {
    margin-bottom: 37px;
    text-align: right;
}
div.hRight ul.hButton li {
    display: inline;
}
div.hRight ul.hNav {
    margin-bottom: 10px;
    text-align: right;
}
div.hRight ul.hNav li {
    display: inline;
    border-right: 1px solid #333;
    padding: 0 5px;
}
div.hRight ul.hNav li.last {
    border-right: none;
    padding-right: 0;
}
div#gNav {
    clear: both;
}
```

III. Navigation bar

Phần Navigation bar của trang web như sau:



Dễ thấy phần menu được tạo bằng danh sách và được định dạng tương tự như phần link trong Header (hiển thị inline, có đường phân cách màu trắng giữa các link).

CSS viết như sau:

```
div#gNav ul li {
    border-right: 1px solid white;
    display: inline;
    padding: 0 10px;
}

div#gNav ul li.last {
    border-right: none;
    padding-right: 0;
}
```

Ngoài ra, Navigation Bar còn có ảnh nền là file ảnh *img/bg_gnav.gif* được lặp theo chiều ngang và có viền màu đen.

CSS viết như sau:

```
div#gNav ul {
    background: url(../img/bg_gnav.gif) repeat-x;
    border: 1px solid black;
}
```

Kết quả:



Tiếp theo, cần chỉnh lại chiều cao của nhóm danh sách (thẻ) là 22px bằng thuộc tính height, đồng thời kết hợp với thuộc tính padding để canh cho phần danh sách nằm giữa Navigation bar theo chiều dọc. Do ảnh nền của Navigation bar có chiều cao 46px mà chiều cao của thẻ đã là 22px, còn lại 24px chia đều cho padding trên và dưới của danh sách này, vậy mỗi padding trên và dưới là 12px.

CSS viết như sau:

```
div#gNav ul {
    .....
    height: 22px;
    padding: 12px 0;
}
```

Kết quả:



Cuối cùng, ta chỉnh lại các link (các thẻ <a> của Navigation bar) có chữ màu trắng, đậm, không có gạch chân và có hiệu ứng đổi màu chữ thành vàng khi rê chuột lên trên link.

CSS viết như sau:

```
div#gNav ul li a {
    color: white;
    font-weight: bold;
    text-decoration: none;
}
```

```
div#gNav ul li a:hover {
    color: yellow;
}
```

Kết quả cuối cùng:

TRANG CHỦ | GIỚI THIỆU | SẢN PHẨM | DỊCH VỤ | LIÊN HỆ

Toàn bộ CSS của phần Navigation bar viết như sau:

```
div#gNav ul {
    background: url(../img/bg_gnav.gif) repeat-x;
    border: 1px solid black;
    height: 22px;
    padding: 12px 0;
}
div#gNav ul li {
    border-right: 1px solid white;
    display: inline;
    padding: 0 10px;
}
div#gNav ul li.last {
    border-right: none;
    padding-right: 0;
}
div#gNav ul li a {
    color: white;
    font-weight: bold;
    text-decoration: none;
}
div#gNav ul li a:hover {
    color: yellow;
}
```

IV. Content



Phần Content nằm bên trái và Sidebar nằm bên phải của trang, do đó ta dùng thuộc tính float để định dạng cho 2 phần này (tương tự như ở Header), kết hợp với thuộc tính width để 2 phần này có vùng hiển thị đủ chỗ cho nhau.

Chiều rộng trang web là 900px, do đó ta cài đặt width của Content và Sidebar sao cho tổng của chúng không vượt quá 900px, và tốt nhất là tổng này nhỏ hơn 900px để 2 phần này cách nhau 1 khoảng ở giữa.

Tương tự như ở Header, cần phải cài đặt thuộc tính clear cho Footer.

CSS viết như sau:

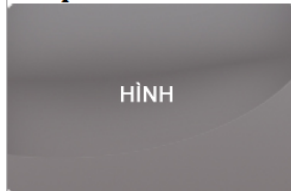
```
div#content {  
    float: left;  
    width: 620px;  
}
```

```
div#sidebar {  
    float: right;  
    width: 240px;  
}
```

```
div#footer {  
    clear: both;  
}
```

Kết quả:

Sản phẩm mới



Tên sản phẩm

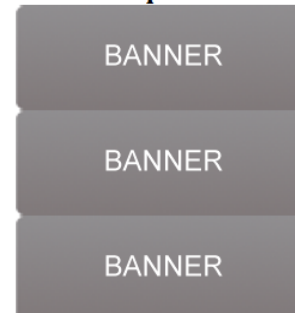
Text giới thiệu sản phẩm text giới thiệu sản phẩm text giới thiệu sản phẩm text giới thiệu sản phẩm text giới thiệu sản phẩm.



Link liên quan

[Link liên quan 01](#)
[Link liên quan 02](#)
[Link liên quan 03](#)
[Link liên quan 04](#)
[Link liên quan 05](#)
[Link liên quan 06](#)

Link liên quan



Phân tích: Phía dưới tiêu đề "Sản phẩm mới" có 1 đường gạch ngang, chúng ta sẽ sử dụng thuộc tính border-bottom cho tiêu đề này. Lí do là nếu định dạng gạch chân với thuộc tính text-decoration: underline thì đường gạch chân này chỉ kéo dài đến hết dòng chữ mà không kéo dài đến hết chiều ngang của phần Content được:



CSS viết như sau:

```
div#content h2 {  
    border-bottom: 1px solid #666666;  
    font-size: 120%;  
}
```

Phân tích: Nhóm sản phẩm mới có hình ảnh nằm bên trái và nội dung nằm bên phải, do đó ta tiếp tục sử dụng thuộc tính float và width tương tự như ở trên (lưu ý rằng tổng width của 2 phần này không vượt quá width của phần Content là 620px), kèm theo thuộc tính clear cho nhóm hình ảnh bên dưới. Ngoài ra, dùng thuộc tính color để đổi màu cho "Tên sản phẩm".

CSS viết như sau:

```
div#content div.newProduct p.image {  
    float: left;  
    width: 241px;  
}  
  
div#content div.newProduct div {  
    float: right;  
    width: 350px;  
}  
  
div#content div.newProduct div h3 {  
    color: #5870e6;  
}  
  
div#content ul.thumbList {  
    clear: both;  
}
```

Vấn đề còn lại của nhóm sản phẩm mới là sử dụng padding hoặc margin phù hợp để điều chỉnh các khoảng không gian cho giống với bản thiết kế.

CSS viết như sau:

```
div#pageBody {  
    padding: 20px 10px;  
}  
  
div#content h2 {  
    .....  
    margin-bottom: 10px;  
}
```

```
div#content div.newProduct div h3 {
    .....
    margin-bottom: 10px;
}
```

Kết quả:



Phân tích: Nhóm 6 hình ảnh ở dưới nhóm sản phẩm mới được tạo bằng danh sách . Do đó, để chuyển danh sách hình này thành hàng ngang, ta có thể sử dụng thuộc tính display: inline cho các thẻ kèm theo thuộc tính padding-right để giãn khoảng cách giữa các hình.

Ngoài ra, danh sách này còn có màu nền, đường viền và 1 khoảng padding giữa danh sách hình và phần viền này:



CSS viết như sau:

```
div#content ul.thumbList {
    background: #f9f8f8;
    border: 1px solid #ebeaea;
    padding: 10px;
    .....
}

div#content ul.thumbList li {
    display: inline;
```

```
padding-right: 8px;
```

```
}
```

Kết quả:



Hiện giờ, chúng ta không thể chỉnh khoảng cách giữa nhóm sản phẩm mới và nhóm danh sách hình phía dưới (bằng padding hay margin) vì nhóm sản phẩm mới chúng ta đã cài đặt float, vùng không gian của nó đã không còn, do đó mốt tính padding hay margin của danh sách hình được tính từ biên trên của hình sản phẩm mới.

Để giải quyết vấn đề này, chúng ta sử dụng 1 thủ thuật **clearfix**: Tạo 1 vùng nội dung rỗng, ẩn, và không có kích thước ngay sau nhóm sản phẩm mới. Khi đó, mốt tính padding hay margin cho nhóm danh sách hình sẽ được tính từ vùng này.

CSS viết như sau:

```
div#content div.newProduct:after {  
    clear: both;  
    content: ".";  
    display: block;  
    height: 0;  
    line-height: 0;  
    visibility: hidden;  
}
```

Sau khi sử dụng thủ thuật trên, chúng ta có thể dùng thuộc tính padding hay margin để điều chỉnh khoảng cách giữa nhóm sản phẩm mới và nhóm danh sách hình dễ dàng.

CSS viết như sau:

```
div#content div.newProduct {  
    margin-bottom: 20px;  
}
```

Phân tích: Công việc còn lại của phần Content gồm có:

- Loại bỏ gạch dưới cho các liên kết. Tuy nhiên khi đưa chuột lên tin tức sẽ xuất hiện lại gạch dưới.
- Thêm border-bottom dạng gạch dưới và padding-bottom cho các mục trong phần tin tức và loại bỏ 2 thuộc tính này cho tin tức cuối cùng.
- Canh giữa cho banner contact, đồng thời sử dụng margin-bottom để giãn khoảng cách giữa banner này với phần Footer.

- Điều chỉnh khoảng cách cho giống với bản thiết kế

Tin tức

07/05/2012

Tin mới tin mới tin mới tin mới tin mới tin mới tin mới. border

07/05/2012

Tin mới tin mới tin mới tin mới tin mới tin mới tin mới tin mới tin
mới tin mới tin mới.

07/05/2012

Tin mới tin mới tin mới tin mới tin mới tin mới tin mới tin mới tin
mới tin mới tin mới tin mới tin mới tin mới tin mới.

CSS viết như sau:

```
div#content dl.news { /* Định dạng lại khoảng cách cho phần tin tức */
    padding: 10px 0 40px 10px;
}

div#content dl.news dd { /* Tạo viền đứt nét bên dưới các tin tức và
padding, margin */
    border-bottom: 1px dotted #000;
    margin-bottom: 10px;
    padding-bottom: 5px;
}

div#content dl.news dd.last { /* Xóa viền và padding, margin cho tin tức
cuối cùng */
    border-bottom: none;
    margin-bottom: 0;
    padding-bottom: 0;
}

div#content dl.news dd a { /* Bỏ gạch dưới cho các liên kết */
    text-decoration: none;
}

div#content dl.news dd a:hover { /* Khi hover sẽ xuất hiện gạch dưới */
    text-decoration: underline;
}

div#content p.banner { /* Canh giữa cho banner contact và giãn cách với
Footer */
    text-align: center;
    margin-bottom: 40px;
}
```

V. Sidebar

Phân tích: 2 tiêu đề <h2> trong phần Sidebar cũng tương tự như trong phần Content (sử dụng border-bottom để làm đường viền phía dưới tiêu đề).

Link liên quan

- ▶ Link liên quan 01
- ▶ Link liên quan 02
- ▶ Link liên quan 03
- ▶ Link liên quan 04
- ▶ Link liên quan 05
- ▶ Link liên quan 06

Link liên quan

BANNER

BANNER

BANNER

CSS viết như sau:

```
div#sidebar h2 {  
    border-bottom: 1px solid #666666;  
    font-size: 120%;  
    margin-bottom: 10px;  
}
```

Phân tích: Danh sách link liên quan tạo bằng thẻ , nhưng ngay từ đầu file CSS chúng ta đã cài đặt list-style của các thẻ và là none, tức là danh sách không thể có biểu tượng (bullet) trừ khi cài đặt lại thuộc tính này. Thay vào đó, chúng ta sử dụng ảnh img/ico_arrow.gif để làm ảnh nền cho các mục trong danh sách này, và để phần text không đè lên ảnh nền, ta cài đặt luôn padding cho chúng:

- icon arrow
- ### Link liên quan
- ▶ Link liên quan 01
 - ▶ Link liên quan 02
 - ▶ Link liên quan 03
 - ▶ Link liên quan 04
 - ▶ Link liên quan 05
 - ▶ Link liên quan 06

CSS viết như sau:

```
div#sidebar ul.linkList li {  
    background: url(../img/ico_arrow.gif) no-repeat;  
    padding-left: 15px;  
}
```

Tuy nhiên, khi đó những biểu tượng này không nằm giữa text. Muốn sửa, ta cài đặt thuộc tính background-position là left center:

- ▶ Link liên quan 01
- ▶ Link liên quan 02
- ▶ Link liên quan 03
- ▶ Link liên quan 04
- ▶ Link liên quan 05
- ▶ Link liên quan 06

CSS viết như sau:

```
div#sidebar ul.linkList li {  
    background: url(../img/ico_arrow.gif) no-repeat left center;  
    padding-left: 15px;  
}
```

Kết quả:

- ▶ [Link liên quan 01](#)
- ▶ [Link liên quan 02](#)
- ▶ [Link liên quan 03](#)
- ▶ [Link liên quan 04](#)
- ▶ [Link liên quan 05](#)
- ▶ [Link liên quan 06](#)

Cuối cùng, chúng ta sử dụng padding và margin để điều chỉnh khoảng cách các thành phần cho giống với bản thiết kế:

- Danh sách link liên quan cách 1 khoảng so với biên trái của Sidebar và so với nhóm banner phía dưới: sử dụng thuộc tính padding cho thẻ ul.linkList.
- Các đường link dẫn cách với nhau 1 khoảng: sử dụng thuộc tính margin-bottom cho các thẻ trong danh sách này.
- Nhóm banner cách tiêu đề 1 khoảng: sử dụng thuộc tính padding-top cho thẻ ul.bannerList.
- Các banner trong nhóm dẫn cách với nhau 1 khoảng: sử dụng thuộc tính margin-bottom cho các thẻ trong danh sách này.

CSS viết như sau:

```
div#sidebar ul.linkList {  
    padding: 0 0 30px 10px;  
}  
  
div#sidebar ul.bannerList {  
    padding-top: 10px;  
}  
  
div#sidebar ul.bannerList li {  
    margin-bottom: 5px;  
}
```

Toàn bộ CSS cho phần Sidebar viết như sau:

```
div#sidebar {  
    float: right;  
    width: 240px;  
}  
  
div#sidebar h2 {  
    border-bottom: 1px solid #666666;  
    font-size: 120%;  
    margin-bottom: 10px;  
}  
  
div#sidebar ul.linkList {  
    padding: 0 0 30px 10px;  
}
```

```
div#sidebar ul.linkList li {
    background: url(../img/ico_arrow.gif) no-repeat left center;
    padding-left: 15px;
    margin-bottom: 3px;
}
div#sidebar ul.bannerList {
    padding-top: 10px;
}
div#sidebar ul.bannerList li {
    margin-bottom: 5px;
}
```

VI. Footer

Phần Footer có cấu trúc chia làm 2 bên, do đó tương tự như những gì đã làm ở phần trên, ta sử dụng thuộc tính float và width để cài đặt bố cục cho Footer (lưu ý rằng width của trang web đang là 900px):

TÊN CÔNG TY Địa chỉ: 123 abc, phường X, quận Y, tp.Z Điện thoại: 012.3456789 Email: support@hocwebchuan.com	Link 01 Link 02 Link 03 Link 04 Copyright (c) 2012 Tên công ty
--	---

CSS viết như sau:

```
div#footer div.fLeft {
    float: left;
    width: 340px;
}

div#footer div.fRight {
    float: right;
    width: 530px;
}
```

Tuy nhiên, do sử dụng float như vậy nên phần border của trang web không còn bao lấy Footer nữa mà chỉ bao đến hết phần Content:



Lí do là khi sử dụng float, vùng không gian của thẻ đó không còn, nên trình duyệt hiểu diện tích trang web kết thúc ở biên dưới của Content.

Để khắc phục tình trạng này, ta sử dụng thủ thuật **clearfix** tương tự như đã làm ở phần Content: tạo 1 vùng nội dung rỗng, ẩn và không có kích thước ở dưới Footer. Khi đó trình duyệt sẽ hiểu là diện tích trang web kết thúc ở vùng này.

CSS viết như sau:

```
div#footer:after {
    clear: both;
    content: ".";
```

```
display: block;
height: 0;
line-height: 0;
visibility: hidden;
```

```
}
```

Kết quả:

TÊN CÔNG TY	Link 01
Địa chỉ: 123 abc, phường X, quận Y, Tp. Z	Link 02
Điện thoại: 0123456789	Link 03
Email: support@web.com	Link 04
Copyright (c) 2012 Tên công ty	

Danh sách link bên phải cài đặt tương tự như danh sách link trong phần Header. Bên cạnh đó, chúng ta cũng sử dụng thuộc tính text-align để canh lề bên phải cho phần Copyright.

CSS viết như sau:

```
div#footer div.fRight ul {
    text-align: right;
}

div#footer div.fRight ul li {
    display: inline;
    border-right: 1px solid #333;
    padding: 0 5px;
}

div#footer div.fRight ul li.last {
    border-right: none;
    padding-right: 0;
}

div#footer div.fRight p {
    text-align: right;
}
```

Kết quả:

TÊN CÔNG TY	Link 01 Link 02 Link 03 Link 04
Địa chỉ: 123 abc, phường X, quận Y, Tp. Z	Copyright (c) 2012 Tên công ty
Điện thoại: 0123456789	
Email: support@web.com	

Phân tích: Đường kẻ ngang phân cách giữa Footer và phần Content-Sidebar có thể cài đặt bằng thuộc tính border-top của Footer:

		<u>border footer</u>
TÊN CÔNG TY	Link 01 Link 02 Link 03 Link 04	
Địa chỉ: 123 abc, phường X, quận Y, tp.Z	Copyright (c) 2012 Tên công ty	
Điện thoại: 012.3456789		
Email: support@hocwebchuan.com		

CSS viết như sau:

```
div#footer {
    .....
    border-top: 1px solid #a3a2a3;
}
```


Công việc còn lại là sử dụng padding và margin để chỉnh khoảng cách cho giống với thiết kế:

- Footer cách phần viền ngoài 1 khoảng: sử dụng padding cho Footer.
- Phần danh sách link bên phải cách viền trên của Footer 1 khoảng: sử dụng margin-top cho thẻ div.fRight.
- Phần danh sách link cách phần copyright 1 khoảng: sử dụng margin-bottom cho danh sách này.

CSS viết như sau:

```
div#footer {
    .....
    padding: 10px;
}

div#footer div.fRight {
    .....
    margin-top: 40px;
}

div#footer div.fRight ul {
    .....
    margin-bottom: 10px;
}
```

Cuối cùng, để giống với thiết kế, ta bỏ hoàn toàn các đường gạch chân ở các liên kết bằng cách cài đặt trực tiếp thẻ <a>, đồng thời chỉnh màu xanh cho các liên kết này.

CSS viết như sau:

```
a {
    text-decoration: none;
    color: blue;
}
```

Toàn bộ CSS cho phần Footer viết như sau:

```
div#footer {
    border-top: 1px solid #a3a2a3;
    padding: 10px;
    clear: both;
}

div#footer div.fLeft {
    float: left;
    width: 340px;
}

div#footer div.fRight {
    float: right;
    margin-top: 40px;
    width: 530px;
}

div#footer div.fRight ul {
    margin-bottom: 10px;
    text-align: right;
}
```

```
div#footer div.fRight ul li {
    display: inline;
    border-right: 1px solid #333;
    padding: 0 5px;
}
div#footer div.fRight ul li.last {
    border-right: none;
    padding-right: 0;
}
div#footer div.fRight p {
    text-align: right;
}
div#footer:after {
    clear: both;
    content: ".";
    display: block;
    height: 0;
    line-height: 0;
    visibility: hidden;
}
```