

情報教育演習II 第12回

レイアウトの設定(1)

スタイルシートを使った、Webページのレイアウトを設定をします。

レイアウトの設定(1)

1. [div要素によるグループ化](#)
2. [div要素によるブロック内のレイアウト](#)
3. [幅と高さの設定](#)
4. [左右への配置と回り込みの設定](#)
5. [回り込みの解除](#)

今回の課題

div要素とクラスを組み合わせ、オリジナルの要素をつくり、ページ全体のレイアウトを設定します。

1. [HTMLファイルを作成する](#)
2. [スタイルシートの切り替えの準備をする](#)
3. [全体のレイアウトを設定する](#)

div要素によるグループ化

div要素の機能

div要素は、それだけでは、デザイン（文字の大きさや色など）やレイアウト（揃え方など）が設定されるわけではありません。

div要素を使えば、複数の要素をまとめて、クラスを設定することで、グループ化することができます。次の例は、Webページの内容を「目次」「本文の内容」「フッター」の3つにグループ化した場合です。

```
<div class="menu">
<h2>目次</h2>
<ul>...</ul>
...>
</div>

<div class="contents">
<h2>第1章</h2>
<p>...</p>
<p>...</p>
...
</div>

<div class="footer">
<hr>
<address>.....</address>
</div>

...
```

div要素によるオリジナル要素の設定

div要素とクラスと組み合わせることで、自分だけの意味づけをしたブロックを作ることができます。つまり、「見出し」「段落」などHTMLにあらかじめ定義されている要素以外に、オリジナルの要素を定義することができます。

また、div要素は、複数の要素に対してまとめてスタイルを設定することができます。

たとえば、上記のdiv要素を使った例に、次のようなスタイルを指定することができます。

```
div.menu {
  background: #ffe0e0;
  color: #ff0000;
}
div.contents {
  background: #ffe0e0;
  color: #ff00ff;
}
div.footer {
  color: #ff0000;
  background: #ffff00;
}
```

[次へ](#)進んでください。

div要素によるブロック内のレイアウト

最初に説明したように、div要素を使えば、1つ以上の要素をグループ化することができます。そして、div要素に対してレイアウトの設定をすれば、グループ内のすべての要素にもレイアウトが設定されます。

行揃えによるレイアウト

行の揃え方を設定するには、`text-align`プロパティを設定します。

`text-align`: 行揃え位置

`text-align`プロパティは、行の揃え方を設定します。揃える位置には、次の設定ができます。

- left: 左揃え
- right: 右揃え
- center: 中央揃え

```
div.note {
  text-align: left;
}
div.example {
  text-align: center;
}
```

なお、`text-align`プロパティは、指定したブロックレベル要素に対して、**その中身の揃え方**を設定します。ブロック自体を中央に揃えたい場合には、次のように左右のマージンを「auto」にします。

```
div.note {
  margin-left: auto;
  margin-right: auto;
}
```

その他のレイアウト

div要素は、行揃え以外にも、次のような設定ができます。つまり、ほかの要素と同じようなレイアウトの設定ができます。

- 枠線 (border)
- マージン (margin)
- 余白 (padding)
- 行の高さ (line-height)
- など

また、ここでは説明しませんが、div要素に、文字の種類や大きさ、色や画像などのデザインに関する設定をすることもできます。

[次へ](#)進んでください。

幅と高さの設定

見出し(h1~h6)や段落(p)、さらにdivを使ってグループ化した部分は、**ひとつのブロック**として扱うことができます。

ブロックなどの、内容を表示する部分の幅と高さを設定するには、**width**プロパティと**height**プロパティを設定します。

```
width: 幅  
height: 高さ
```

widthプロパティとheightプロパティは、ブロックレベル要素や画像、テーブル(表)などの、幅や高さを設定でいます。

幅や高さには、次のような設定ができます。

- 単位つき数字: 100px, 30emなど
- %: もとの幅や高さに対する割合
- auto: 状況に応じて自動調整

```
.half {  
  width: 50%;  
}  
.small {  
  width: 50px;  
  height: 50px;  
}
```

[次へ](#)進んでください。

左右への配置と回り込みの設定

ブロックや画像を左右に配置するには、`float`プロパティを設定します。ブロックと文字などの「回りこみ」（配置）を設定することができます。

`float`: 配置する位置

`float`プロパティは、画像や指定した要素を、左右に配置して、その反対側に後続く要素を回りこませます。配置する位置には、次のような設定ができます。

- `left`: 左に配置 (後の要素は右に回りこみ)
- `right`: 右に配置 (後の要素は左に回りこみ)
- `none`: 左右への配置と回り込みをしない

```
div.right {
  border: thin gray dotted;
  margin: 0.5em;
  width: 30%;
  background: #eeeeee;
  float: right;
}
img.left {
  margin: 0.8em;
  float: left;
  clear: right;
}
```

Squeakでプログラミング

Squeak(スクイーク)は、プログラミング環境であり、ひとつのコンピュータシステムでもあります。マウスを使った操作で、命令を組み合わせ、簡単にプログラムを作ることができます。

Squeakって何

Squeakは、フリーソフトです。誰でもダウンロードして利用することができます。海外で作られたためメッセージは英語で表示されますが、現在、日本のボランティアメンバーによって、日本語化されています。

Squeakのダウンロード

- 多言語化Squeakホームページ
(<http://squeak.hp.infoseek.co.jp/>)
- Welcome to Squeak
(<http://www.squeak.org/>)



「Squeak」は「スクイーク」と発音します。
「Squeak」はアラン・ケイ博士が理想とする未来のコンピュータに限りなく近い環境を、WindowsやMacなどの既存のパソコン上に実現する魔法のようなソフトウェアです。博士は、子どもたちの教育用のコンピュータ環境としても理想的なものとなるよう、この「Squeak」を改良し続けられています。

〈アラン・ケイプロジェクト〉
<http://www.edu.city.kyoto.jp/alankay/>より転載

floatを使うときの注意

回り込みの設定は、解除をしないと、設定されたままになります。

回り込みを解除するには、次に説明するclearプロパティを使います。

[次へ](#)進んでください。

回り込みの解除

ブロックや画像などに回り込み (float) の設定をやめるときには、**clear**プロパティを設定します。回り込みの設定を解除することができます。

clear: どちらの側の回りこみを解除するか

clearプロパティは、画像や指定した要素を左右に配置して、後に続く要素を回りこませた場合に、その回り込みを解除します。どちらの要素を解除するかには、次のような設定ができます。

- left: 左側の要素に対する回りこみを解除
- right: 右側の要素に対する回りこみを解除
- both: 両側の要素に対する回りこみを解除
- none: 回り込みを解除しない

```
div.right {  
  margin: 0.8em;  
  padding: 0.5em;  
  float: right;  
}  
p.clear {  
  margin: 0.8em;  
  padding: 0.5em;  
  line-height: 150%;  
  clear: both;  
}
```

[次へ](#)進んでください。

課題(1): HTMLファイルを書く

前回までと同様に、「インターネット上のあらゆる記事へのコメント」を作成します。今回も、2つのファイルを作ります。

- HTMLファイル: コメントをHTMLで書く
- CSSファイル: HTMLファイルのスタイルを書く

1. インターネット上の記事にコメントをする

(1) 情報を集める

自分が現在興味のあるテーマについて、インターネットからニュース等を探して、**自分が興味を持った記事を1つ集めなさい**。

やり方は、第3回の課題と同じです。[第3回の課題の説明](#)を見てください。

(2) 情報にコメントをつける

記事を探したら、次の順にコメントを作りなさい。なお、文字数は、「**200文字以上**」としますが、内容はささいなことでもかまいません。

やり方は、第3回の課題と同じです。[第3回の課題の説明](#)を見てください。

2. HTMLファイルの作成

第11回の課題で作成したファイルをもとに、今回の課題を作成します。

(1) ファイルの作成

第11回の課題のHTMLファイルを開いたあとに、次のようにして、名前をつけて別のファイルを作ります。

- メニューから「ファイル」 「名前をつけて保存」
- ファイルを次のようにつけて保存
 - 「0706」+「学籍番号」+「.html」(半角文字で！)
 - (学籍番号がC2092000の場合、ファイル名は「0706c2092000.html」)

(2) コメントの追加・目次の更新

次のようなレイアウトになるように、今回作ったコメントを追加して、目次の部分を更新します。第2回～第4回までの、HTMLの技術を使って、作成してください。

作成するときには、[第4回の課題での注意点](#)を参考に、今回分のコメントと目次へのリンクを作成します。

ページのレイアウトの例

ページのタイトル

目次

- [第12回のコメントへのリンク](#)
- [第11回のコメントへのリンク](#)
- ... (途中省略) ...
- [第3回のコメントへのリンク](#)
- [第2回のコメントへのリンク](#)
- プロフィール

[第12回の課題のコメント](#)

…
[トップへ戻る]
第11回の課題のコメント
…
[トップへ戻る]
… (途中省略) …
第3回の課題のコメント
…
[トップへ戻る]
第2回の課題のコメント
…
[トップへ戻る]
プロフィール
…
[トップへ戻る]
…
連絡先・著作権の情報

作成するときの注意点は、次のとおりです。

- 目次を更新する
 - 11個のコメント(第2～12回)へジャンプできるリンクをつくる
- 情報源へのURLは、その情報源にアクセスできるようにリンクにすること
- コメントを作る
 - それぞれのコメントの最後には、ページのトップ(一番上)にジャンプする「トップへ戻る」というリンクを作る
- [第3回の課題で説明した注意点](#)を守る
- ファイルの最後にある、連絡先の「更新日」の日付を更新する

```
<address>  
作成日: 2010-04-20; 更新日: 2010-07-06<br>  
Copyright (C) 2010 兵庫太郎, All rights reserved.<br>  
このページに関する問合せ先:<br>  
E-Mail. <a href="mailto:c2092000@ed.hyogo-dai.ac.jp">c2092000@ed.hyogo-dai.ac.jp</a>  
</address>
```

まだ提出ではありません。 [次へ](#)に進んで、課題をすすめてください。

課題(2): スタイルシートの切り替えの準備をする

HTMLファイルを編集して、スタイルシートを切り替えるタグを追加します。

切り替えのためのタグを記述する

前回、HTMLファイルに、スタイルシートを切り替えるタグを追加しました。

今回は、さらにもう一つ、スタイルシートを切り替えるタグを追加します。スタイルシートの切り替え部分のあとに、次の1行を追加してください。

```
<a href="#" onclick="replace_css('cssid','0706学籍番号.css');">第12回</a>  
(注:「学籍番号」には、自分の学籍番号(c209200X)を入力してください)
```

これは「第12回」という文字をクリックしたら、スタイルシートを「0706学籍番号.css」で設定したスタイルシートに切り替えるという意味になります。

結果として、スタイルシートの切り替え部分は次のようになるはずです。確認してください。

```
<a href="#" onclick="replace_css('cssid','0622学籍番号.css');">基本</a>  
<a href="#" onclick="replace_css('cssid','0629学籍番号.css');">第11回</a>  
<a href="#" onclick="replace_css('cssid','0706学籍番号.css');">第12回</a>  
<a href="#" onclick="replace_css('cssid','');">なし</a>  
(注:「学籍番号」には、自分の学籍番号(c209200X)を入力してください)
```

[次へ](#)に進んでください。

課題(3): 全体のレイアウトの設定

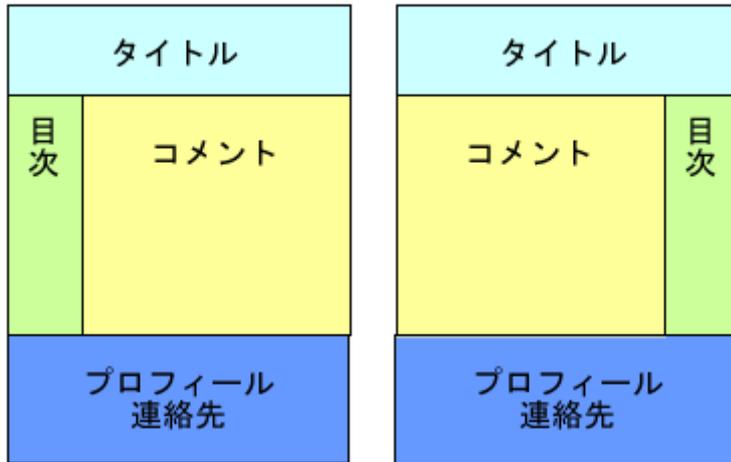
Webページ全体のレイアウトを、スタイルシートを使って設定します。

「タイトル」「目次」「コメント」「プロフィール」のように、Webページをいくつかのグループに分割するために、div要素を使います。そして、CSSファイルにレイアウトのための設定をします。

なお、**すでに自分で設定できている場合は、あらためて書き換える必要はありません。**

1. HTMLにレイアウトを設定する

次の図のような、目次を左や右に配置するレイアウトを設定します。説明にしたがって、設定をしてください。



まず、HTMLファイルのbody要素内に、div要素を次のように追加します。これは、中身を「コンテンツ全体」「タイトル」「目次」「コメント(文章)」「プロフィールと連絡先」の4つのブロックにわけするための設定です。

```
<div class="all">  
  
<div class="header">  
  (タイトルの部分)  
</div>  
  
<div class="sidebar">  
  (目次の部分)  
</div>  
  
<div class="contents">  
  (コメント(文章)の部分)  
</div>  
  
<div class="footer">  
  (プロフィールと連絡先の部分)  
</div>  
  
</div>
```

つまり、次のような意味の設定になります。

1. ページ全体に「all」というクラス名のブロックを作る

2. その中に「header」「sidebar」「contents」「footer」というブロックを置く

2. スタイルシートを設定する

(1) CSSファイルの作成

レイアウトの設定が書かれたCSSファイルを作成します。

今回は、新しいCSSファイルを作成します。ファイル名は、次のとおりです。

- 「0706」+「学籍番号」+「.css」(半角文字で！)
(学籍番号がC2092000の場合、ファイル名は「0706c2092000.css」)

(2) スタイルシートの設定

まず、次の設定を追加してください。

- すべてのブロックのmarginとpaddingを0に設定
- 全体のブロックの幅を700pxにする
(幅は自由に変更してください)
- タイトルのブロックの幅を100%にする(全体のブロックの横幅にする)

```
.all, .header, .sidebar, .contents, .footer
{
  margin: 0;
  padding: 0;
}
.all {
  width: 700px;
}
.header {
  width: 100%;
}
```

次に、目次を左右のどちらかに設定します。[今回説明した](#)幅の設定や左右への配置・回り込みの設定を使います。

目次を左に配置する場合は、次のようにスタイルシートに設定を追加します。

```
.sidebar {
  float: left;
  width: 200px;
}
.contents {
  float: right;
  width: 500px;
}
```

これは、次のような意味になります。

- 目次のブロックを左に配置
- 目次のブロックの幅を200pxに指定
(幅は自由に変更してください)
- コメントのブロックを右に配置(「float: left」を指定しても結果は同じです)
- コメント(文章)のブロックの幅を500pxに指定
(幅は自由に変更してください)

また、目次を右に配置する場合は、次のようにスタイルシートに設定を追加します。

```
.sidebar {
  float: right;
  width: 200px;
}
.contents {
  float: left;
  width: 500px;
}
```

これは、次のような意味になります。

- 目次のブロックを右に配置(その後のブロックは左に配置)
- 目次のブロックの幅を200pxに指定
(幅は自由に変更してください)
- コメントのブロックを左に配置(「float: right」を指定しても結果は同じです)
- コメント(文章)のブロックの幅を500pxに指定
(幅は自由に変更してください)

最後に、プロフィールと著作権のブロックに設定をします。

```
.footer {
  clear: both;
  width: 100%;
}
```

これは、次のような意味になります。

- 回り込みを解除する(左右へのレイアウトを解除する)
- ブロックの幅を100%にする(全体のブロックの横幅にする)

また、次のようにbody要素と.allクラスへ設定すれば、全体をセンタリング(中央揃え)することができます。試してみてください。

```
body {
  text-align: center;
}
.all {
  width: 700px;
  margin-left: auto;
  margin-right: auto;
  text-align: left;
}
```

注意：

Windows版のInternet Explorer 5.5などでは、「margin- : auto」に対応していないため、「text-align: center」でブロックをセンタリングするバグを利用しています。

課題の提出

提出方法

作成したファイルと**使用してる画像ファイルすべて**を、次の場所にコピーして提出してください。

- 「マイコンピュータ」 「Fsの資料・課題」 「kawano」 「提出」

提出期限

- 原則として、**平成22年7月13日9時**までとします。
 - 期限以後は、特別な理由がない限り、提出を受け付けません。
 - 課題のやりなおしは、提出期限まで受け付けます。
-