

情報基礎 Webページ作成(後半)

HTML&CSSの経緯

- ◆1990頃 Web登場
 - ◆HTMLはWebページの論理構造を記述するもので、レイアウトを記述するものではない
 - ◆レイアウトを記述するためのスタイルシート言語は規定されなかった
- ◆1993頃以降 ブラウザ普及
 - ◆サイト製作者は見た目をよくしたい
 - ◆ブラウザにあわせてレイアウト記述
 - ◆例)文字を大きくしたいときに<h3>を使う
 - ◆ブラウザによって異なる独自要素も登場 例)
 - ◆HTMLの混乱と複雑化

HTML&CSSの経緯

- ◆1996 レイアウトを記述する枠組として規定(CSS1)
- ◆1997 HTML3.2:折衷的なHTML
 - ◆暫定的措置としてレイアウトに関する独自要素の取り入れ
- ◆1997 HTML4.0:論理構造とレイアウトの分離
- ◆[1998 CSS2](#)
- ◆[1999 HTML 4.01](#)
- ◆2000 XHTML 1.0
- ◆2001 XHTML 1.1

HTML&CSSの経緯

- ◆[2011 CSS 2.1](#)
- ◆[2011/2012 CSS3](#)
- ◆[2014 HTML5:元はWHATWGによる](#)
- ◆[2016 HTML 5.01](#)
- ◆[2017 HTML 5.02](#)

CSS利用のメリット

- ◆詳細なレイアウトを記述できる
 - ◆HTMLよりレイアウト記述力が高い
 - ◆例)テキストや画像を好きな位置に配置できる
- ◆HTMLから、レイアウトに関する記述を除去できる
 - ◆HTMLがシンプルに
 - ◆文法間違いを減らせる→情報を正しく伝達できる、表示の間違いを減らせる、維持管理が楽に
 - ◆アクセシビリティ向上
 - ◆例)HTMLのtable要素を使ってレイアウトすると、音声読み上げソフトを利用してWebページを聞く視覚障害者に正しく情報が伝わりにくい
 - ◆検索エンジン対策

CSS利用のメリット

- ◆レイアウトに関する記述を外部CSSファイルにできる
 - ◆サイトの維持管理が楽に(後述)
- ◆ユーザが自分の好きなスタイルシートでページを見ることができる
- ◆正しい(推奨される)考え方

CSSのメリットの例

- ◆「New」という文字列の色を赤から黄色にかえたい場合
- ◆100ページ(100個のHTMLファイル)のWebサイト

HTML+CSSで記述

◆HTML

```
◆<span class="attention">New</span>
```

“attention (名前は自由)” というclassを設定
100個のHTMLファイルは変更不要

◆CSS(別ファイル)

◆.attention

```
◆color: yellow;
```

1個のCSSファイルにおいて “attention”の色を
redからyellowに変更

課題

- ◆1. cssファイルをcssディレクトリにまとめる
- ◆2. 画像ファイルをimageディレクトリにまとめる
- ◆3. 3つ目のHTMLファイルを作成
- ◆4. W3C HTML ValidatorでHTML文法チェック, W3C CSS ValidatorでCSS文法チェック(トップページだけでもよい)
- ◆5. サイト全体を二つ以上のブラウザで確認(スマホでも確認)

課題

- ◆6. 学外公開

課題(余裕のある人)

- ◆7. CSS3の要素の利用(最初からCSS3でもよい)
- ◆8. javascriptの利用

HTMLとCSSの組合せ(参考)

- ◆HTMLとCSSの組合せは自由
- ◆HTML5 + CSS2(現在)
- ◆HTML5 + CSS3(余裕のある人へ)

課題(後半)の目標

- ◆ディレクトリの整理
- ◆文法チェックに合格(HTML/CSS)
- ◆Webサイトの完成
- ◆外部公開

一般的なディレクトリ構成

- ◆public_html(ディレクトリ)
 - ◆css(ディレクトリ)
 - ◆style.css(名前は自由)
 - ◆image(ディレクトリ)
 - ◆index.html
 - ◆他のHTMLファイル(あるいはフォルダ)

ディレクトリの整理

- ◆1. CSSファイルをcssディレクトリにまとめる
 - ◆2.1 public_htmlの下にcssディレクトリを作成
 - ◆2.2 CSSファイルをcssディレクトリに移動
 - ◆2.3 HTMLのCSSファイルへのリンクのパスを変更して保存
 - ◆例) <link rel="stylesheet" href="style.css">を<link rel="stylesheet" href="css/style.css">に

ディレクトリの整理

- ◆2. 画像ファイルをimageディレクトリにまとめる
 - ◆5.1 public_htmlの下にimageディレクトリを作成
 - ◆5.2 画像ファイル(.jpg, .pngなど)をimageディレクトリに移動
 - ◆5.3 HTMLとCSSファイルの中の画像へのパスを変更して保存
 - ◆例) をに

Webにおける絶対パスと相対パス

- ◆絶対パス
 - ◆ファイルのURL
 - ◆例) https://www.ecs.osaka-cu.ac.jp/~学籍番号/
 - ◆例) https://www.ecs.osaka-cu.ac.jp/~学籍番号/index.html
 - ◆例) https://www.ecs.osaka-cu.ac.jp/~学籍番号/profile.html
 - ◆例) https://www.ecs.osaka-cu.ac.jp/~学籍番号/css/style.css
 - ◆例) https://www.ecs.osaka-cu.ac.jp/~学籍番号/image/photo.gif

Webにおける絶対パスと相対パス

- ◆相対パス
 - ◆自分の位置から見た相手の位置
 - ◆自分「.」、自分の上「..」(ただし「./」は省略可能)
 - ◆例) index.htmlからprofile.htmlを見る場合:
 - ◆./profile.html または profile.html
 - ◆例) index.htmlからstyle.cssを見る場合:
 - ◆./css/style.css または css/style.css
 - ◆例) style.cssからphoto.gifを見る場合:
 - ◆../image/photo.gif

他の見本

- ◆HTML5+CSS3
 - ◆<https://mihon2/>以下
 - ◆index.html
 - ◆css/css3.css
 - ◆images/photo-1000-01.jpg, photo-1000-02.jpg
 - ◆scrollsmoothly.js
- ◆コピーして使用してもよい

CSS利用の問題

- ◆HTMLより仕様が複雑
- ◆ユーザがスタイルシートを利用しないことがある
- ◆ブラウザによってCSSの対応が異なる。ブラウザのバグもある
 - ◆CSSの文法通りに記述しても正しく表示されないことがある。ブラウザの対応度合いの差はHTMLのより大きい
- ◆複数のブラウザで美しく正しく表示させるためには、多くの手間と時間とノウハウが必要