

CSSレイアウト基礎実習

中央配置と横並び

株式会社ジードライブ

本講義のトピック

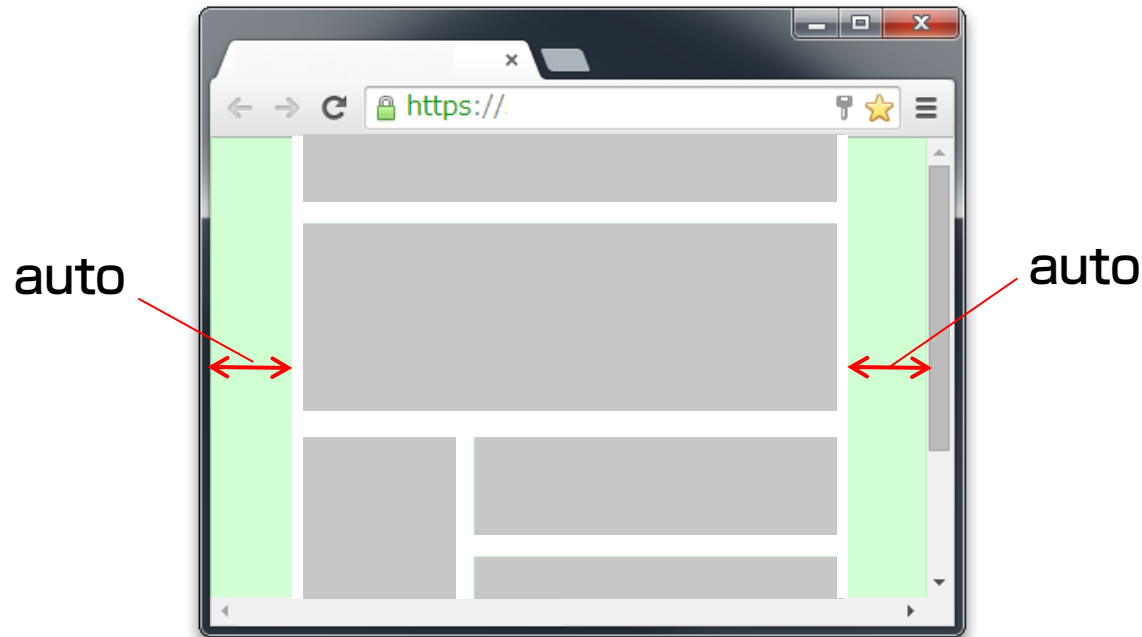
コンテンツを左右中央に配置する方法、
および横並びに配置する方法を学習する

- 中央配置
 - `margin: auto;`
- 横並び
 - 回り込みと回り込みの解除
 - `clearfix`

中央に配置する

コンテンツの中央配置

- ブラウザ内にWebサイトを表示する場合は、中央配置（または全画面表示）が一般的



ボックスには外側余白(margin)を自動配分する**auto**値を使用する

ボックスの中央配置

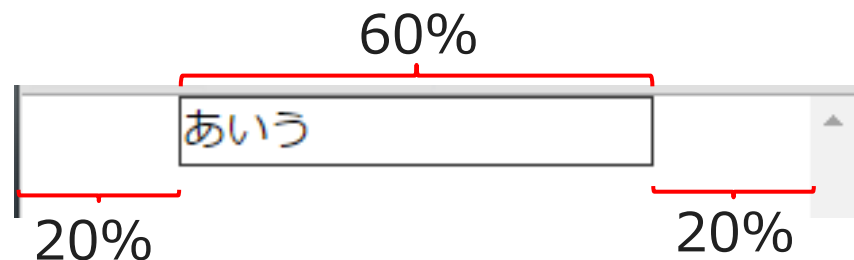
- 必ず width の設定(px, em, %)の指定を行ってから、左右のmarginの値をautoにする

HTML書式

```
<div>あいう</div>
```

CSS書式(+sanitize.css)

```
div {  
  border: 1px solid #333;  
  width: 60%;  
  margin: 0 auto;  
}
```



$(100-60) \div 2 = 20\% = \text{片側の余白分}$

左右のmarginを自動で算出

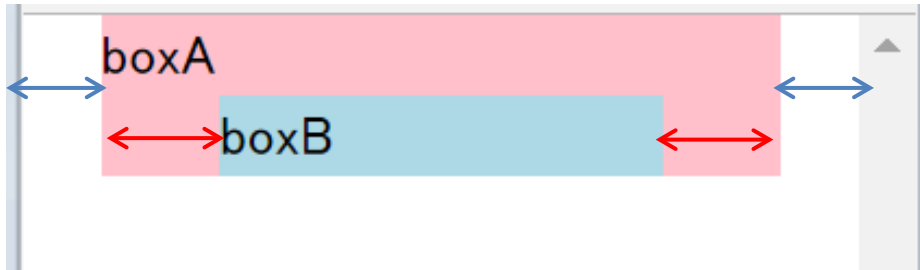
上下のmarginをautoにしても中央に配置されない

ボックス内での中央配置

- ボックス内のボックスを中央配置する場合も同様に、幅を決めて、左右のmarginをautoにする

HTML書式

```
<div class="boxA">boxA  
  <div class="boxB">boxB</div>  
</div>
```



CSS書式(+sanitize.css)

```
.boxA {  
  background: pink;  
  width: 200px;  
  margin: 0 auto;  
}  
.boxB {  
  background: lightblue;  
  width: 130px;  
  margin: 0 auto;  
}
```

ボックス内での中央配置

- ボックス内の文字や画像を中央に配置する場合は **text-align: center;** を指定する

HTML書式

```
<div>あいう<br>  
  
</div>
```

CSS書式(+sanitize.css)

```
div {  
  border: 1px solid #333;  
  width: 100px;  
  margin: 0 auto;  
  text-align: center;  
}
```



画像を絵文字と捉えると考えやすい

練習問題

- 練習03-1 を行う

横並びに配置する

回り込みとは

- 要素を横に並べる方法の1つ
- 画像に対して文章を回り込ませたり、ボックスを横並びに配置するために用いられる



ナビゲーションの要素である li を横並びに配置させるためにも用いられる

floatプロパティ

- 指定した要素の後に続く内容を回り込ませる

値	説明
left	要素を左に寄せる（後に続く内容は右側に回り込む）
right	要素を右に寄せる（後に続く内容は左側に回り込む）
none	標準表示（初期値）

※floatは直訳で**浮く**という意味。

指定する要素が手前に少し浮くイメージを持つと、floatの持つ特性や、表示結果を理解しやすくなる

文字の回り込み

- 画像に対して文字を回り込ませる

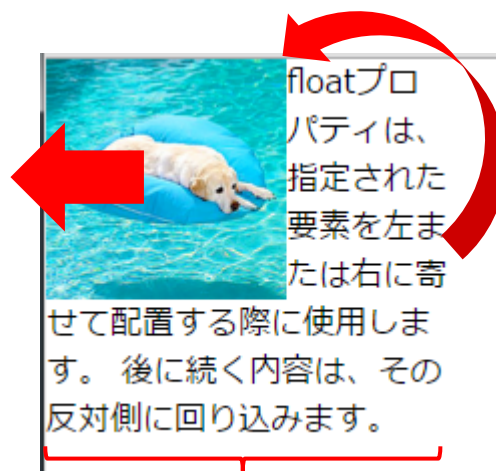
HTML書式

<p>floatプロパティは、指定された要素を左または右に寄せて配置する際に使用します。以下略</p>

CSS書式

```
p {  
  width: 200px;  
}  
img {  
  float: left;  
}
```

画像が左にfloatし、
文字が右側に回り込む



親のp要素200px幅

文字の回り込み

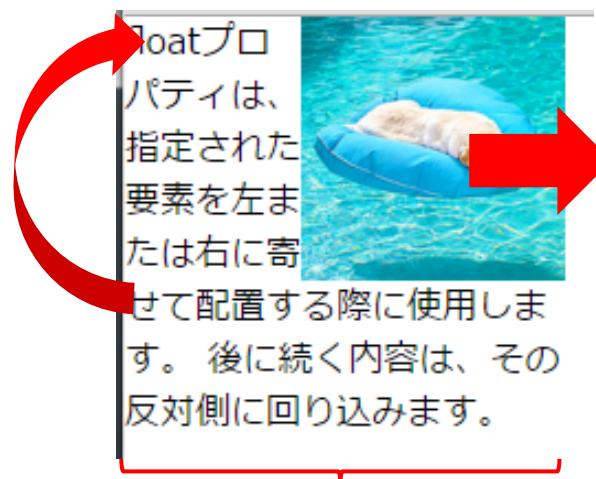
HTML書式

<p>floatプロパティは、指定された要素を左または右に寄せて配置する際に使用します。以下略</p>

CSS書式

```
p {  
  width: 200px;  
}  
img {  
  float: right;  
}
```

画像が右にfloatし、
文字が左側に回り込む



親のp要素200px幅

練習問題

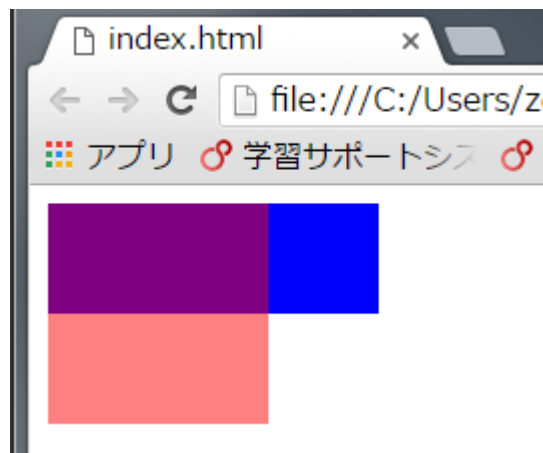
- 練習03-2 を行う

floatの特性

- floatをかけたボックスが上に浮き、続くボックスはその下に潜り込む

HTML書式

```
<div class="a"></div>  
<div class="b"></div>
```



CSS書式

```
.a {  
  width: 100px;  
  height: 100px;  
  opacity: 0.5;  
  background-color: red;  
  float: left;  
}  
.b {  
  width: 150px;  
  height: 50px;  
  background-color: blue;  
}
```

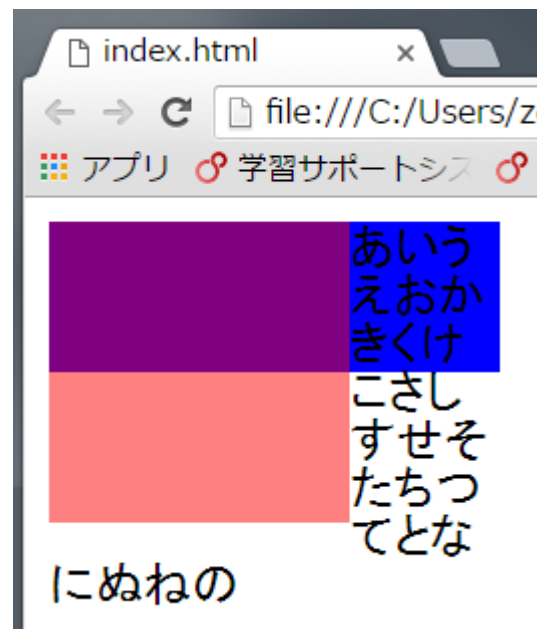
floatの特性

- 続くボックスは潜り込むが、文字は回り込む

HTML書式

```
<div class="a"></div>  
<div class="b">  
あいうえおかきくけこ（以下略）  
</div>
```

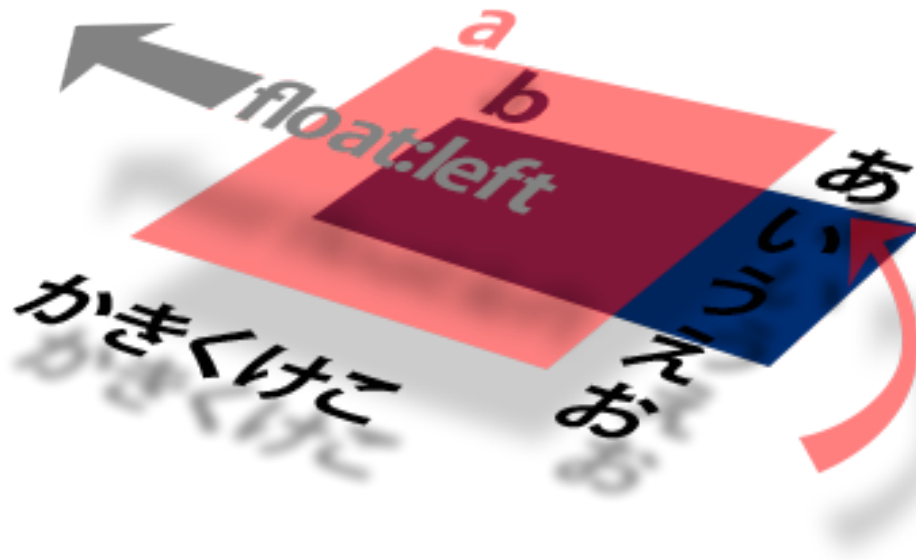
※CSSは前頁と同じ



ボックス b は潜り込んだまま
文字だけが a に回り込んでいる

floatの特性

- 立体的な視点で見ると浮いたfloatの要素に文字が浮きながら回り込むイメージ



floatの特性

- floatをかけたボックスはサイズ幅の値を失う

HTML書式

```
<div class="a">あいう</div>
```

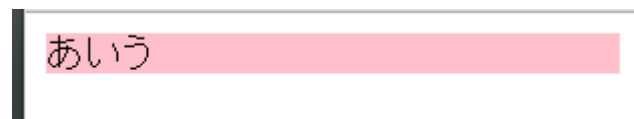
CSS書式

```
.a {  
  background-color: pink;  
}
```



CSS書式

```
.a {  
  background-color: pink;  
  float: left;  
}
```



幅の指定をしない場合、デフォルトで、親ボックス幅いっぱい広がる



floatをかけると左右幅が、内容の文字幅分だけに縮まってしまう

中に何も入ってなければ表示されずに潰れてしまうので、floatをかける際には幅の**サイズ指定を行う**ようにしておく

ボックスを横に並べる

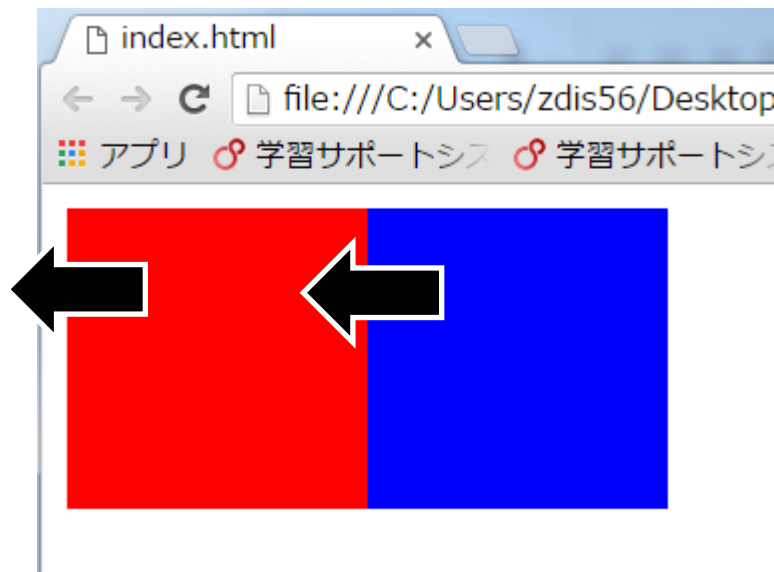
- 両方のボックスにfloatをかける

HTML書式

```
<div class="a"></div>  
<div class="b"></div>
```

CSS書式

```
div {  
  width: 100px;  
  height: 100px;  
.a {  
  background-color: red;  
  float: left;  
}  
.b {  
  background-color: blue;  
  float: left;  
}
```



ボックスを横に並べる

- 前頁と同様のHTMLとCSSでfloatのみ変更



float: left;

float: right;

親ボックス（この場合はウィンドウ）
の幅いっぱい左右に広がる

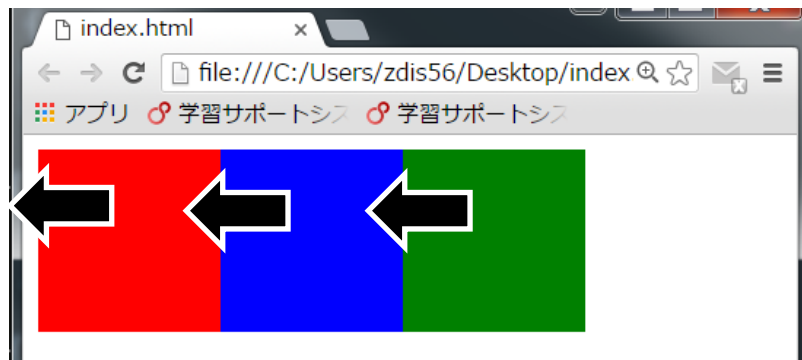


float: right; float: right;

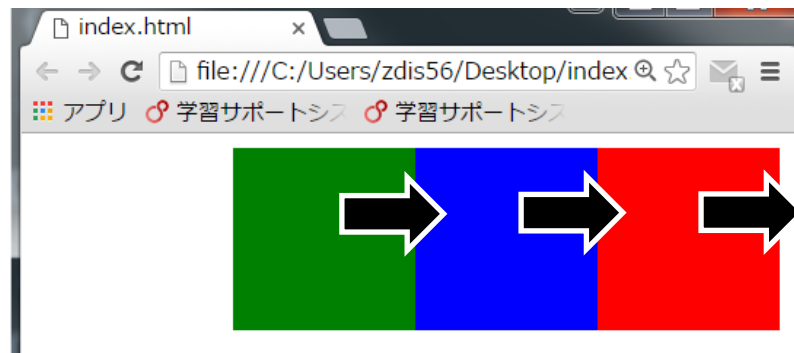
右から順番に並ぶ

ボックスを横に並べる

- 同様の設定の**ボックスC**を追加



float: left; float: left; float: left;

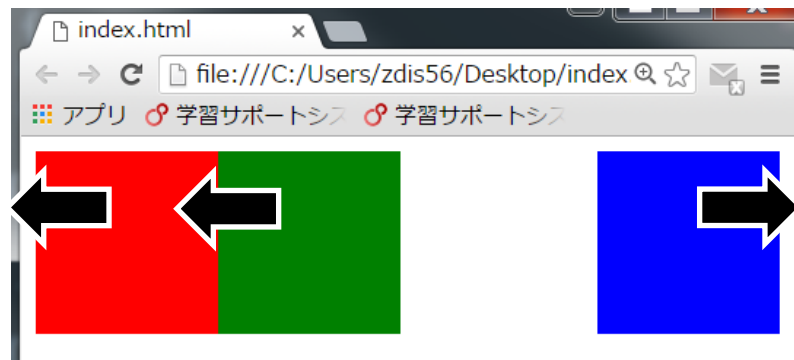


float: right; float: right; float: right;



float: left;

float: right; float: right;



float: left; float: left;

float: right;

ボックスを横に並べる

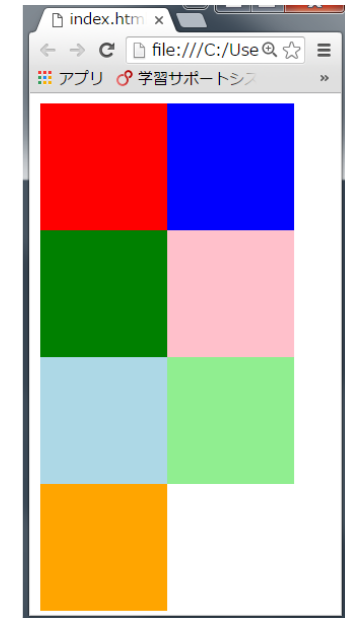
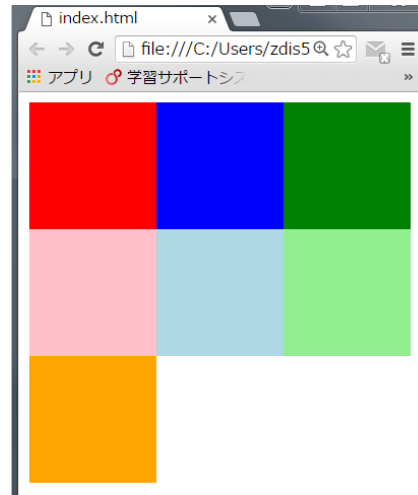
- 親ボックスの幅に収まらない場合、次の行に折り返す



すべてのボックスに
`float:left`を設定

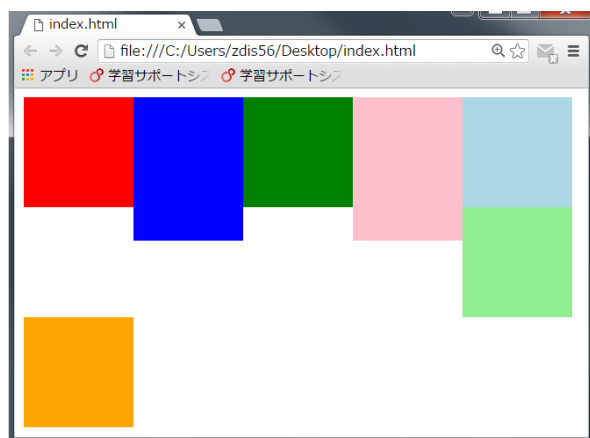
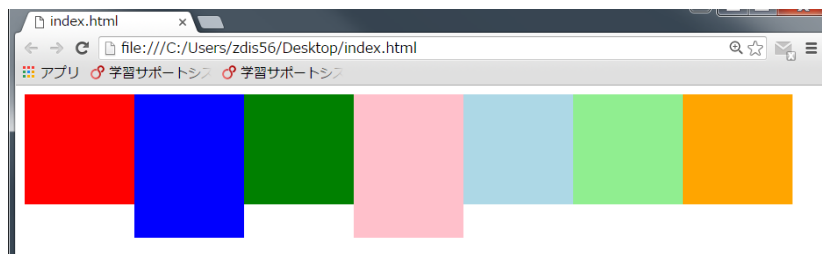


親ボックスの幅が狭まると
次の行に折り返す



ボックスを横に並べる

- 高さの違うボックスの場合、きれいに折り返せない



きれいに並べたい場合は、clearプロパティやJavaScriptなどを利用する

ボックスを横に並べる

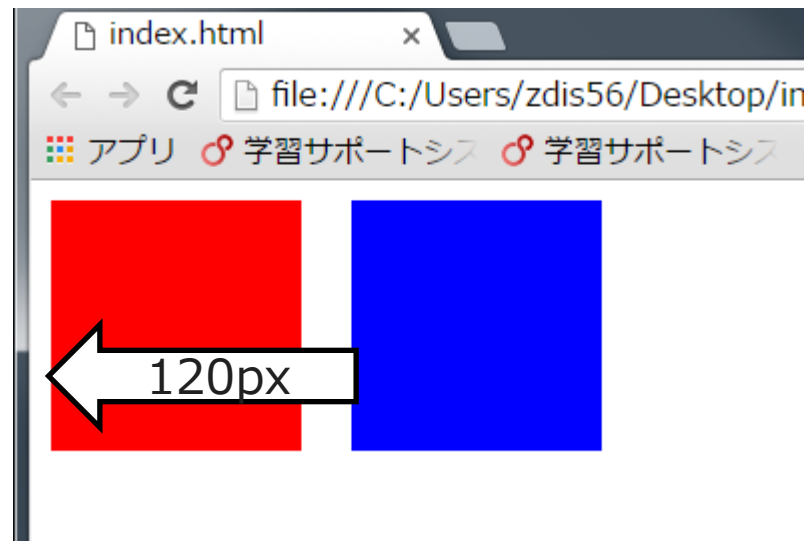
- margin指定で横並びにすることも可能

HTML書式

```
<div class="a"></div>  
<div class="b"></div>
```

CSS書式

```
div {  
  width: 100px;  
  height: 100px;  
.a {  
  background-color: red;  
  float: left;  
}  
.b {  
  background-color: blue;  
  margin-left: 120px;  
}
```



赤ボックスはfloatで浮いているので、
浮いていない青ボックスのマージン
設定は下に潜り込む

練習問題

- 練習03-3 を行う

回り込みの解除

- floatをかけると後に続く要素は全て回り込んでいく
⇒ floatの影響を解除しないとレイアウトが崩れてしまう



floatの影響によるレイアウトの崩れ



フッターのボックスが背面に隠れ、
文字だけが隙間に回り込んでいる状態

clearプロパティ

- floatの回り込みの流れを解除する

値	説明
left	float:left の影響を解除する
right	float:right の影響解除する
both	float:left/right 両方の影響を解除する
none	標準表示する（初期値）

※floatの左右をその都度確認するのが手間なので、一般的には**both**で両方に対応させることが多い。

clearプロパティの利用

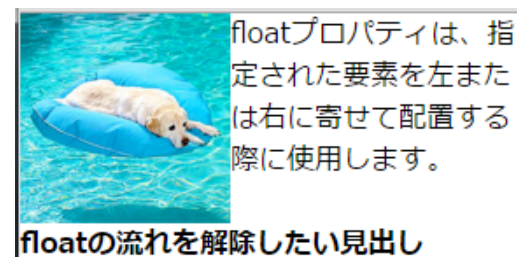
- 画像に対する文字の回り込みを解除する

HTML書式

```
<div>  
  
floatプロパティは、指定された要素を左または  
右に寄せて配置する際に使用します。  
<h2>floatの流れを解除したい見出し</h2>  
</div>
```

CSS書式

```
img {  
    float: left;  
}  
h2 {  
    clear: both;  
}
```

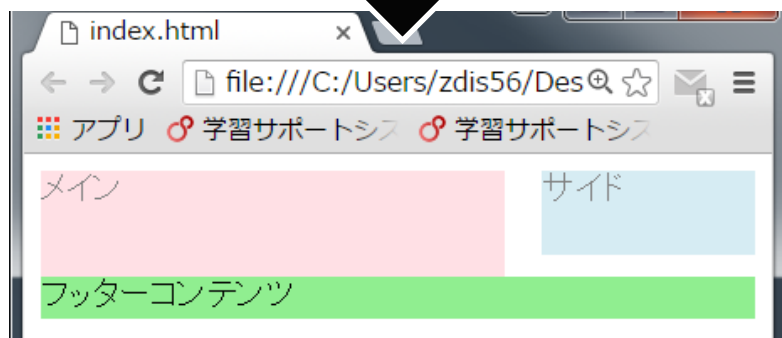
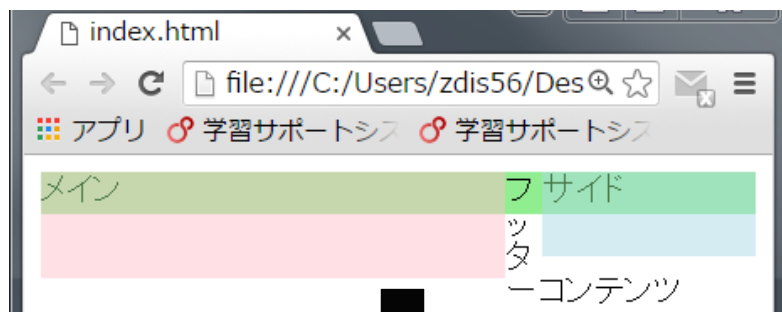


clearプロパティの利用

- ボックスの回りこみを解除する

HTML書式

```
<div class="main">メイン</div>  
<div class="side">サイド</div>  
<div class="footer">フッターコンテンツ</div>
```

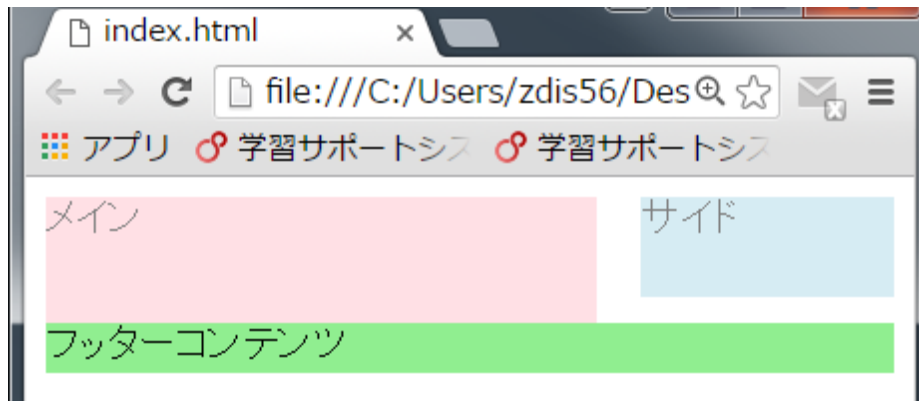


CSS書式

```
.main {  
  width: 65%;  
  height: 50px;  
  opacity: 0.5;  
  background-color: pink;  
  float: left;  
}  
.side {  
  width: 30%;  
  height: 40px;  
  opacity: 0.5;  
  background-color: lightblue;  
  float: right;  
}  
.footer {  
  height: 20px;  
  background-color: lightgreen;  
  clear: both;  
}
```

clearプロパティとmargin

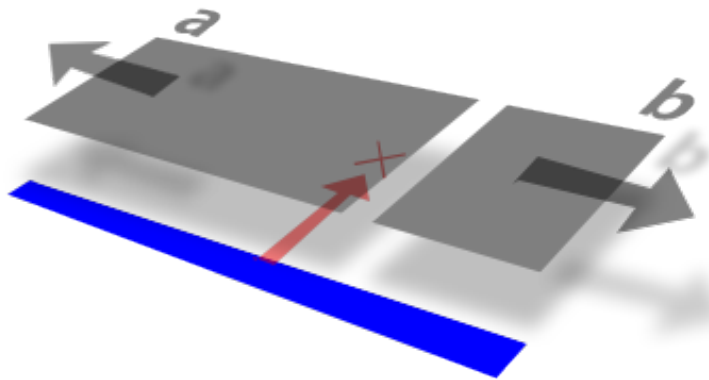
- clearを設定したボックスのmargin-topをfloatしているボックスは認識しない（下に潜り込む）



CSS書式

… 省略 …

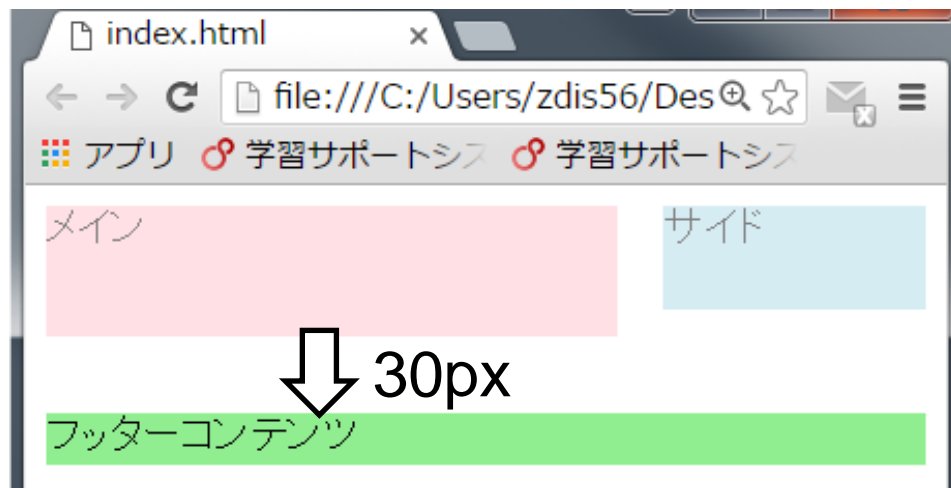
```
.footer {  
  height: 20px;  
  background-color: lightgreen;  
  clear: both;  
  margin-top: 100px;  
}
```



clear:bothとmargin-top
の併用は難しい

clearプロパティとmargin

- marginを設定する場合は、floatしているボックスのmargin-bottomを指定する



CSS書式

```
.main {  
  width: 65%;  
  height: 50px;  
  opacity: 0.5;  
  background-color: pink;  
  float: left;  
  margin-bottom: 30px;  
}
```

... 省略 ...

練習問題

- 練習03-4 を行う

floatの問題点

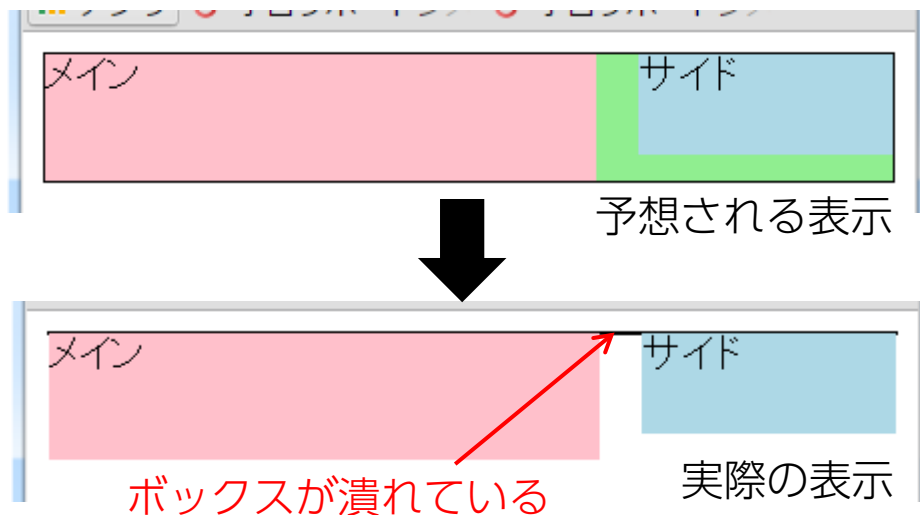
- **clearする要素がない場合**、後続く要素が影響を受け続けるだけでなく、floatするボックスを内包する**親ボックスが潰れてしまう**という問題がある

HTML書式

```
<div class="wrap">  
  <div class="main">メイン</div>  
  <div class="side">サイド</div>  
</div>
```

CSS書式

```
.wrap {  
  background-color: lightgreen;  
  border: 1px solid black;  
}  
.main {  
  width: 65%;  
  height: 50px;  
  background-color: pink;  
  float: left;  
}  
.side {  
  width: 30%;  
  height: 40px;  
  background-color: lightblue;  
  float: right;  
}
```



clearfix

- floatに影響される問題を解決する方法をclearfixという

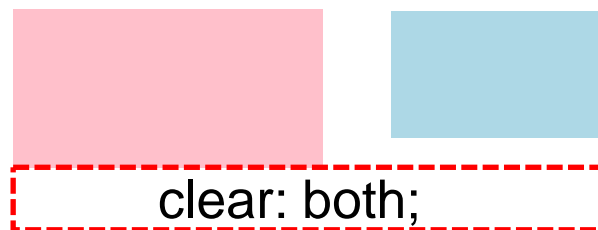
HTML書式

```
<div class="wrap clearfix">
  <div class="main">メイン</div>
  <div class="side">サイド</div>
</div>
```

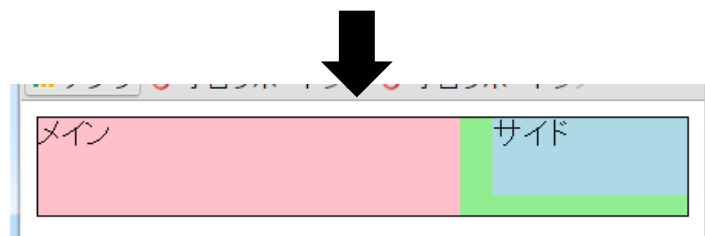
CSS書式

… 省略 …

```
.clearfix:after {      /*指定箇所の後ろに追加*/
  content: "";        /*空のコンテンツを作成*/
  display: block;    /*ブロックボックスに設定*/
  clear: both;      /*回り込み解除*/
}
```



目に見えないボックスを生成し、
clear: bothを適用することで
floatの影響を解除している



※レガシーブラウザに対するため、時代ごとに設定が異なる

簡易的なclearfix

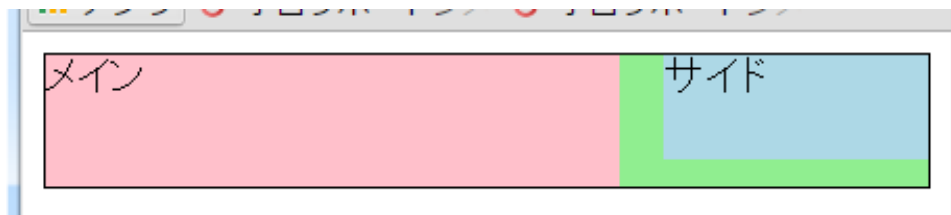
- 親ボックスに `overflow: hidden;` を設定する

HTML書式

```
<div class="wrap">  
  <div class="main">メイン</div>  
  <div class="side">サイド</div>  
</div>
```

CSS書式

```
.wrap {  
  background-color: lightgreen;  
  border: 1px solid black;  
  overflow: hidden;  
}  
... 省略 ...
```



`overflow: hidden;` は、プルダウンメニューやシャドウのように親要素の外にはみ出す要素を隠してしまう。必要に応じて、前頁のclearfixと使い分ける。

overflowプロパティ

- ボックスから溢れてしまうコンテンツの表示に関する指定を行う
- 上記が本来の用途だが、floatに関する問題を解決するためにも用いられる

値	説明
visible	はみ出した部分も表示される(初期値)
hidden	はみ出した部分は表示されない
scroll	常にスクロールバーが表示され、はみ出さない
auto	文字量がボックス領域を超えた時、スクロールバーを表示

overflowプロパティは、内包する全ての要素をマスクする必要があるので、floatのように浮遊した階層の要素も、全て統一して認識する機能を持っている

練習問題

- 練習03-5 を行う