

HTML の基本的な書き方

テキストエディタで「index.html」を作成して、以下のサンプルを入力してみましょう。

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ja" lang="ja">
  <head>
    <meta http-equiv="Content-type" content="text/html; charset=UTF-8" />
    <title>Home</title>
  </head>

  <body>
    <h1>My First Homepage</h1>
    <h2>Home</h2>
    <ul>
      <li>自己紹介</li>
      <li>時間割</li>
      <li>課題ページ</li>
    </ul>
  </body>
</html>
```

最初の 2 行は、HTML を書く前のおまじないの様なものです。
ここで登場する HTML タグについての説明は以下の通りです。

```
<html> </html> : HTML 文書全体
<head> </head> : HTML の設定や情報を書く部分
<body> </body> : HTML の本文
<h1> </h1> : 見出し (数字が大きい程「下位レベル」となる)
<ul> </ul> : リスト
<li> </li> : リスト
<br /> : 改行
```

HTML タグには「<html> </html>」の様にサンドイッチ形式にするものと、「
」の様にサンドイッチにせずにタグ単独で効果を発揮するものがあります。
この他にも HTML タグは沢山あるので、色々試してみましょう。

Hyperlink

HTML ではハイパーリンクを使って、複数のドキュメントを結びつけて参照する事が出来ます。例えば[ココ](#)をクリックすると SFC の HP を参照する事が出来る、といった具合です。ハイパーリンクを書く為の HTML タグは、こちらを使います。

```
<a href="URL"> </a>
```

「URL」の部分で、参照したいドキュメントの URL を、絶対パスや相対パスで指定します。タグの間に文章を記述すると、その文章がハイパーリンクとして表示されます。タグの間に何も記述しないとリンク自体を表示するものが何も無いので、必ず何か目

に見えるものを記述して下さい。また「a href="URL"」の続きに半角スペースを空けて「target="_blank"」と記述すると、リンク先のドキュメントを新しいウィンドウで開く様に設定する事が出来ます。外部リンクなどを記載する場合はこの方法を使うと良いでしょう。

練習 1

上で作成した「index.html」を編集して、以下の 3 つのハイパーリンクを作成して下さい。「自己紹介(例 : me.html)」「時間割(例 : timetable.html)」「課題ページ(例 : homework.html)」。

また、SFC の HP など外部サイトのリンクを新しいウィンドウで表示する様なハイパーリンクを作成して下さい。

画像の表示

HTML では、文章だけではなく画像や写真を表示する事も出来ます。画像は以下の HTML タグを使って表示する事が出来ます。

```

```

ハイパーリンクの時と同様「URL」の部分で、表示したい画像の URL を、絶対パスや相対パスで指定します。この「」は「
」と同じ様に、サンドイッチしないで使うタイプの HTML タグです。また「src="URL"」の続きに半角スペースを空けて「width="__px"」とか「height="__px"」と記述すると、画像の大きさを指定した通りに拡大縮小して、表示する事が出来ます。

練習 2

自分の Web ページの中に画像を表示したり、画像から他のドキュメントにアクセス出来る様なハイパーリンクを作成したりしてみましょう。

文字の装飾

これまではプレーンな文字で文書を書いてきましたが、例えば以下のタグを使って文字に色を付けてみましょう。

```
<font color="#_____"> </font>
```

この様にタグの中に「color="#_____"」と記述すると、文字の色を変更する事が出来ます。「____」の部分は「RGB」の色の値を 2 桁ずつ指定しており、それぞれ「00」から「FF」までの値を取る事が出来ます。「黒 : #000000」「赤 : #FF0000」「緑 : #00FF00」「青 : #0000FF」「白 : #FFFFFF」といった具合です。また「face="FONT"」と(「FONT」にはフ

フォント名を)記述する事でフォントを変更する事が出来ます。使用可能なフォントはブラウザに依存しますが、試しに「Arial」や「Century」などを使ってみましょう。

練習 3

自分の Web ページのフォントを変更してみましょう。上で紹介した 2 つ以外のフォントを探して使ってみて下さっても結構です。

Table の基本

表を作成する時は、テーブルを使うと便利です。

```
<table>
  <tr>
    <td>1-1</td> <td>1-2</td> <td>1-3</td>
  </tr>
  <tr>
    <td>2-1</td> <td>2-2</td> <td>2-3</td>
  </tr>
</table>
```

この様に記述した場合、以下の様な表が出来上がります。

1-1 1-2 1-3

2-1 2-2 2-3

しかしこれでは、枠などが無いので表として少し見難いですね。そこでテーブルでは、以下のパラメータを記述する事で表のスタイルを設定する事が出来ます。

width=" _px " : 幅

height=" _px " : 高さ

border=" _px " : 境界線の太さ

bgcolor=" # ___ " : 背景の色

bordercolor=" # ___ " : 境界線の色

cellspacing=" _px " : セル同士の間隔(「table」タグでのみ有効)

cellpadding=" _px " : セルと文字の間隔(「table」タグでのみ有効)

これらのパラメータを「table」「tr」「td」のどのタグに記述するかによって、設定が適用される場所が変わります。また、テーブル内の文字の記述に関するパラメータもあります。

align=" _ " : 水平方向の整頓(left / center / right)

valign=" _ " : 垂直方向の整頓(top / middle / bottom)

nowrap : 自動改行禁止(「td」タグでのみ有効)

練習 4

これらのパラメータを設定して、より枠組みの見易い表を作ってみましょう。

	a	b	c
1	1-a	1-b	1-c
2	2-a	2-b	2-c
3	3-a	3-b	3-c

こんな感じで表を作れると良いですね。

table の結合

表を作る過程でセル同士を結合させる必要が出て来た時は「colspan」や「rowspan」を使いましょう。「colspan」は横方向の結合、「rowspan」は縦方向の結合を設定します。

```
<table border="1px">
  <tr align="center">
    <td colspan="3">SFC</td>
  </tr>
  <tr>
    <td>総合政策</td>
    <td>環境情報</td>
    <td>看護医療</td>
  </tr>
</table>
```

SFC		
総合政策	環境情報	看護医療

```
</table>
<br />
<table border="1px">
  <tr align="center">
    <td rowspan="3">SFC</td>
    <td>総合政策</td>
  </tr>
  <tr><td>環境情報</td></tr>
  <tr><td>看護医療</td></tr>
</table>
```

	総合政策
SFC	環境情報
	看護医療

練習 5

テーブルを使用して「timetable.html」に、今期の自分の時間割表を作って下さい。ただ単に表を組むだけではなく、積極的にタグを活用してより見易い時間割表になる様に工夫しましょう。

課題提出

今回の課題は、自分のWebページを作る事です。講義中の「練習 1」から「練習 5」までをクリアして、「自己紹介」「時間割」「課題ページ」などのコンテンツを含む Web ページを作りましょう。授業の内容や、より発展的な内容を積極的に取り入れて、是非素敵な Web ページに仕上げてください。課題を以下の URL から参照出来る様にアップロードして、SFC-SFS に URL を提出してください。添付ファイルによる提出は認められませんので、注意しましょう。

課題の URL (例 : <http://web.sfc.keio.ac.jp/~自分のログイン名/mtb/index.html>)

CSS

「CSS」とは「Cascading Style Sheets」の略です。HTML の要素をどの様に視覚表現するのかを指定する為のプレゼンテーション技術です。つまり「ページのデザインを決める為のファイル」という事です。下のサイトを見て、少し CSS を実感してみましょう。

[Zen Garden](#)

「悟りへの道」という始まり方でとても怪しい感じですが、決して怪しいサイトではありません。右側の「デザインを選択」の項目にあるデザインを 1 つ選んでクリックしてみてください。表示されている文章は変わらずに、ページのデザインだけが変更されました。これが「HTML(情報)」と「CSS(デザイン)」を分ける最大のメリットです。長い時間を掛けて HTML を書き直さなくても、CSS を変更するだけでデザイン変更を行うことが出来るのです。

HTML と CSS

それでは実際に CSS を使ってデザインしてみましょう。下の HTML のサンプルと CSS のサンプルを作って、ブラウザから HTML ファイルを確認してみましょう。

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ja" lang="ja">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <link href="sample.css" rel="stylesheet" type="text/css" />
    <title>Sample</title>
  </head>

  <body>
    <div id="container">
      <div id="menu">
        <h1>Menu</h1>
        <ul>
          <li>トップページ</li>
          <li>自己紹介</li>
          <li>時間割</li>
        </ul>
      </div>
      <div id="main">
        <h1>About SFC</h1>
        <p>
          SFCには大きな池があります。その池は鴨池と呼ばれています。<br />
          芝生は綺麗な緑色で、春になると桜が咲くととても素敵な場所です。<br />
          天気の良い日はとても気持ちが良いので、学生からも永く愛されています。<br />
          Link to <a href="http://www.sfc.keio.ac.jp/" target="_blank">SFC</a>
        </p>
      </div>
    </div>
  </body>
</html>
```

```
@charset "UTF-8";
/* CSS Document */

a:link{
    color: #FF6600;
    text-decoration: none;
}

a:visited{
    color: #FF6600;
    text-decoration: none;
}

a:hover{
    color: #0000CC;
    text-decoration: none;
}

a:active{
    color: #FF00CC;
    text-decoration: none;
}

body{
    background-color:#FFFFFF;
    width: auto;
    height: auto;
}

table{
}

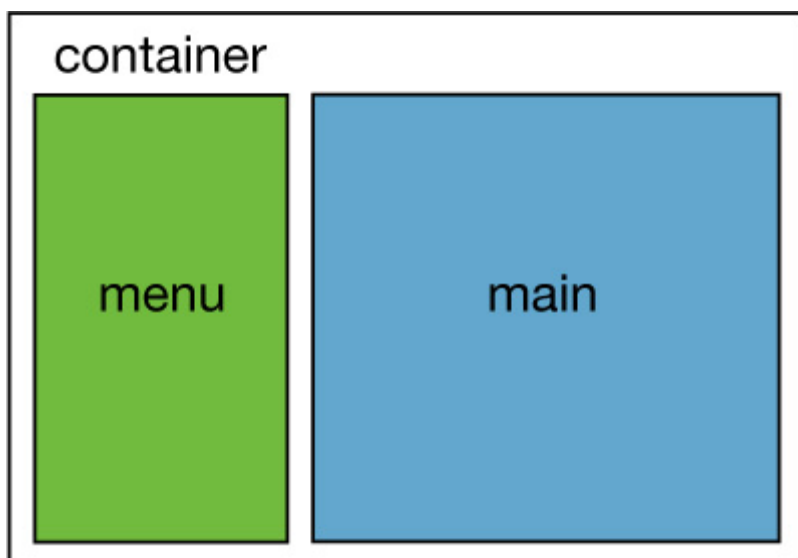
td{
}

tr{
}

#main{
    background-color: #87A5FF;
    width: 80%;
    height: 300px;
    float: right;
}

#menu{
    background-color: #CFFFA4;
    width: 20%;
    height: 300px;
    float: left;
}
```

CSS では、「<div> </div>」というタグを使ってレイアウトを組みます。<div>タグは見出しや文章といった個々のパーツをグループ化する為の box です。この sample.html では以下の図の様な入れ子構造になっています。



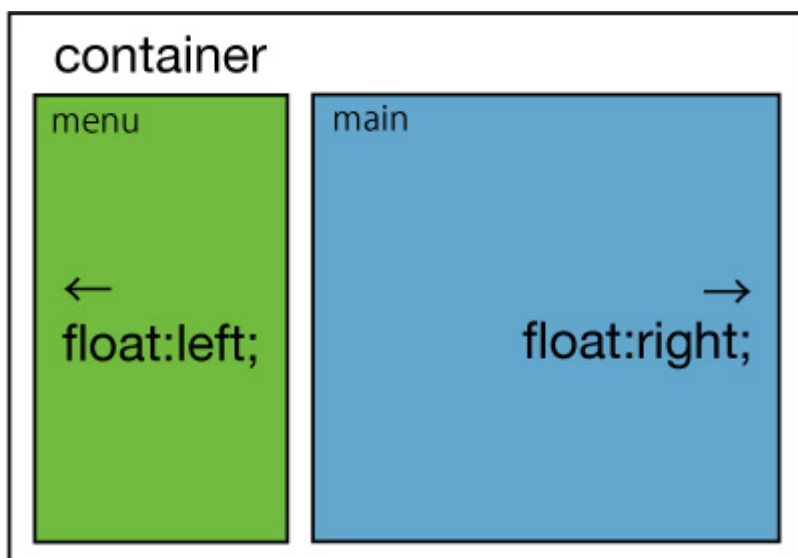
「class」で名前を付けた場合は CSS 側で「.class 名{ }」として設定し、「id」で名前を付けた場合は CSS 側で「#id 名{ }」として設定します。「class」と「id」の違いですが「class」は HTML 内で何度も使えるのに対し「id」は HTML 内で 1 度しか使えません。クラスは同じクラスに何人も入れますが、学籍番号は 1 人 1 つで使い回しが効きませんね。

「<link href="sample.css" rel="stylesheet" type="text/css" />」
HTML 側のこの 1 行は、CSS を指定して読み込む為の宣言です。
href="STYLE SHEET.css"の部分で CSS の URL を指定します。

「<p> </p>」は、段落 (Paragraph) です。

「 」と「 」は箇条書きを作る為のタグで、

「」(Unordered List) 要素は箇条書き全体を表し、(List Item) 要素は項目 1 つを表します。「float」プロパティは、あるボックスが左側に配置するか右側に配置するかを指定するものです。この sample.css では、menu を「float: left;」と指定して左に寄せていますね。main は「float: right;」で右に寄せています。



練習 1

先程のサンプルを変更して、body のデザインを変更しましょう。

CSS に書き込む設定には以下のものをはじめ、様々なものがあります。

background-image: url("IMAGE.jpg"); ...背景画像

background-color: #___; ...背景色

font-size: _pt; ...文字サイズ

font-family: FONT; ...フォント

こちらのリンクも参考にしてみましょう。

[TAGindex](#)

[とほほの WWW 入門](#)

練習 2

更に先程のサンプルを変更して、リンクのデザインを変更しましょう。

リンクの設定はリンクの状態によって変える事が出来ます。

a:link{ } ...通常

a:visited{ } ...アクセス済み

a:hover{ } ...ポイント時

a:active{ } ...クリック時

練習 3

練習 2 ではページ全体のリンクのデザインを設定しましたが、
今度は特定のエリアのリンクのデザインを変更してみましょう。

[HTML]

```
<a class="class 名前" href="URL">link</a>
```

[CSS]

```
a.class 名:link{ }
```

```
a.class 名:visited{ }
```

```
a.class 名:hover{ }
```

```
a.class 名:active{ }
```

練習 4

テーブルについても、リンクの様にまとめて設定をすることが出来ます。

テーブルのデザインを CSS から設定してみましょう。

```
table{ }
```

```
tr{ }
```

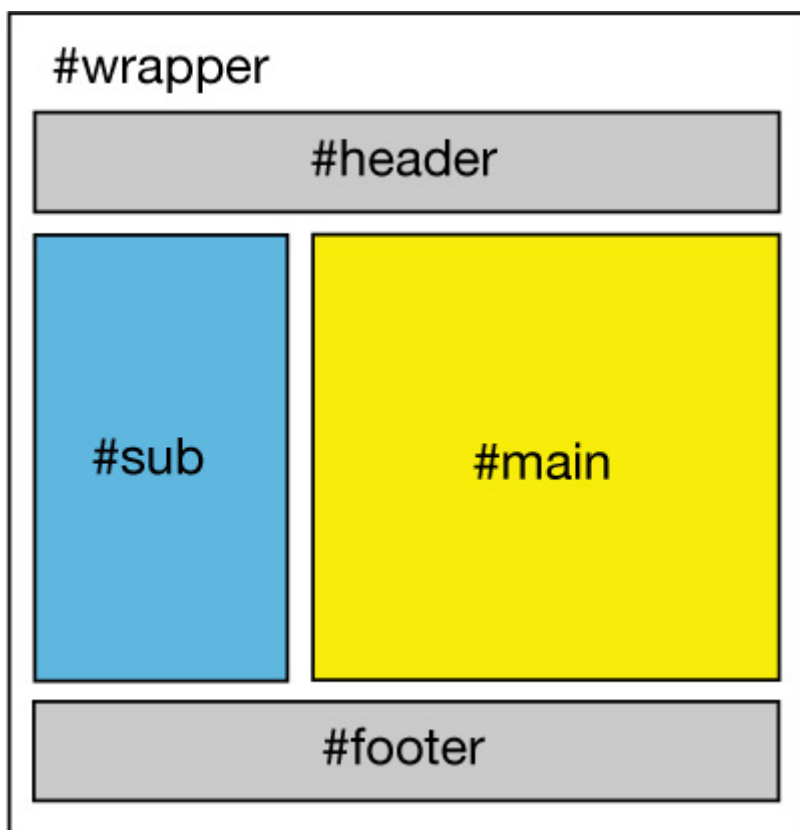
今回の課題は、自分の Web ページを CSS によってデザインする事です。

講義中の「練習 1」から「練習 4」までをクリアして、Web ページをもっと素敵に変身させましょう。授業の内容や、より発展的な内容を積極的に取り入れて、是非素敵な Web ページに仕上げてください。課題を以下の URL から参照出来る様にアップロードして、SFC-SFS にて URL を提出してください。

CSS ファイルを作り、Web の「データ」と「デザイン」を別々にした事で、更新が行い易くなったと思います。前回の課題では、[こんな感じ](#)の Web ページを作って頂きました。しかし、これでは物足りない気がしますね。今回は Web の定番的なレイアウト・パターンを紹介します。CSS によるレイアウトの基本的な手法を理解し、自分の Web の作り込みを試みましょう。

CSS によるレイアウト

前回学習した「div 要素」については理解出来ていますか。今回は先ず、以下の図を参考にして、サンプルの HTML と CSS を自分で作ってみましょう。また下のサンプルは穴埋め方式になっていますから、「?????」の部分を考えて埋めてみてください。



sample.html

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ja" lang="ja">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <link href="sample.css" rel="stylesheet" type="text/css" />
    <title>Sample</title>
  </head>

  <body>
    <div id="???????">

      <div id="???????">
        <h1>header</h1>
      </div>

      <div id="???????">
        <h1>main</h1>

        <h2 class="red">red</h2>
        <p class="red">
          ここは「#main」に含まれる段落内のテキストです。<br />
          ここは「#main」に含まれる段落内のテキストです。
        </p>

        <h2 class="green">Green</h2>
        <p class="green">
          ここは「#main」に含まれる段落内のテキストです。<br />
          ここは「#main」に含まれる段落内のテキストです。
        </p>

        <h2 class="blue">Blue</h2>
        <p class="blue">
          ここは「#main」に含まれる段落内のテキストです。<br />
          ここは「#main」に含まれる段落内のテキストです。
        </p>
      </div>

      <div id="???????">
        <h1>sub</h1>
        <p>
          ここは「#sub」に含まれる段落内のテキストです。
          <ul>
            <li>item</li>
            <li>item</li>
            <li>item</li>
            <li>item</li>
            <li>item</li>
          </ul>
        </p>
      </div>

      <div id="???????">
        <h1>footer</h1>
        <address>&copy; 2009 [Copyright Notice] All Rights Reserved.</address>
      </div>

    </div>
  </body>
</html>
```

sample.css

```
@charset "UTF-8";
/* CSS Document */

a:link{
    color: #0066FF;
    text-decoration: none;
}

a:visited{
    color: #0066FF;
    text-decoration: none;
}

a:hover{
    color: #00CCFF;
    text-decoration: none;
}

a:active{
    color: #00CCFF;
    text-decoration: none;
}

body{
    background-color: #FFFFFF;
    margin: 0;
    padding: 0;
    text-align: center;
}

#wrapper{
    margin: 0 auto;
    width: 600px;
}

#header{
    background-color: #666666;
    width: 100%;
    height: 150px;
}

#main{
    background-color: #FFF59A;
    width: 400px;
    height: 600px;
    float: right;
}

#sub{
    background-color: #9BAEFF;
    width: 200px;
    height: 600px;
    float: left;
}

#footer{
    clear: both;
    width: 100%;
    background-color: #666666;
}

p.red{
    color: #990000;
}

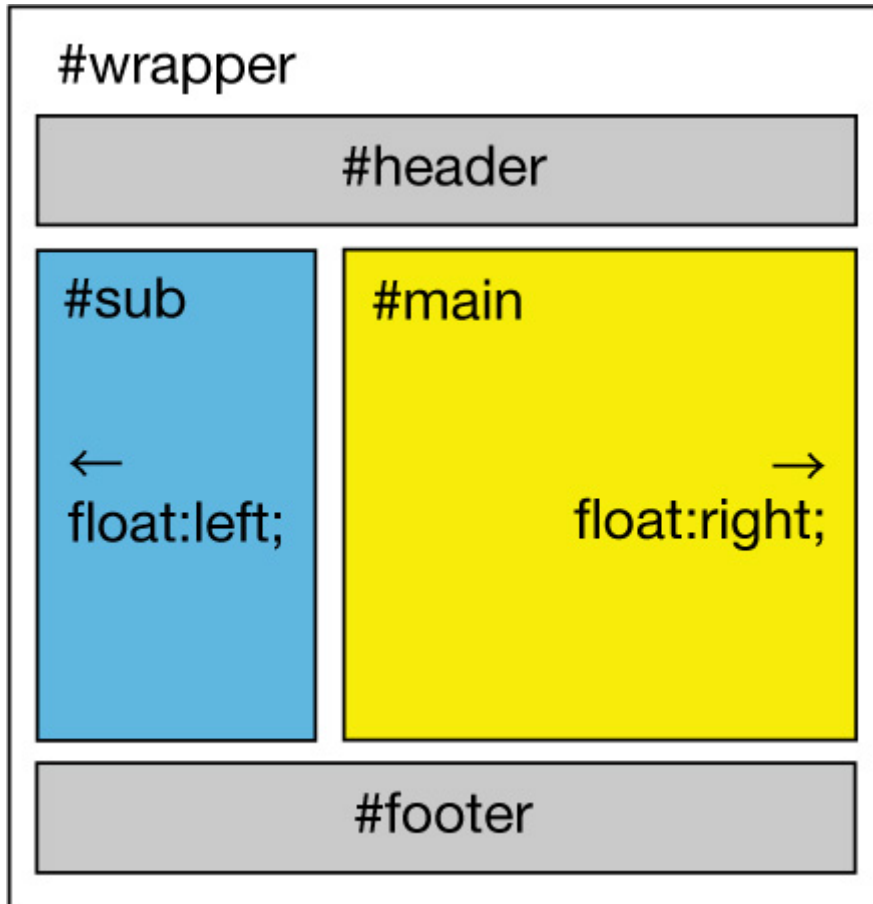
.red{
    color: #990000;
}

.green{
```

きっと、[こんな感じ](#)になると思います。

「<div>」要素

前回の復習です。CSSでは、「<div></div>」というタグを使ってレイアウトを組みます。<div>タグは見出しや文章といった個々のパーツをグループ化する為の Box です。この「sample.html」では以下の図の様な入れ子構造になっています。



「float」プロパティ

復習です。

「float」プロパティは、ある Box が左側に配置するか右側に配置するかを指定しています。

この「sample.css」では、「sub」を「float: left;」と指定して左に寄せていますね。

「main」は「float: right;」で右に寄せています。

上の図を参考にしてください。

「id」と「class」

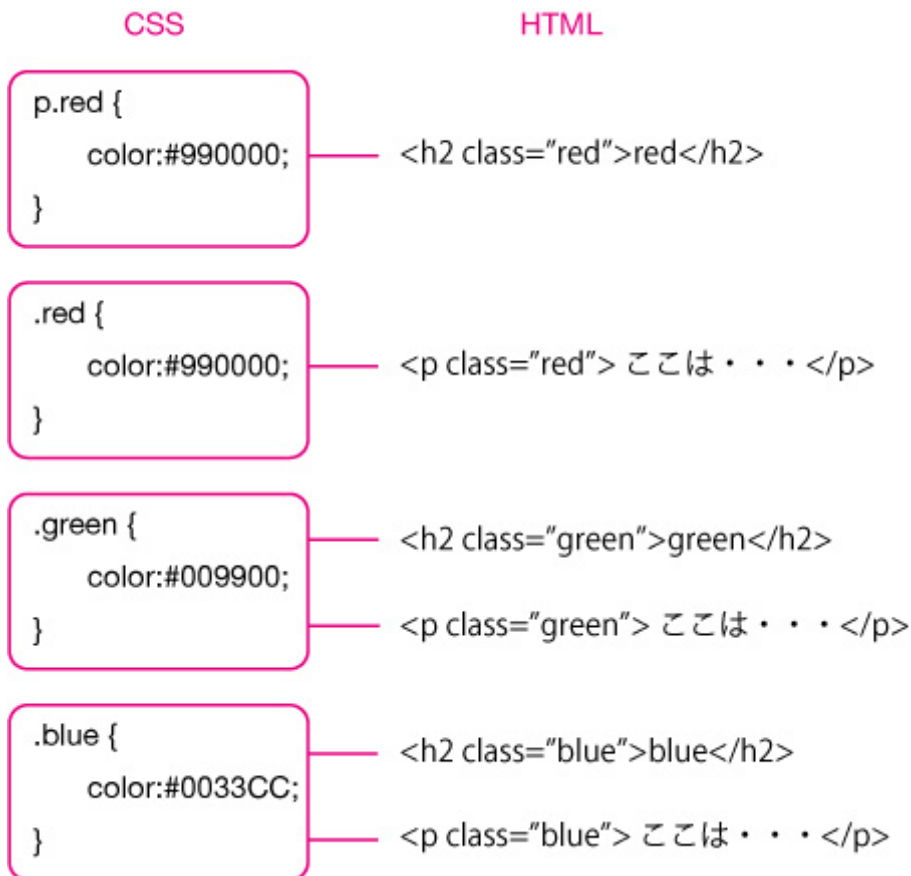
前回の復習です。

id と class の違いは、以下の通りでした。

id : 1 つの HTML ファイルで一度しか使えない

class : 1 つの HTML ファイルで何度も使用できる

サンプルを見てみましょう。

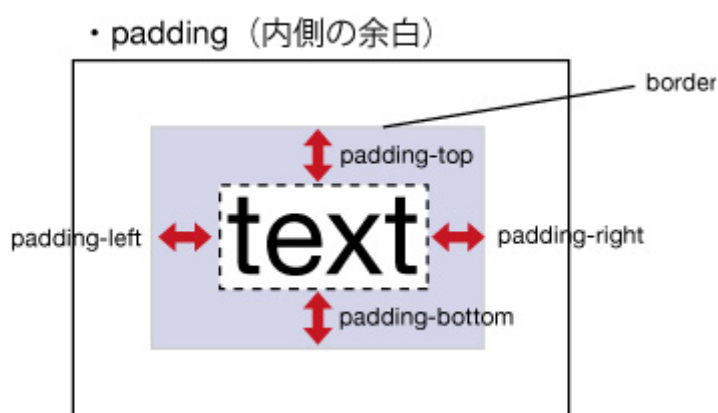
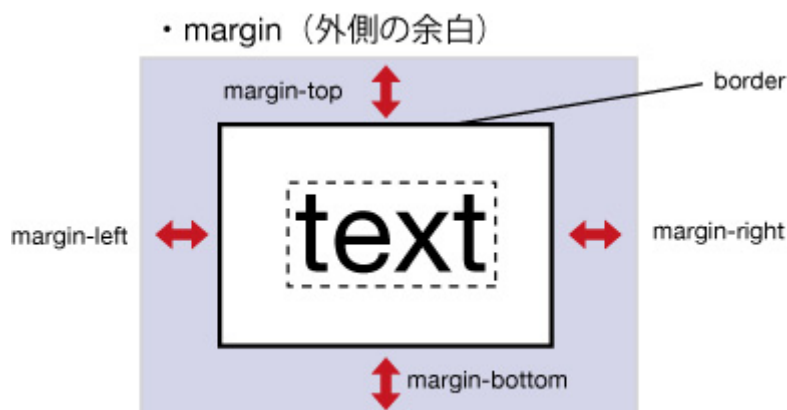


「margin」と「padding」

「margin」プロパティは、コンテンツ領域の外側の余白、

「padding」プロパティは、コンテンツ領域の内側の余白を指定します。

以下の図を参考にしてください。



CSS への指定の仕方は以下の 4 通りです。

padding も指定する位置は同様です。

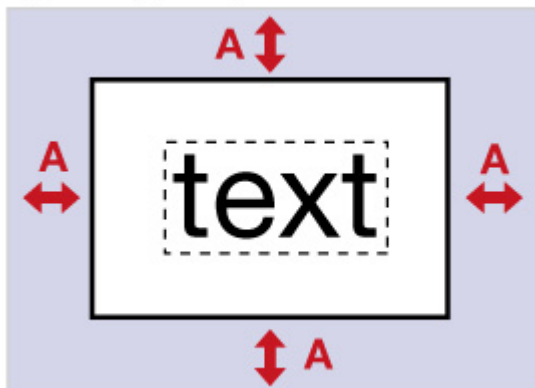
サンプルでは「#wrapper」の中に例があります。

「margin: 0 auto;」と指定する事で左右の外側の余白を auto に設定し、ブラウザの中心に配置することが可能です。

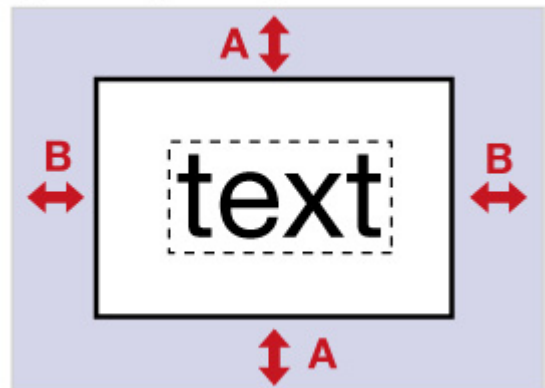
margin と padding は、その pixel 数も全体の幅や高さに含まれます。

レイアウトを組む時には、指定した pixel 数に気を付ける様にしましょう。

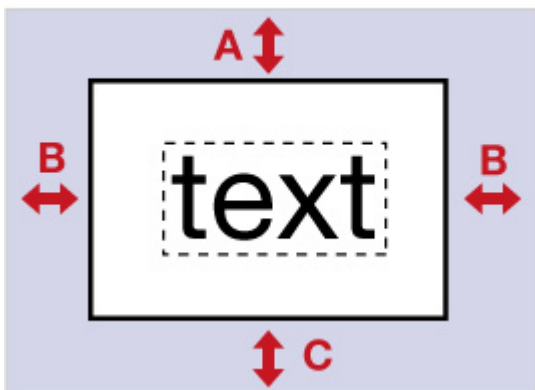
① margin: A;



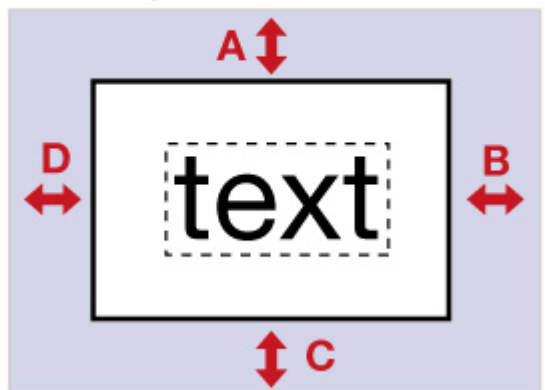
② margin: A B;



③ margin: A B C;



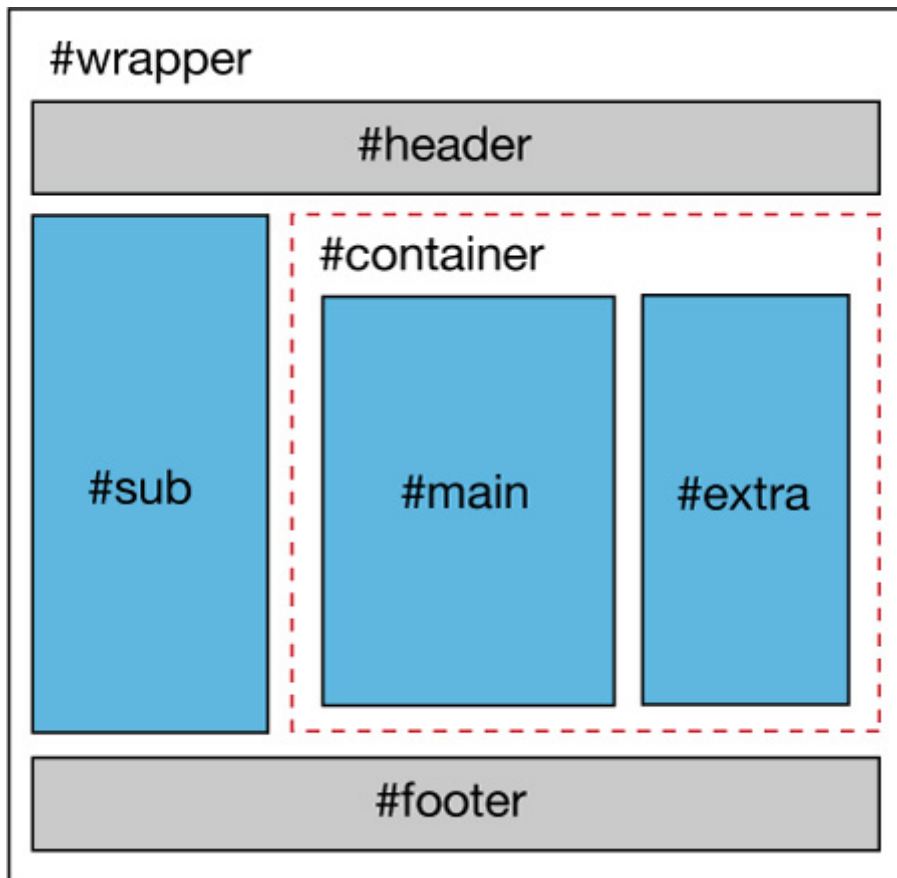
④ margin: A B C D;



3 段組みレイアウト

練習 1

サンプルを参考にして 3 段組みのレイアウトに挑戦してみましょう。



きっと、[こんな感じ](#)になると思います。

Web をデザインする

ブラウザのデフォルト設定によって、フォントや文字サイズ、行間、字間は異なって来ます。

Web の情報を読み易く、伝え易くするためには、テキストの設定が重要となります。

練習 2

背景色、背景画像、フォントを装飾してみましょう。

文字サイズ : font-size

書体 : font-family

字間 : letter-spacing

行間 : line-height

例えば、[こんな感じ](#)。

背景画像などの素材も、自分で作ってみましょう。

練習 3

メニューの作り込みを試してみましょう。
例えば、[こんな感じ](#)にする事も出来ます。

練習 4

以下の Web サイトの CNS のギャラリーを参考にしながら、自分の Web のデザインを考えて下さい。

[Unmatchedstyle](#)

[cssBeauty](#)

[cssvault](#)

[Stylegala](#)

[CSS Drive](#)

[css thesis](#)

[Styleboost](#)

[CSSElite](#)

[CSS Import](#)

[Web Creme](#)

[CSS Mania](#)

[DesignShack](#)

(引用元 :

<http://www.pronetadvertising.com/articles/my-50-favorite-design-resources.html>)

以下のリファレンスを参考にしながら、授業で取り上げていない事についても試してみてください。

[TAGindex](#)

[とほほの WWW 入門](#)

[ZenGarden](#)

課題提出

今回の講義で、Web の色々な設定について学ぶ事が出来たと思います。

講義中の「練習 1」から「練習 4」までをクリアして、Web ページをもっと素敵に変身させましょう。

授業の内容や、より発展的な内容を積極的に取り入れて、是非素敵な Web ページに仕上げてください。

課題を以下の URL から参照出来る様にアップロードして、SFC-SFS にて URL を提出して下さい。

横メニュー

前回と同様に、引き続きメニューの作り込みをしてみましょう。

今回はメニューを横 1 列に並べたデザインを紹介します。

新しく学ぶ事は殆ど無いと言って良いでしょう。

これで Web サイトで多用される「横メニュー」や「縦メニュー」が作れる様になります。

sample.html

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="ja" lang="ja">
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <link href="sample.css" rel="stylesheet" type="text/css" />
    <title>横メニュー</title>
  </head>

  <body>
    <div id="wrapper">
      <div id="header">
        <h1>header</h1>
        <ul>
          <li><a href="">item</a></li>
          <li><a href="">item</a></li>
          <li><a href="">item</a></li>
          <li><a href="">item</a></li>
        </ul>
      </div>

      <div id="main">
        <h2>コンテンツ</h2>
        <p>ここにいるいと内容を書く。</p>
      </div>

      <div id="footer">
        <h2>footer</h1>
      </div>
    </div>
  </body>
</html>
```

sample.css

```
@charset "UTF-8";
/* CSS Document */

body{
  color:#000000;
  font-family:helvetica;
  font-size:12px;
  line-height:15px;
  margin:0;
}

#wrapper{
  background-color:#FFFFFF;
  margin:0 auto;
  width:800px;
}

#header{
  background-color:#CCCCCC;
  width:100%;
  overflow:hidden;
}

#main{
  background-color:#FFFFFF ;
  width:100%;
  min-height:400px;
  float:left;
}

#footer{
  clear:both;
  width:auto;
  background-color:#CCCCCC;
}

h1{
  margin:0;
  padding:20px;
}

h2{
  padding:20px;
}

p{
  padding-left:20px;
}

#header ul{
  background-color:#000000;
  width:100%;
  margin:0;
  padding:0;
  overflow:hidden;
  list-style:none;
  height:34px;
}

#header li{
  float:left;
  text-align:center;
  line-height:34px;
}

#header li a{
  text-decoration:none;
  color:#FFFFFF;
  font-size:14px;
}
```

きっと、[こんな感じ](#)になると思います。

文字の装飾

文字の色、大きさ、間隔、配置等まで手が回っていない人が多い様ですが、サイト全体で統一感や雰囲気を作り出す事はとても大事です。

1つ1つの要素名に対して、丁寧にスタイルを適用してみましょう。

文字サイズ : font-size

書体 : font-family

形 : font-style

色 : color

背景 : background-color

枠線 : border

字間 : letter-spacing

行間 : line-height

文字揃え : text-align

練習 2

統一感や雰囲気を意識しながら、サイト全体のデザインにこだわってみましょう。

[こんな感じ](#)のデザインも、参考にしてみてください。

「margin」や「padding」の設定もきちんと行ってみましょう。

HTML 入門 - 3 :: [「margin」と「padding」](#)

画像の回り込み

綺麗な写真を効果的に配置する事で、Web サイトの雰囲気を格段に向上させる事が出来ます。「<div> </div>」で構造を意識しながら作ってみましょう。

「float」プロパティを使用して、ある Box が左側に配置されるか、右側に配置されるかを指定します。

練習 3

Web ページに画像を配置してみましょう。

例えば、[こんな感じ](#)。

課題提出

HTML や CSS についての学習は、今回で終わりです。

基本的な HTML の書き方やデザインの方法が解って来たと思います。

講義中の「練習 1」から「練習 3」をクリアして、Web ページに仕上げをしましょう。

授業の内容や、より発展的な内容を積極的に取り入れて、是非素敵な Web ページに仕上げてください。課題を以下の URL から参照出来る様にアップロードして、SFC-SFS にて URL を提出してください。