

# CSSレイアウト基礎実習

## ボックスモデル

株式会社ジードライブ

# 本講義のトピック

---

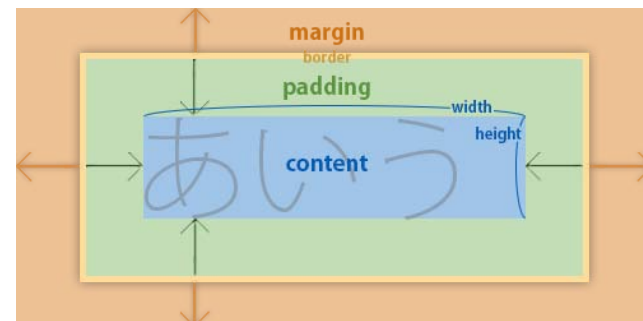
レイアウトをしていく上で基本となる  
ボックスモデルの概念を学習する

- ボックスモデルとは
  - 内容の領域、内側の余白、境界線
  - ボックス同士の間隔
- ボックスモデルの特徴
  - ボックスのサイズと背景画像の表示領域
  - マージンの相殺と加算
  - blockとinline

# ボックスモデルとは

- 要素を矩形の領域（ボックス）で捉える概念
  - ボックスは、内容領域・パディング・ボーダーから成る
  - ボックス同士の間隔をマージンという

用語	説明
内容の領域 Content Area	要素を表示するための領域。 width, height プロパティで任意の幅と高さを指定可能
padding	内容領域とborderの間の余白。背景色や背景画像が表示される
border	ボックスの境界線。太さ、線種、色を指定することができる
margin	他のボックスとの間隔

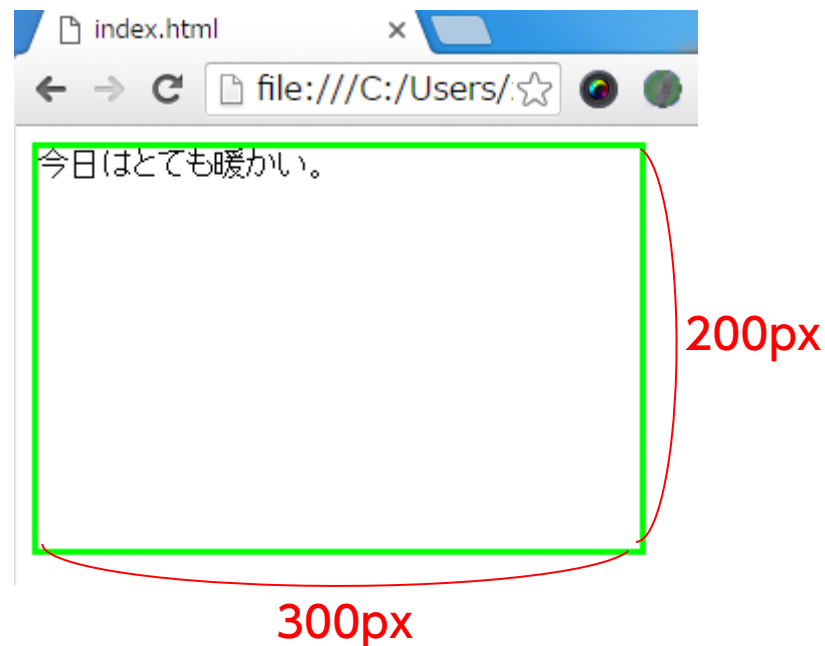


# 内容の領域

- 文字や画像(要素)が表示される領域
- **幅(width)**や**高さ(height)**の指定ができる

```
<p>今日はとても暖かい。</p>
```

```
p {  
  border: 3px solid #0F0;  
  width: 300px;  
  height: 200px;  
}
```



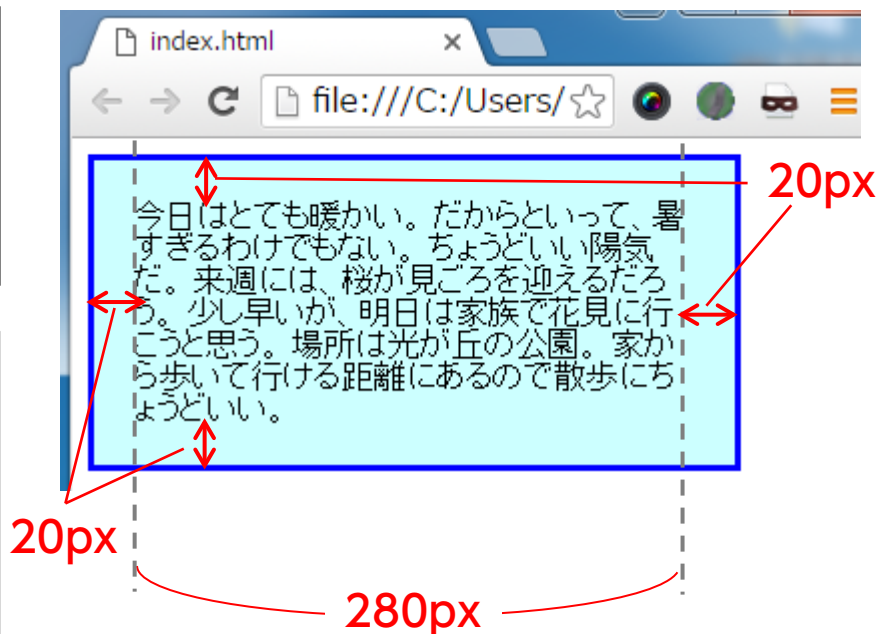
幅や高さの指定がない場合、幅は親要素いっぱいになり、高さはコンテンツ量(文章量)に応じる

# 内側の余白 padding

- 境界線と内容領域の間の余白
- 背景色／背景画像は、この余白の部分にも表示される

```
<p>  
今日はとても暖かい。今日はとても暖  
かい。（省略）家から歩いて行ける距  
離にあるので散歩にちょうどいい。  
</p>
```

```
p {  
  background-color: #CFF;  
  border: 3px solid #00F;  
  width: 280px;  
  padding: 20px;  
}
```



# 境界線 border

- ボックスの境界線の幅、線のスタイル、色を指定
  - 線幅、線種、色の記述は順不同。線種の指定は必須

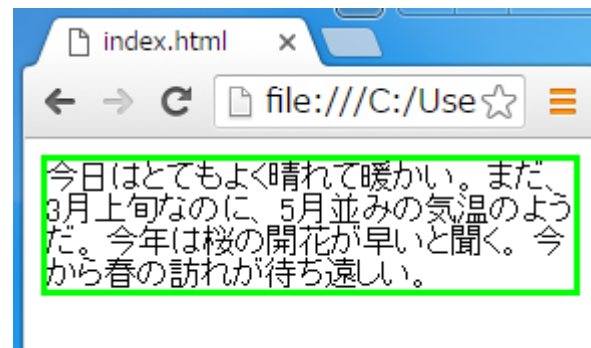
```
<p>  
今日はとてもよく晴れて暖かい。  
まだ、3月上旬なのに、  
5月並みの気温のようだ。  
今年は桜の開花が早いと聞く。  
今から春の訪れが待ち遠しい。  
</p>
```

```
p {  
  border: 3px solid #0F0;  
}
```

線幅

線のスタイル

色

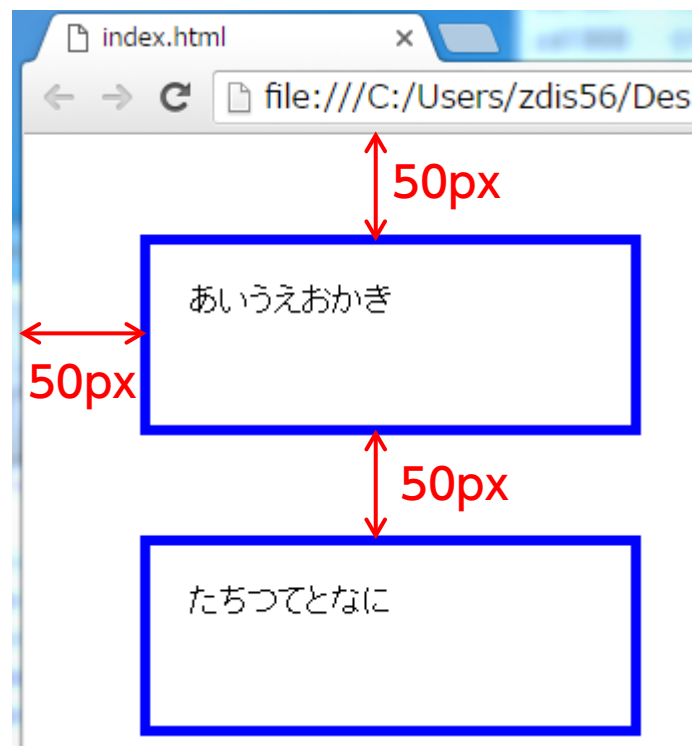


# ボックスの間隔 margin

- 隣接するボックスからの距離を指定する
- pxや%などでの指定が可能

```
<p>あいうえおかき</p>  
<p>たちつてとなに</p>
```

```
p {  
  width: 200px;  
  height: 50px;  
  border: 5px solid #00F;  
  padding: 20px;  
  margin: 50px;  
}
```



# margin の個別指定

- パディングと同様の方法で上下左右に個別の指定が可能

① サブプロパティに**top, bottom, left, right** を付ける

```
p {  
  margin-top: 20px;  
  margin-bottom: 20px;  
  margin-left: 10px;  
  margin-right: 0px;  
}
```

② 半角アキで複数の値を入れる

指定数	例	意味
1	margin: 5px;	上下左右すべて 5px
2	margin: 10px 5px;	上下は10px、左右は5px
3	margin: 10px 0px 20px;	上は10px、左右は0px、下は20px
4	margin: 10px 0px 10px 5px;	上は10px、右は0px、下は10px、左は5px 時計回り



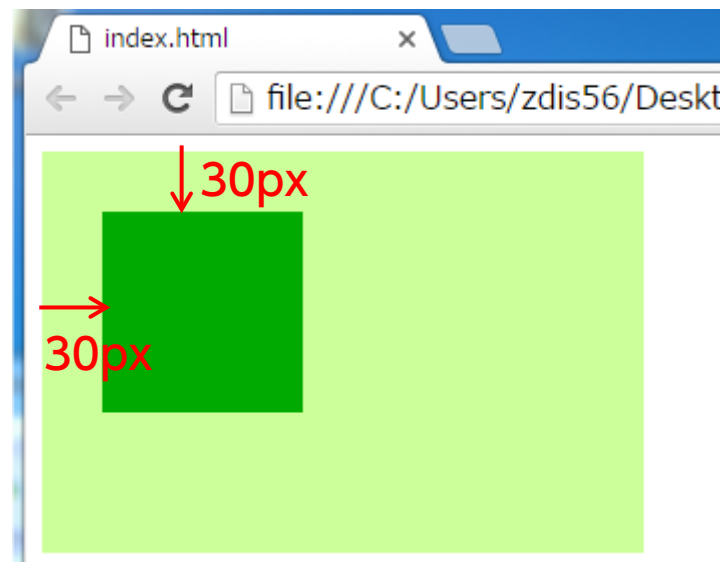
# padding と margin

- 内側の余白は、親ボックスの padding と子ボックスの margin の2通りの選択肢がある

## paddingで設定する場合

```
.parent {  
  background: #CF9;  
  box-sizing: border-box;  
  width: 300px;  
  height: 200px;  
  padding: 30px;  
}  
.child {  
  background: #0A0;  
  width: 100px;  
  height: 100px;  
}
```

```
<div class="parent">  
  <div class="child"></div>  
</div>
```



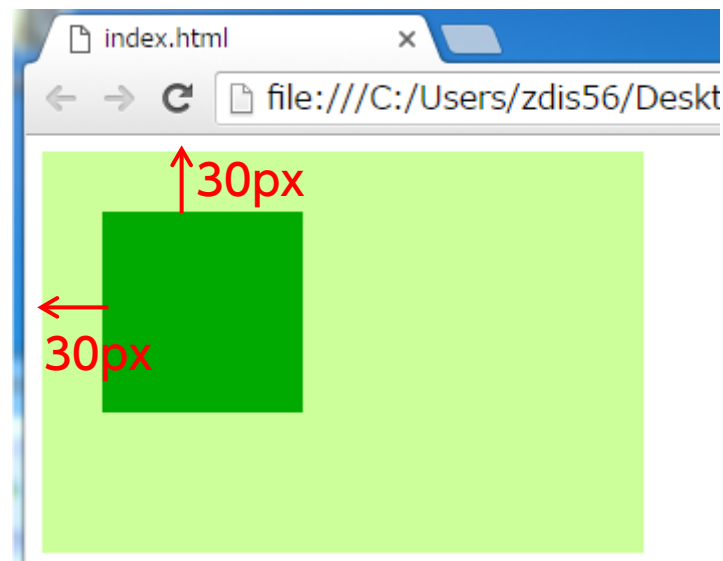
# padding と margin

## marginで設定する場合

```
.parent {  
  background: #CF9;  
  overflow: hidden;  
  width: 300px;  
  height: 200px;  
}  
.child {  
  background: #0A0;  
  width: 100px;  
  height: 100px;  
  margin: 30px;  
}
```

overflow:hidden を入れないと、上のマージンが効かない（または、.parent内にテキストを入れる必要がある）

```
<div class="parent">  
  <div class="child"></div>  
</div>
```



どちらでも同じ結果が得られるが、内側の余白という意味ではpaddingの方がわかりやすい

# 練習問題

---

- 練習02-1 を行う

# ボックスモデルの特徴

# ボックスモデルの特徴

---

- 以下のボックスモデルに関する特徴について見ていく
  - ボックスのサイズ
  - 背景画像の表示領域
  - マージンの相殺と加算
  - blockとinline

# ボックスのサイズ

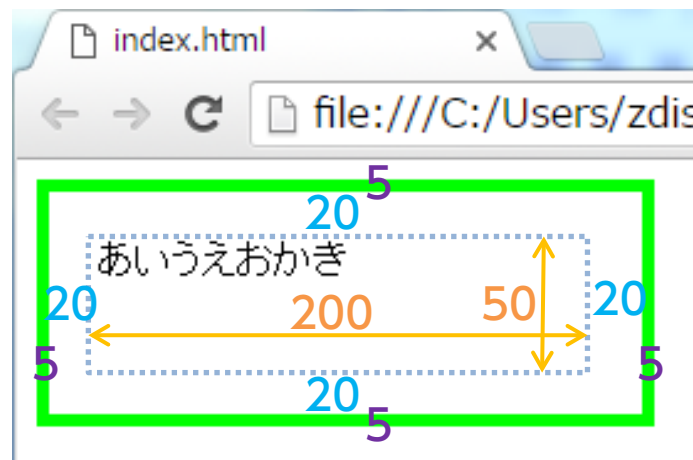
- レイアウトはボックスを配置する作業  
⇒ ボックスのサイズを把握する必要がある

<p>あいうえおかき</p>

```
p {  
  width: 200px;  
  height: 50px;  
  padding: 20px;  
  border: 5px solid #0F0;  
}
```

ボックスの幅 = 250px  
(200 + 20 + 20 + 5 + 5)

ボックスの高さ = 90px  
(50 + 20 + 20 + 5 + 5)



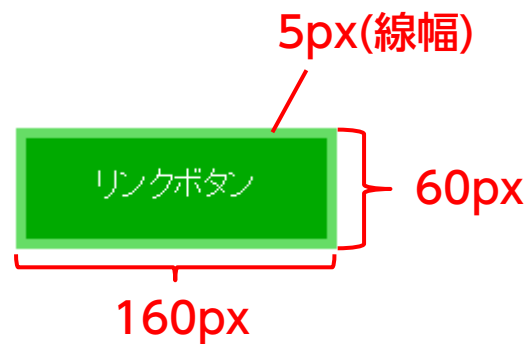
widthとheightを見ると 200 × 50だが、  
ボックスとしてのサイズは 250 × 90 になる

# ボックスのサイズ調整

- 必要に応じて、width / height からパディングやボーダーの大きさを差し引く

リンクボタンを作る場合の例

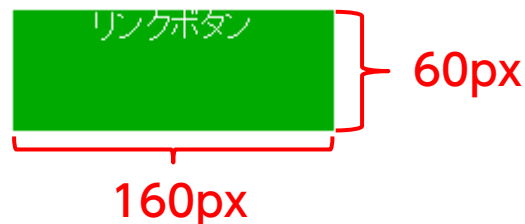
① 作るものを決める



② widthとheightを設定

```
<a href="">リンクボタン</a>
```

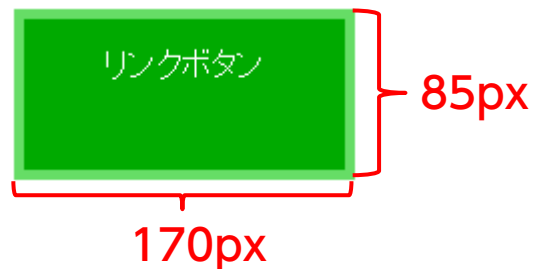
```
a {  
  color: #FFF;  
  text-decoration: none;  
  background: #0A0;  
  display: block;  
  text-align: center;  
  width: 160px;  
  height: 60px;  
}
```



# ボックスのサイズ調整

## ③ パディングとボーダーを設定

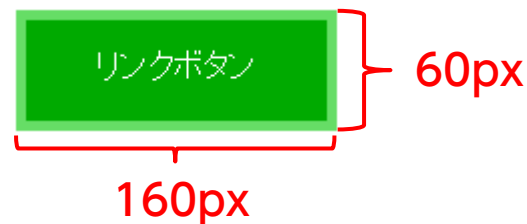
```
a {  
  ... 省略 ...  
  width: 160px;  
  height: 60px;  
  border: 5px solid #6D6;  
  padding-top: 15px;  
}
```



パディングとボーダーの分、  
サイズが大きくなる

## ④ パディングとボーダーで増えた分をwidthとheightから差し引く

```
a {  
  ... 省略 ...  
  width: 150px;  
  height: 35px;  
  border: 5px solid #6D6;  
  padding-top: 15px;  
}
```





# ボックスのサイズ (box-sizing)

- **box-sizing** プロパティを利用することで、width と height が、そのままボックスの幅と高さになる

```
<p>あいうえおかき</p>
```

```
p {  
  width: 200px;  
  height: 100px;  
  padding: 20px;  
  border: 5px solid #0F0;  
  box-sizing: border-box;  
}
```

初期値は content-box



# 背景の表示領域

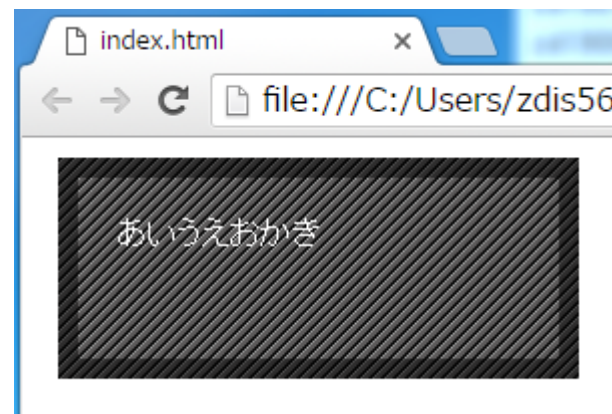
- 背景色や背景画像はボーダー一部分まで表示される

<p>あいうえおかき</p>

```
p {  
  color: white;  
  width: 200px;  
  height: 50px;  
  border: 10px solid rgba(0,0,0,0.5);  
  padding: 20px;  
  margin: 10px;  
  background: url("../images/line.gif");  
}
```



line.gif

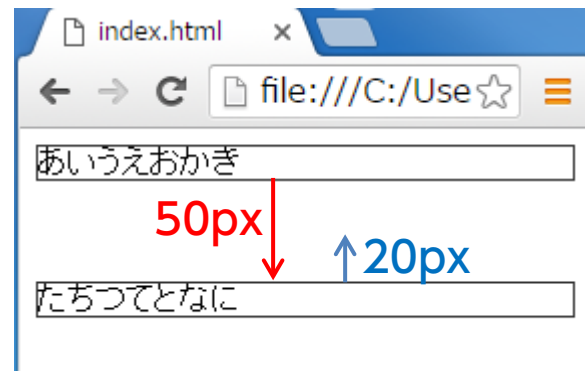


# マージンの相殺（上下）

- 上下でマージンが重なった場合、大きい値が優先される

```
<p class="p1">あいうえおかき</p>  
<p class="p2">たちつてとなに</p>
```

```
p {  
  border: 1px solid #333;  
}  
.p1 {  
  margin-bottom: 50px;  
}  
.p2 {  
  margin-top: 20px;  
}
```

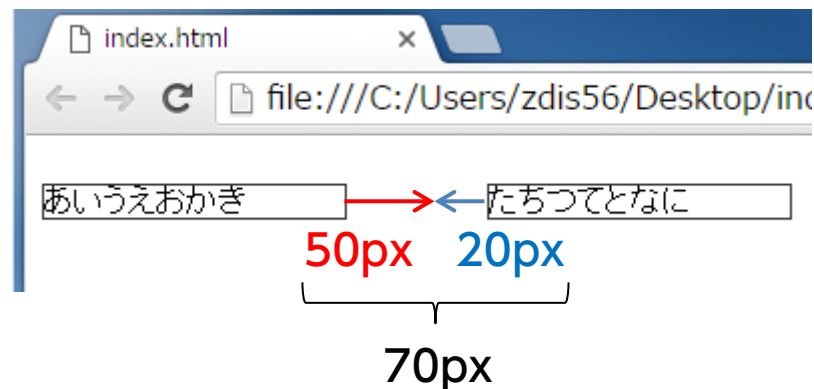


# マージンの加算（左右）

- 左右でマージンが重なった場合、加算される

```
<p class="p1">あいうえおかき</p>  
<p class="p2">たちつてとなに</p>
```

```
p {  
  border: 1px solid #333;  
  width: 150px;  
  float: left;  
}  
.p1 {  
  margin-right: 50px;  
}  
.p2 {  
  margin-left: 20px;  
}
```



# displayプロパティ

---

- ボックスの表示に関するプロパティで、以下のような値を指定する

値	説明
<b>block</b>	一つのまとまったブロックとして扱われ、一般的なブラウザでは前後に改行を伴って表示される。幅や高さの設定が可能。 div, section, h1~h6, p など文書の構造を示す要素の初期値であることが多い
<b>inline</b>	文章の一部として扱われ、前後に改行を伴わない。幅や高さの設定は不可。 a, span などフレージングコンテンツの初期値であることが多い
inline-block	inlineと同じような表示形式だが、幅や高さの設定が可能。 img要素の初期値
none	要素を表示しない。また、その領域も残らないという点で、 visibility: hidden; とは異なる

# displayプロパティとボックス

- displayプロパティによってボックスの性質は異なる

	block	inline
height（高さ）	指定可能	指定不可
width（幅）	指定可能	指定不可
padding（上下）	指定可能	指定可能 ※ボーダーの位置と背景色に影響が出るが、レイアウトには変化なし
padding（左右）	指定可能	指定可能
border（上下）	指定可能	指定可能 ※行ごとの上下につく
border（左右）	指定可能	指定可能 ※最初の行の左と最後の行の右につく
margin（上下）	指定可能	指定不可
margin（左右）	指定可能	指定可能

# 練習問題

---

- 練習02-2 を行う