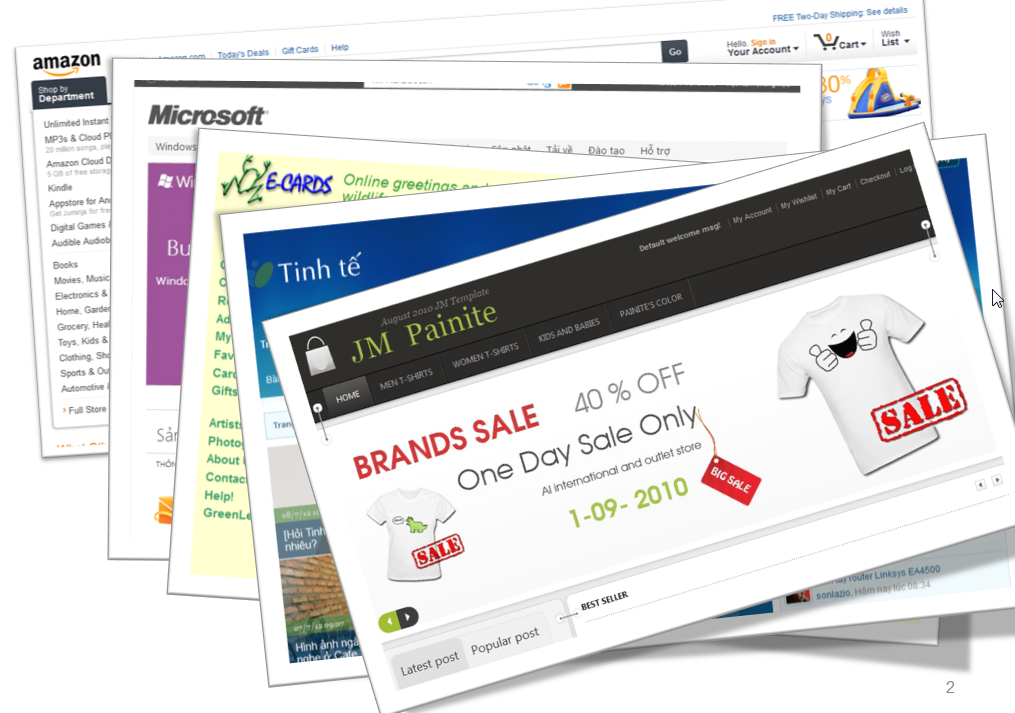
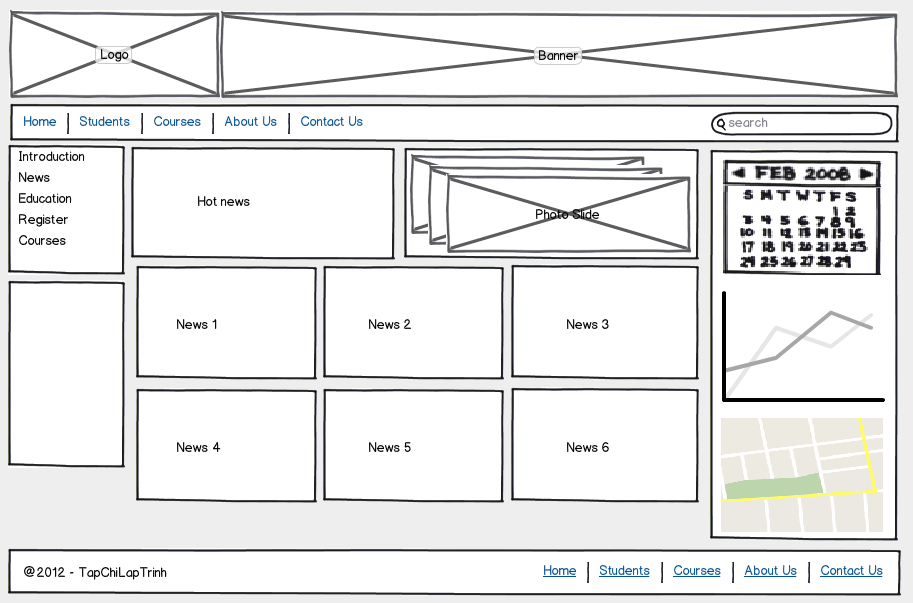
# **Tạo Layout đơn giản với DIV và CSS**



Việc sử dụng DIV kết hợp với CSS để làm layout (bố cục) cho một trang web đã trở lên phổ biến và đạt hiệu quả cao. Chẳng hạn bạn cần thiết kế một trang có giao diện được thiết kế với layout như sau:



Với layout như trên bạn hoàn toàn bạn có thể sử dụng table (bảng) cho việc thiết kế, tuy nhiên table lại bộc lộ khá nhiều nhược điểm khi sử dụng làm layout cho một trang web có cấu trúc như trên. Các nhược điểm có thể gặp phải với table với những dạng layout này là chậm, khó tùy chỉnh và khó kết hợp với CSS Javascript để tạo nên sự linh hoạt cho trang web.

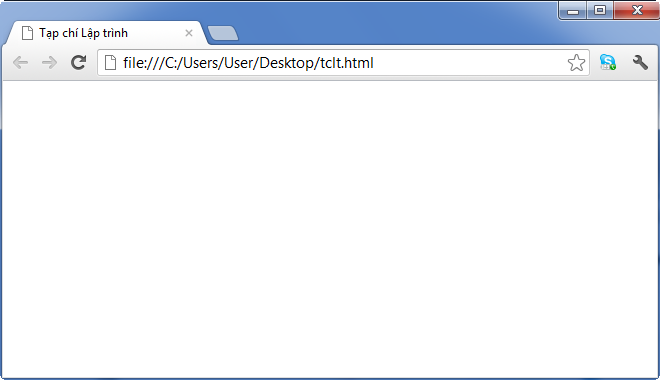
Sau đây tôi sẽ hướng dẫn các bạn các bước căn bản nhất để có thể tạo được các layout với DIV và CSS. Trước hết bạn cần tạo mới một trang HTML với cấu trúc chuẩn (gồm đầy đủ các phần html, head, title, body). Tiếp theo bạn code đoạn mã HTML sau vào phần body của trang.

| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | <div id="main">  <div id="head">  </div>  <div id="head-link">  </div>  <div id="left">  </div>  <div id="content">  </div>  <div id="right">  </div>  <div id="footer">  </div>  </div> |
| --- | --- |

Các cặp thẻ trong đoạn mã trên giúp bạn tạo ra các phân vùng khác nhau trên trang HTML của mình, mỗi phân vùng được đánh ID riêng biệt (các ID này sẽ dùng để định nghĩa CSS), chi tiết như sau:

1. **main**: Phân vùng chứa toàn bộ nội dung của trang.
2. **head:** Phân vùng chứa nội dung đầu tiên của trang (theo VD trên đó là phần logo + banner)
3. **head-link:** Phân vùng chứa các liên kết đầu trang (ở VD trên: Home, Students, Course, v.v.)
4. **left:** Phân vùng chứa các nội dung bên trái (ở VD trên là các liên kết trái)
5. **content:** Phân vùng chứa các nội dung chính của trang (ở VD trên là các phân vùng thông tin như: Hot news, Photo Slide, News 1, v.v.)
6. **right:** Phân vùng chứa các thông tin bên phải trang (ở VD trên là lịch, biểu đồ, bản đồ)
7. **footer:** Phân vùng chứa các thông tin cuối trang (ở VD trên là thông tin về chủ sở hữu và các liên kết)

Chạy thử trang trên trình duyệt kết quả sẽ như sau:



Bây giờ là đến công đoạn code các bộ chọn CSS (selector) để phân vùng rõ ràng cho các DIV đã tạo ở trên:

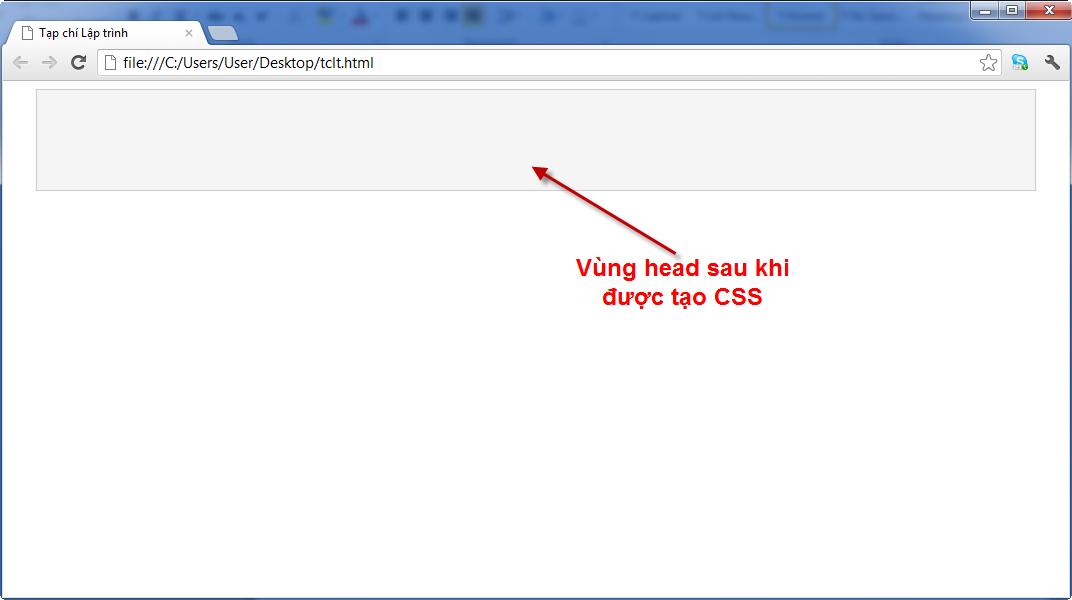
1. Tạo bộ chọn **body** để cố định phông chữ cho cả trang và bộ chọn **#main** để cố định độ rộng vùng thông tin và trang sẽ chiếm và căn chỉnh vùng này nằm giữa màn hình.

| 1  2  3  4  5  6  7  8  9  10  11 | body{  font-family: Arial, Tahoma;  font-size: 12px;  }    #main{  width: 1000px;  padding: 0;  margin-left: auto;  margin-right: auto;  } |
| --- | --- |

2. Tạo bộ chọn **#head** xác định chiều cao, màu nền, đường viền và khoảng cách so với các vùng khác (khoảng cách so với vùng bên dưới – margin-bottom).

| 1  2  3  4  5  6 | #head{  height: 100px;  background-color: #F5F5F5;  border: 1px solid #CDCDCD;  margin-bottom:5px;  } |
| --- | --- |

3. Kiểm tra lại trang trên trình duyệt bạn sẽ có kết quả sau (phần div với id là head đã được xác định):



4. Tiếp tục với định nghĩa bộ chọn **#head-link**, để xác định vùng đặt các liên kế đầu trang:

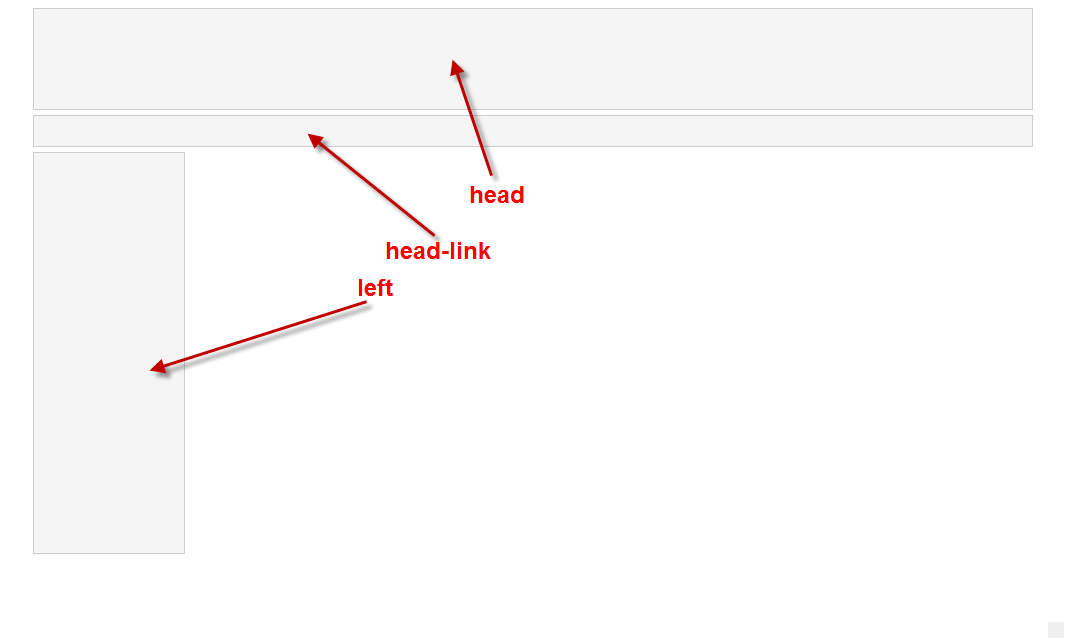
| 1  2  3  4  5  6  7  8  9  10 | #head-link{  height: 30px;  line-height: 30px;  padding-left: 10px;  padding-right: 10px;  border: 1px solid #CDCDCD;  background-color: #F5F5F5;  margin-bottom:5px;  clear: both;  } |
| --- | --- |

5. Quan sát lại trang trên trình duyệt, bạn sẽ thấy vùng liên kết đầu trang hiển thị ngay dưới phần head.

6. Định nghĩa tiếp bộ chọn #left để xác định vùng cho nội dung bên trái, với độ rộng là 200px, chiều cao tối thiểu là 400px và đặc biệt chú ý tới mô tả **float: left;** mô tả này làm cho vùng bên trái (DIV với ID là left) sẽ dạt sang bên trái nhường chỗ cho các phân vùng khác: content và right).

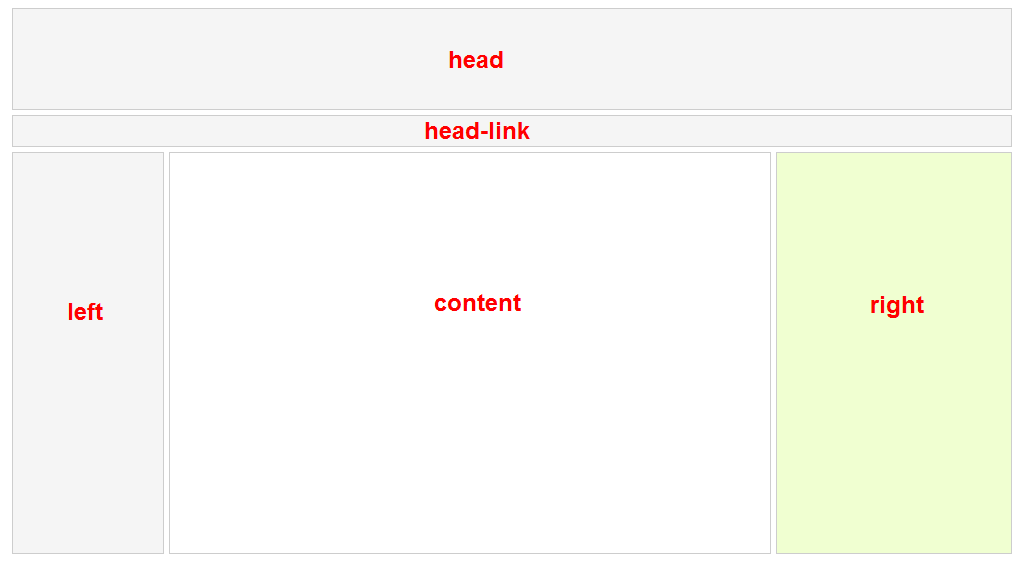
| 1  2  3  4  5  6  7  8 | #left{  width: 150px;  min-height: 400px;  border: 1px solid #CDCDCD;  float:left;  background-color: #004C00;  margin-bottom: 5px;  } |
| --- | --- |

7. Quan sát lại sự thay đổi về các vùng của trang trên trình duyệt:

8. Bổ sung tiếp các selector cho vùng giữa và vùng phải lần lượt là **#content** và **#right**:

| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | #content{  width: 600px;  min-height: 400px;  border: 1px solid #CDCDCD;  float:left;  margin-left: 5px;  margin-right: 5px;  margin-bottom: 5px;  }  #right{  width: 234px;  min-height: 400px;  border: 1px solid #CDCDCD;  float:right;  margin-bottom: 5px;  } |
| --- | --- |

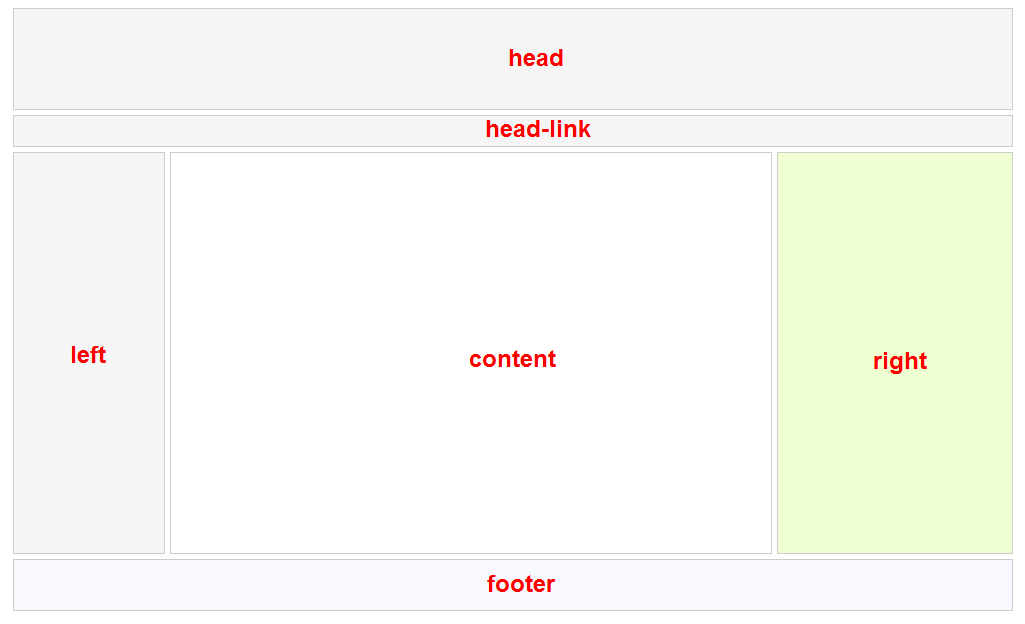
9. Quan sát lại trang



10. Cuối cùng, bạn định nghĩa bộ chọn cho phân vùng cuối trang với id là **#footer**

| 1  2  3  4  5  6 | #footer{  height: 50px;  clear: both;  border: 1px solid #CDCDCD;  background-color: #F8F8FF;  } |
| --- | --- |

11. Quan sát lại trang trên trình duyệt bạn sẽ thấy các phân vùng cần thiết đã vào đúng vị trí cần đặt:



Như vậy qua các bước ở trên chúng ta đã thử nghiệm với các phương pháp cơ bản để dàn trang sử dụng thẻ Div có sự kết hợp với CSS. Điều quan trọng nhất là bạn biết cách áp dụng những kỹ thuật cơ bản này để tạo ra các trang web có layout mà bạn khách hàng của bạn mong muốn. Bây giờ bạn tiếp tục bổ sung thông tin vào các phân vùng và định nghĩa thêm các bộ chọn cần thiết để có thể có một trang HTML giống như mẫu thiết kế ở đầu bài viết.

Chúc bạn thành thạo DIV và CSS để có được những trang web vừa mắt người dùng!