

HTML初心者講座

HTML for Beginners

Yusuke Miyazaki

この講座について

- ◆ 対象
 - ◆ HTMLを書いたことの無い人でも
- ◆ 目標
 - ◆ HTML文書の構造を理解する
 - ◆ 簡単なHTML文書を作成できるようになる

HTMLとは

- Hypertext Markup Languageの略
- Webで利用されるマークアップ言語
- 文書の構造を記述するために用いる
- ここでは基本的にHTML5について記述

HTMLの例

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="utf-8">
    <title>Simple HTTP Page Title</title>
  </head>
  <body>
    <h1>Simple HTTP Page</h1>
  </body>
</html>
```


HTMLの歴史

- 1990年にCERNで開発された
- 1993年にHTML 1.0のドラフト
- 1999年にHTML4.01がW3C勧告
- 現在HTML5は勧告候補

HTML/XML/XHTML

- ◆ XML (Extensible Markup Language)
 - ◆ 拡張可能なマークアップ言語
- ◆ XHTML
 - ◆ HTMLをXMLで再定義したもの

HTML/CSS/JavaScript

- ◆ CSS (Cascading Style Sheets)
 - ◆ 色, フォント等のスタイルを記述
 - ◆ 文書の構造と装飾の分離のために
- ◆ JavaScript (ECMAScript)
 - ◆ プログラミング言語

HTMLの例

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="utf-8">
    <title>Simple HTTP Page Title</title>
  </head>
  <body>
    <h1>Simple HTTP Page</h1>
  </body>
</html>
```


HTMLの例

Simple HTTP Page

HTMLの構造

- ◆ DOCTYPE
 - ◆ 1行目に記述する
- ◆ 要素 - elements
 - ◆ html要素をルートとする, 入れ子構造になった要素

DOCTYPE

- 必須のpreamble
- 指定しない場合はブラウザが適切な描画モードで表示しないことある
- これを適切に指定することで, 多くのブラウザで仕様に沿った表示が出来る

DOCTYPEの例

- ◆ HTML5
 - ◆ `<!DOCTYPE html>`
- ◆ HTML 4.01
 - ◆ `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">`

要素 - element

```
<html lang="ja">~~~</html>
```

html 要素 (element)

- 黄色部分全体が要素
- 要素 ≠ タグ

開始/終了タグ - tags

<html lang="ja">~~~~</html>

開始タグ

終了タグ

- 要素は開始タグで始まり終了タグで終わる
- タグはタグ名や属性を<>で囲んだもの
- 終了タグは</開始タグ名>

空要素の開始タグ - tags

```
<meta charset="utf-8">
```

開始タグ

- 空要素は開始タグのみで, 終了タグはない
- 開始タグは以下のように書いても良い
 - `<meta charset="utf-8" />`

属性 - attributes

```
<html lang="ja">~~~</html>
```

lang 属性

- 属性は属性名=属性値
- 属性値を省略すると空文字列を指定したのと同じ

コンテンツ - contents

<html>~~~~~</html>

コンテンツ

- コンテンツには他の要素やテキストが入る
- その他にも制約あり
- 空要素はコンテンツを持たない

要素 - elements

- ◆ 様々な要素が定義されている
 - ◆ 例: html, head, body, div, a etc...
- ◆ 要素毎に記述できる場所, 設定できる属性等が定義されている
- ◆ 定義に従ってHTMLを記述していく

HTMLの例

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="utf-8">
    <title>Simple HTTP Page Title</title>
  </head>
  <body>
    <h1>Simple HTTP Page</h1>
  </body>
</html>
```


html 要素

- ルートとなる要素
- head要素とbody要素を順にコンテンツに持つ

head 要素

- html 要素の最初の要素
- メタデータコンテンツを記述
- メタデータコンテンツ
 - 表示の仕方
 - 他のドキュメントとの関係など

title 要素

- head 要素に1つだけ記述
- コンテンツにページのタイトルを設定
- ブラウザではタブやウィンドウに表示される

meta 要素

- title 要素などでは記述できないメタデータを記述
- 文字コード, 説明, キーワードなど
- 空要素

body 要素

- html 要素の2番目の要素
- 文書のコンテンツを記述

h1, h2, … h6 要素

- 見出しを表す要素
- h1が最高位で, h6が最低位

HTMLの例

```
<!DOCTYPE html>  
<html lang="ja">  
  <head>  
    <meta charset="utf-8">  
    <title>Simple HTTP Page Title</title>  
  </head>  
  <body>  
    <h1>Simple HTTP Page</h1>  
  </body>  
</html>
```


もう少し複雑なHTMLの例

```
<!DOCTYPE html>
<html lang="ja">
  <head>
    <meta charset="utf-8">
    <title>楠 太郎</title>
  </head>
  <body>
    <h1>楠 太郎</h1>
    <h2>自己紹介</h2>
    <p>Consectetur iusto magnam dignissimos quaerat aspernatur, eos quisquam. </p>
    <h2>好きな言語</h2>
    <ul>
      <li>Python</li>
      <li>C++</li>
      <li>Scheme</li>
    </ul>
    <h2>リンク</h2>
    <ul>
      <li><a href="https://twitter.com/CamphorKyoto">Twitter @CamphorKyoto</a></li>
      <li><a href="https://www.facebook.com/Camphorcamphor">Facebook Page</a></li>
    </ul>
  </body>
</html>
```


もう少し複雑なHTMLの例

楠 太郎

自己紹介

Consectetur iusto magnam dignissimos quaerat aspernatur, eos quisquam. Voluptates quam quas beatae eum odio nesciunt architecto vitae nulla et? Qui placeat veritatis optio sit nihil sed odio magnam aperiam minus.

Dolor eaque nulla ullam aliquid iusto. Fuga quae ipsum facere iste odio eos accusantium quibusdam suscipit. Voluptas necessitatibus itaque accusantium reprehenderit earum. Facilis dolorum possimus aspernatur necessitatibus quas. Ducimus sunt!

好きな言語

- Python
- C++
- Scheme

リンク

- [Twitter @CamphorKyoto](#)
- [Facebook Page](#)

p 要素

- ◆ 段落を表す要素

ul / li 要素

- ◆ ul
 - ◆ 順序が重要でないリストを表す要素
 - ◆ olは順序に意味があるリストを表す要素
- ◆ li
 - ◆ リストの項目を表す要素

a 要素

- href 属性を持つ場合
 - ハイパーリンクを表す要素
 - href属性に遷移先のURL
- href 属性を持たない場合
 - プレースホルダ

構造と装飾の分離 - CSS

- これまでHTMLで作成した構造をCSSで装飾して表示してみる
- ここではBootstrapのCSSを利用してみます
- head 要素に以下の行を追加する
- `<link rel="stylesheet" href="http://netdna.bootstrapcdn.com/bootstrap/3.0.1/css/bootstrap.min.css">`

構造と装飾の分離 - CSS

楠 太郎

自己紹介

Consectetur iusto magnam dignissimos quaerat aspernatur, eos quisquam. Voluptates quam quas beatae eum odio nesciunt architecto vitae nulla et? Qui placeat veritatis optio sit nihil sed odio magnam aperiam minus.

Dolor eaque nulla ullam aliquid iusto. Fuga quae ipsum facere iste odio eos accusantium quibusdam suscipit. Voluptas necessitatibus itaque accusantium reprehenderit earum. Facilis dolorum possimus aspernatur necessitatibus quas. Ducimus sunt!

好きな言語

- Python
- C++
- Scheme

リンク

- [Twitter @CamphorKyoto](#)
- [Facebook Page](#)

構造と装飾の分離 - CSS

- HTMLに少し修正を加えると…

構造と装飾の分離 - CSS

楠 太郎

自己紹介

Consectetur iusto magnam dignissimos
quaerat aspernatur, eos quisquam.
Voluptates quam quas beatae eum odio
nesciunt architecto vitae nulla et? Qui placeat
veritatis optio sit nihil sed odio magnam
aperiam minus.

Dolor eaque nulla ullam aliquid iusto. Fuga
quae ipsum facere iste odio eos accusantium
quibusdam suscipit. Voluptas necessitatibus
itaque accusantium reprehenderit earum.
Facilis dolorum possimus aspernatur
necessitatibus quas. Ducimus sunt!

好きな言語

- Python
- C++
- Scheme

リンク

- [Twitter @CamphorKyoto](#)
- [Facebook Page](#)

まとめ

- HTMLは文書の構造を記述する言語
- 文書の構造と装飾は分離する
- きちんと仕様に従って記述する

資料

- ◆ W3CによるHTML5勧告候補
 - ◆ <http://www.w3.org/TR/html5/>
- ◆ W3C Markup Validation Service
 - ◆ <http://validator.w3.org/>