

情報デザイン演習I 2.HTML基本

目次

1. 初めに
2. HTMLの歴史
3. HTMLの基礎
 - i. 課題

初めに

- 606に部屋変更しました。
- テキスト持っていない人どのくらいいる？(今日もまだ持っていない人いる想定だけど)

教科書

必ず購入してください。

- 1冊ですべて身につくHTML & CSSとWebデザイン入門講座

HPの置き場所(再掲)

<https://sammyppr.github.io/>

に置きます。これは学外からも閲覧可能です。予習復習にはもちろん、あれなんだっけー？
みたいな時に利用してください。

HTMLの歴史

HTMLとは

- HTML: HyperText Markup Language

HyperText?

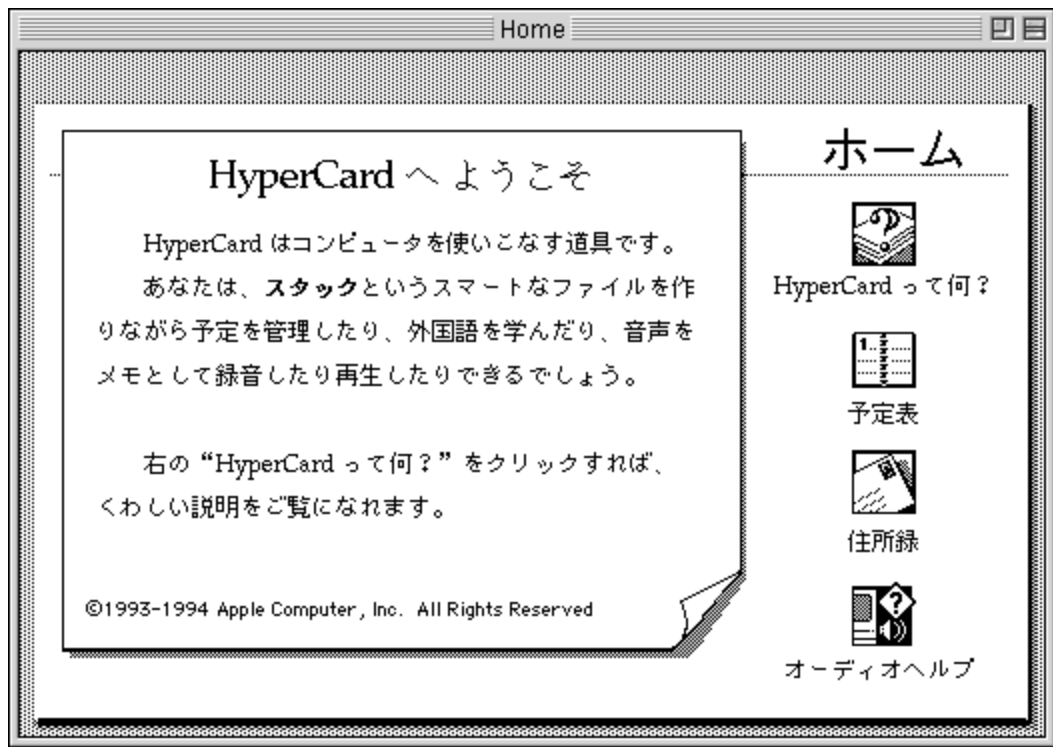
今では皆さん当たり前

- 検索して
- リンクを踏んで

情報に到達すると思いますが、この「リンク」の歴史、そんなに古くないんです。

HyperCard

ハイパーテキストを実現した最初の商用ソフトウェアで、1987年にできました。クリックすることで、別のページに飛べるというものでした。



HTMLの誕生

HTMLはスイス・ジュネーブにある欧州原子核研究機構（通称：CERN）においてイギリスの計算機科学者であるティム・バーナーズ・リー氏を中心に1989年に開発されました。

CERNには膨大な数の論文が存在していたため、それを探す手間をなくすために論文をHTMLで記載し、リンクで検索しやすくする、という方法が取られました。

(最近ではAIで検索できてしまいますが...)

ポイント

- HTMLは文章構造を担う言語である
- HTMLはもともと非公開の中、ある研究機関で使われていた
- WWWの生みの親ティム・バーナーズ・リーはHTMLも開発していた
- 最新のHTML5では動的な動きが可能になる
- HTMLがもたらした最大の功績はリンクによるページ移動である

引用:世界中の情報整理を後押ししたHTMLの進歩

デザインはCSSの役割となります。

HyperText Markup Language

で、結局のところ文書に対して

- ここは見出し
- ここは段落
- ここは箇条書き
- ここはリンク

意味づけすることでHTML文書は記載されます。

現在のバージョン

HTML5が利用されています。

...ちょっとややこしいのですが、「HTML5は廃止された標準仕様。」 ???

正しくは「HTML Living Standard」が現在のバージョンとなります(2019から)。

書籍などではHTML5という記載方法で本がまだ売られています。厳密に言えば「HTML5」

「HTML Living standard」は別物ですが、かなり似ています。ですので、極めない限りはそんなに差は感じないでしょう。

参考：[どうしてHTML5が廃止されたのか](#)

HTML アイコン

HTML5は廃止されているにも関わらず、HTML Living Standardのアイコンは存在しておらず、HTML5のアイコンが引き続き利用されている、という謎の状態が現状となっています。

- [W3CがHTML5のロゴを発表！ ホームページに貼ろう！](#)

今後どうなるの？

HTML 6が登場することはありません。

HTML Living Standardはバージョンの概念を持たず、内容が日々改訂されていき、改訂されたその時点からその内容が標準となるため、これまでW3Cの勧告にあわせて仕様を確認してきた以上に、今後はHTML Living Standardの仕様を随時チェックしていく必要があります。

引用：HTML標準規格はHTML Living Standard！HTML5は2021年1月に廃止

HTML Living Standard

[英語のサイト](#)

[日本語版](#)

全ての仕様がここに書かれていますが、多く使うものは限られています。
多すぎて無理——、と思わないようにしましょう。

HTMLの基礎

の前に

どのような順番で学んでいくか説明します。

- HTML基礎
- CSS
- レイアウト
- レスポンシブ

最低限のHTML

```
<!DOCTYPE html>
<html>
<head>
  <!-- 宣言部分 -->
</head>
<body>
  <!-- 本文部分 -->
</body>
</html>
```

兎にも角にも、この最低限のHTMLは必ず覚えましょう。

HTMLタグ(再掲)

- `<xxxx>`を開始タグ
- `</xxxx>`を終了タグ

と呼びます。

全てのタグが開始タグ・終了タグから成り立っているわけではなく、終了タグを書かないものもあります。

行頭のスペース(再掲)

インデントと呼びます。

これにより、タグの始まりと終わりの関係が分かりやすくなります。

TABキーで入力できます。

それぞれの意味(再掲)

- `<!DOCTYPE html>`: HTML5で書かれていることを指します
- `html`: この中に必ず`head`と`body`が入ります
- `title`: ウィンドウのタイトルに表示される文字です。本文ではないため、`head`内に書きます。
- `body`: この中に本文が書かれます。

確認：インストール

- [Visual Studio Code\(VSC\)](#){:target="_blank"}
- [Chrome](#){:target="_blank"}

入ってますね？VSCodeに

- Japanese Language Pack
- LiveServer
- Zenkaku

入ってますね？

やってみよう

「ID_root」までは前回の再掲です。

- 書類フォルダ等に、「ID_root」というフォルダを作成
- VSCode(Visual Studio Codeのことを以下こう呼びます)起動
- 「フォルダーを開く」で、「ID_root」を指定
- 新規フォルダを作成して「ID_02」
- 新規ファイルを作成して、「test.html」とファイルを作成

起動は「Command+スペース」でvsc,またはcodeでできるかと思います。

「.html」とすることで、VSCodeは「あ、HTMLファイルを作りたいんだ」と認識して、便利な機能を提供してくれます。

VSCodeの注意！

フォルダの指定

作業するときに必ず「フォルダを開く」を利用して、どこのフォルダで作業するかを明示すること！

LiveServerを利用するときなど、**どこを基準に作業しているか**が非常に大事になってきます。

拡張子を間違えない

VSCは拡張子で何を書こうとしているかを認識します。間違えないように。

ファイル名について - 使っていい文字

- 半角英数の小文字
- 記号の - (ハイフン)
- 記号 _ (アンダースコア)

ファイル名について - 使ってはいけない文字

- 全角文字（日本語、英語）
- 半角・全角スペース
- ハイフンとアンダースコア以外の記号
- 機種依存文字

ファイル名について - 大文字

使ってもどうにかなる場合がありますが、たまに大文字と小文字を混乱するシステムがあります。

そのため、大文字の利用は避けましょう。

先に手本を見せるので、同じようにやってみよう。

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>HTMLの基礎</title>
</head>
<body>
  <h1>見出し1</h1>
  <h2>見出し2</h2>
  <h3>見出し3</h3>
  <h4>見出し4</h4>
  <h5>見出し5</h5>
  <h6>見出し6</h6>
  <p>段落</p>
</body>
</html>
```

- h1: heading1...
- p: paragraph

Live Server

実際にブラウザで閲覧するには、右下のGo Liveをクリックだったね。

Live Serverをインストールしているので、再読み込みをしなくても、ソースを保存するとアップデートされるよ。(調子良ければ、保存しなくても)

保存しないと、データ失われるので気をつけましょう。

metaタグ

metaタグとは、Webページの記述に用いるHTMLで定義された要素（タグ）の一つで、そのページについての様々な情報（メタ情報）を記述するためのもの。ページ先頭のヘッド領域（headタグで囲まれた部分）に記載され、主に他の要素で定義されていない種類の情報を記述するのに用いられる。

引用: metaタグ（メタタグ）とは - IT用語辞典 e-Words

charset

テキストファイルには、「文字コード」と呼ばれるデータの保存方法がいくつかあります。
有名なところでは

- Shift-JIS
- EUC
- UTF-8

がありますが、UTF-8が一般的になりつつあります。

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

は、「この文書をUTF-8という文字コードで記述しているよ」という意味になり、おまじないだと思ってOKです。

この後、追加する場所について

この後は、`</body>`の前に追加していこう。

見やすいように、インデントにも気をつけよう。

リストを作成してみよう

```
<ul>
  <li>リスト1</li>
  <li>リスト2</li>
  <li>リスト3</li>
  <li>リスト4</li>
</ul>
<ol>
  <li>リスト1</li>
  <li>リスト2</li>
  <li>リスト3</li>
  <li>リスト4</li>
</ol>
```

- ul: unordered list(順序なしリスト)
- ol: ordered list(順序ありリスト)
- li: list item(リストアイテム)

説明リスト

```
<dl>
  <dt>用語1</dt>
  <dd>説明文1</dd>
  <dt>用語2</dt>
  <dd>説明文2</dd>
</dl>
```

- dl: description list(説明リスト)
- dt: description term(説明する言葉)
- dd: definition / description(定義分もしくは説明文)

※古くはdefinition listだったのが気づいたら変わってた...

画像貼ってみよう

```
<img src="">
```

- まずこのように書いて
- 適当にweb上の画像を選んで
- 右クリックで「画像アドレスをコピー」
- 下のようにsrc=""の中に貼り付け

```

```

alt

```

```

の様に画像を貼る時にはaltをつけることが推奨されています。

役割としては

- 目の見えない人に何の画像が説明する
- 画像が壊れていた時に、説明を表示する

があります。

リンクを使ってみよう

```
<a href="https://www.google.co.jp/">Google</a>
```

- a: anchor(錨(いかり))
- href: Hypertext Reference

```
<a href="https://www.google.co.jp/" target="_blank">Google</a>
```

とすると、別ウィンドウが開く

画像にリンクを貼ってみよう

```
<a href="https://hello-world-movie.com/" target="_blank">  
    
</a>
```

Webで利用できる画像について

- 種類
 - 昔から使える: jpg, gif
 - 現在標準的: png
 - 新しいこれから標準になる: webp, avif
 - ベクター画像用: svg
- 解像度
 - 大きすぎると容量が重くなるのでwebには不適切
 - 必要最低限の解像度に抑えよう

webで利用できない画像

- heic:iPhoneの最近の高効率な圧縮方法の形式
- psd:Photoshopの編集用

画像についてはまた詳しくやります。

あ...

別にこの映画が好きなわけでもなんでもなく、プログラマーはとりあえずなんか作る時にまずは「Hello World」って表示することになっていて、その画像探したら、こんなになりました。

改行

段落でなく改行をしたい時には

```
<br>
```

を使いましょう。終了タグありません。

水平線

区切りたい時には

```
<hr>
```

を使いましょう。終了タグありません。

コメントアウト

HTMLを書いていると、人がわかりやすいようにコメントを残したくなることがあります。

```
<!-- ここにコメントを書く -->
```

スペース忘れずに

もっとあるの？

あります。が、覚えなくていいです。

必要な時にググれば良いだけですから。

チートシート

忘れやすいことをまとめた「チートシート」というものがあります。わかりやすく言えば、カンニングペーパーです。

これは、

- HTML
- CSS

だけでなく、

- Illustrator
- Photoshop

等、さまざまなバージョンが配布されています。知っておくと良いでしょう。

チートシート

- [HTML5 タグ★チートシート](#){:target="_blank"}
- [HTML Living Standard チートシート](#){:target="_blank"}

caniuse.com

HTML, CSS, 画像フォーマット等どのブラウザで何が利用できるかを確認できる

- caniuse.com

があります。非常に便利ですので知っておきましょう。

入れ子

あるモノの中にそれと同じモノを入れた構造

- Excelのif文
- シーケンスの中のシーケンス(Premiere Pro)
- コンポジションの中のコンポジション(AfterEffects)
- HTMLタグの中のHTMLタグ

入れ子という構造も知っておきましょう。

入れ子の例...ul

```
<ul>
  <li>項目1</li>
  <li>項目2
    <ul>
      <li>サブ項目1</li>
      <li>サブ項目2</li>
    </ul>
  </li>
</ul>
```

実は...

ここまでで教科書の

- Chapter2 2-1～2-9,2-14(P.50-68)
終わっちゃってます。
- 2-7 ファイルパス
- 2-10 表
- 2-11,12 フォーム
- 2-13 ブロック要素

がまだですが、簡単かな？

教科書持っていない人用に、AirPlay使って確認してみましょう。

知ってほしいこと

- HTMLファイルには基本となる形がある
- あとはタグをつけて内容を増やしていけばいいだけ

課題

今日学んだタグをなるべく全て使って、なんでもいいのでHTML文書を作成してみましょう。

ファイル名は「kadai.html」としましょう。

ググって学んでないタグ使っても構いません。

「なんでもいい」が困る人は、うちの大学のキャンパス・学部・学科・コースなど階層を意識して書いてみましょう。好きなアーティストグループの説明でもいいですよ。

提出先

Student Volume - 2023演習 - 情報デザイン演習II - 02回

に学籍番号フォルダを作成して、ID_02をコピーしてください。