

ウェブページ作成

名古屋大学 情報基盤センター
情報基盤ネットワーク研究部門
嶋田 創

HTML: HyperText Markup Language

- 通常のウェブページはHTMLで記されている
 - URLの最後が.htmlで終わっている(右下図)ことが多いのはそのせい
 - Windowsは拡張子が.htmlもしくは.htmの物をHTMLファイルとして扱う
- HyperTextとは
 - 複数の文書を相互に関連付けたもの
 - 「ハイパーリンク(リンク)」がHyperTextのキモ
- Markup Languageとは
 - 文章中に**タグ**を入れることで書式やリンクを指定する方式(プログラミング言語に近い)
 - 最近だとXML(eXtended Markup Language)なども使われる
 - ブラウザ上で「右クリックメニュー→ページのソースを表示」で、タグを処理する前の文書(ソース: source)を見ることができる

HTMLのタグの基本規則

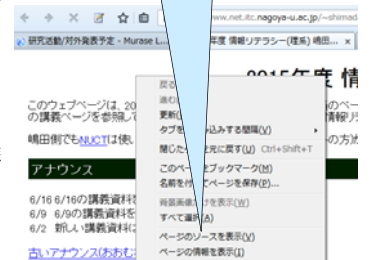
HTMLでは「<>」に囲まれた部分がタグとなる

- 「<>」内にはタグの内容を示すキーワードが最初に来る
 - 例: <body>
- 「<...>」でマークアップ対象範囲を開始し、「</...>」で終了する
 - 例: <body>...</body>
- タグの内容を示すキーワードの後、スペースを空けて属性が書かれる場合がある
 - 例: <a href...>, <a name...>
 - 後述するCSSのclassやstyle指定も同様

HTMLのソース(source: 源)の閲覧

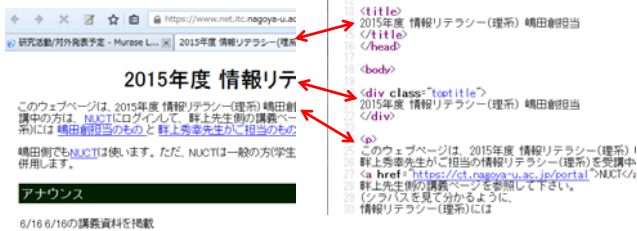
- 普通のブラウザはソースを閲覧するモードがある
 - 右クリックメニュー→ページのソースを表示
- HTMLタグの使い方の勉強を見るのもあり
 - ただし、最近のページは込み入っていて見にくい
 - HTMLを書いた人のお遊びが見れたりすることも

HTMLのソースの表示



ウェブページとHTMLソースの関係

- 冒頭部は細かな設定(文字コードなど)
- タイトル、見出し、最初の段落などの対応が見て取れる

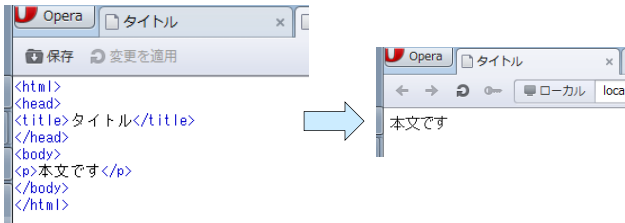


HTMLで使われるタグ(HTMLの概形)

- <html></html>
 - このタグで囲まれた範囲がHTMLで記述されたものとして処理される
 - 通常はHTMLファイルの先頭と最後にある
- <head></head>
 - HTMLのヘッダ部分の範囲を示す
 - ヘッダ部分では、タイトル、使用言語、文字コード、ページ移動、などのブラウザ側に伝えたい情報を記載
 - タイトル(<title>...</title>)はウィンドウバーやタブに記されるため、この分類
- <body></body>
 - ブラウザのウィンドウ部に表示されるコンテンツを記載する部分の範囲を示す

ほぼHTMLの概形だけのページ

- `<html><head><body>`の概形に以下だけを追加
 - `<title>...</title>`: ブラウザのタブやウィンドウバーに表示するタイトル
 - `<p>...</p>`: 段落(Paragraph)
- これぐらいならメモ帳でも書けます
- 文字コード指定のタグが含まれてないので文字化けすることも



文章構造を示すHTMLタグ

- `<a>` (Anchor)
 - ハイパーリンクを設定するためのアンカー
 - 以下の属性と併用することがほとんど
 - href属性 (Hypertext REFerence): ハイパーリンクを設定
 - 例: ``
 - 同一サーバ無いならば、ディレクトリ(フォルダ)の相対位置で指定可能(例: ``)
 - name属性: 文章の途中のアンカーを設定
 - 例: ``
 - 上記は``のような形で参照可能
- `<p>` (Paragraph): 段落
 - 変な解説書だと`<p>`単独で使う説明になっているが、正しくは`<p>...</p>`の形で使う

文章構造を示すHTMLタグ

- `<h>` (Heading): 見出し
 - 数字と同時に使われる(`<h1><h2><h3><h4><h5><h6>`)
 - 数字の小さいほうが上位の見出し
- `` (Unordered List) / `` (Ordered List): (数字付き)箇条書き
 - 範囲内に複数のリスト(``)を含む
 - 例: `1つ目2つ目3つ目`
 - ``を``にすると数字付き箇条書きになる
- `<table>`: 表
 - 範囲内に複数の行(`<tr>` (Table Row))を含む
 - 行の範囲内には複数の列データ(`<td>` (Table Data))を含む
 - タイトル行は`<tr>`ではなく`<th>` (Table Header)を使用

テキストの表現用タグ

- ``: フォントの種類(face)、大きさ(size)、色(color)
 - 例: ``
 - 色指定の名称は「ウェブカラー」で検索
- `` (Bold)
- `<i>` (Italic)
- `<u>` (Underline)
- `<s>` (Strike): 打ち消し線
- `<big>` `<small>`: 文字を一回り大きく(小さく)する
- `<sup>` `<sub>`: 上付き文字、下付き文字

テキストの意味を示すタグ

通常のスタイルでは表現用タグと書式がかぶるが、後述するCSSで書式を指定する時に有用

- ``: 強調
 - 通常のスタイルでは太字になる(``に同じ)
- `` (DElete): 削除
 - 通常のスタイルでは打ち消し線になる(`<s>`に同じ)

その他のよく使うタグ(1/2)

- `
` (BReak): 改行
- `<hr>` (Horizontal Rule): 水平の罫線
- `` (IMaGe): 画像張り込み
 - src属性 (SouRce): 画像ファイルの指定 (例: `src="/circle.png"`)
 - width/height属性: 画像サイズの指定 (例: `width="120" height="60"`)
 - 元画像から拡大縮小も可能
 - 60%などの指定も可能
 - 片方だけ指定すると、もう片方はそれに合わせて自動拡大縮小
 - alt属性 (ALTernative): 画像が表示されてない時に代替文字を表示する(例: `alt="円の画像"`)
 - 例: ``
 - 嶋田のホームページの顔写真部分

その他のよく使うタグ(2/2)

- <table>: 表組
 - <table>~</table>間に<tr>~</tr>タグ(table rowの意味)で行を定義
 - <tr>~</tr>の間に<td>~</td>タグ(table dataの意味)で列を定義
 - <table border>で列ありの表になる

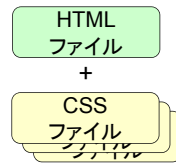
○ 例:

```
<table border>
<tr><td>1行目1列</td><td>1行目2列</td></tr>
<tr><td>2行目1列</td><td>2行目2列</td></tr>
</table>
```



Cascading Style Sheet(CSS)

- 最近では装飾はHTMLタグを用いない
 - 装飾はCascading Style Sheetに任せる
- CSSを変更することでHTMLファイルに手を加えずにデザインを変更可能
 - PowerPointのデザインテンプレートのような使い方
- 複数のCSSを重ねて適用することで再利用性が増す
 - 例: 本文用CSS、表用のCSS、箇条書き用のCSSを重ねる



CSSの有り無しの違いの例

- 講義ページにCSSを適用した場合と適用しない場合

CSS有り

CSS無し

<p>アナウンス</p> <p>6/16 6/16の講義資料を掲載 6/9 6/9の講義資料を掲載 6/2 新入講義資料はありません。 古いアナウンス(お休み明け)</p> <p>教員とティーチング・アシスタント(TA)</p> <table border="1"> <thead> <tr> <th>氏名</th> <th>メールアドレス</th> <th>居室</th> </tr> </thead> <tbody> <tr> <td>教員 嶋田 創(はじめ)</td> <td>shimada (*1)</td> <td>情報基盤センター 5F 504号</td> </tr> <tr> <td>教員 藤原 やなせ(しゅん)</td> <td>yamasawa (*2)</td> <td></td> </tr> <tr> <td>TA 藤上 智史(ふみさとし)</td> <td>fuchigami (*2)</td> <td>10電子情報館 西館 7F 721号</td> </tr> </tbody> </table> <p>(*1) 後ろに@nc.nagoya-u.ac.jpを追加 (*2) 後ろに@net.its.nagoya-u.ac.jpを追加</p> <p>スケジュール</p> <ul style="list-style-type: none"> ※ 1回 工学部員研修 4F デザインセンターA教室 ※ (注) 皆さんの登録状況を見て進行は調整しますので、予定は実際の実施時 	氏名	メールアドレス	居室	教員 嶋田 創(はじめ)	shimada (*1)	情報基盤センター 5F 504号	教員 藤原 やなせ(しゅん)	yamasawa (*2)		TA 藤上 智史(ふみさとし)	fuchigami (*2)	10電子情報館 西館 7F 721号	<p>アナウンス</p> <p>6/16 6/16の講義資料を掲載 6/9 6/9の講義資料を掲載 6/2 新入講義資料はありません。 古いアナウンス(お休み明け)</p> <p>教員とティーチング・アシスタント(TA)</p> <table border="1"> <thead> <tr> <th>氏名</th> <th>メールアドレス</th> <th>居室</th> </tr> </thead> <tbody> <tr> <td>教員 嶋田 創(はじめ)</td> <td>shimada (*1)</td> <td>情報基盤センター 5F 504号</td> </tr> <tr> <td>TA 藤上 智史(ふみさとし)</td> <td>fuchigami (*2)</td> <td>10電子情報館 西館 7F 721号</td> </tr> </tbody> </table> <p>(*1) 後ろに@nc.nagoya-u.ac.jpを追加 (*2) 後ろに@net.its.nagoya-u.ac.jpを追加</p> <p>スケジュール</p> <ul style="list-style-type: none"> ※ 1回 工学部員研修 4F デザインセンターA教室 ※ (注) 皆さんの登録状況を見て進行は調整しますので、予定は実際の実施時 	氏名	メールアドレス	居室	教員 嶋田 創(はじめ)	shimada (*1)	情報基盤センター 5F 504号	TA 藤上 智史(ふみさとし)	fuchigami (*2)	10電子情報館 西館 7F 721号
氏名	メールアドレス	居室																				
教員 嶋田 創(はじめ)	shimada (*1)	情報基盤センター 5F 504号																				
教員 藤原 やなせ(しゅん)	yamasawa (*2)																					
TA 藤上 智史(ふみさとし)	fuchigami (*2)	10電子情報館 西館 7F 721号																				
氏名	メールアドレス	居室																				
教員 嶋田 創(はじめ)	shimada (*1)	情報基盤センター 5F 504号																				
TA 藤上 智史(ふみさとし)	fuchigami (*2)	10電子情報館 西館 7F 721号																				

CSSの適用例

- タグ全体に適用(左下)
- タグのclassに適用(右下)
 - <div class="toptitle">...</div>の区間内のみ適用
- 参考:
 - 下で使われている色の指定は、3原色(Red/Green/Blue)の強さの指定による指定方法です
 - 最初の2桁(16進数)がRed、次の2桁がGreen、次の2桁がBlue
 - 例: #000000が黒、#ffffffが白、#ff0000が赤、#ffff00が黄

```
h2 {
font-family: sans-serif;
background-color: #002000;
color: #ddddd;
padding: 5px
}
```

```
div.toptitle {
text-align: center;
font-size: xx-large;
font-family: sans-serif;
}
```

ウェブオーサリングツールの利用

- いちいちHTMLタグを書くのは面倒
 - 属性まで書いていたらさらに面倒
 - そもそも、うる覚えのものが多いから
 - 特に、CSSは長ったらしいキーワードが多い
- WordのようにWYSIWYGで編集できない?
- 結果を確認するのにブラウザを見るのも面倒



ウェブオーサリングツールの利用
(例: BlueGriffon, Dreamweaver, iWeb,)



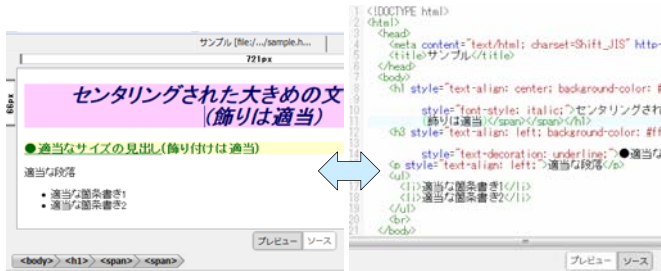
BlueGriffonの特徴(1/2)

- Word的なアイコンで文章構造や文字装飾を挿入可能
 - HTMLの文章構造は事前に指定してから文字を挿入
- CSSベースの装飾やCSSの編集も可能



BlueGriffonの特徴(2/2)

- 直接HTMLタグをいじって細かな修正も可能
 - プレビュー(編集)画面とソース画面のどちらをいじってもOK
 - 例: 下記のタイトルの改行はぱっと挿入する方法が見つからなかったので、
タグを手で挿入

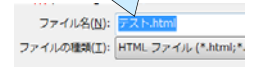


BlueGriffonでHTMLファイル作成

1. まずファイルを名前をつけて保存する
 - <title>部を入力して設定できます(右上図)
 - ファイル名は英数字でつける(右下図)
 - URLに日本語は使えない
2. HTMLファイルを編集して保存する
3. 必要に応じてウェブサーバにアップロードする
 - 普通はFTP(File Transfer Protocol)やSFTP(Secure FTP)で転送するソフトを利用



タイトルと同じ名前がファイル名候補になるが、英数字に直す



細かな操作はWord等と似ているので割愛します

BlueGriffonでのHTML編集のTips(1/2)

- タイトルバーにあるアイコンは基本的にHTMLのタグで設定可能なものしかありません
 - アイコン2行目の文字装飾をしてみるとよく分かる
- 細かな装飾はCSSで設定すること
 - 1つ右のCSS Stylesheet Editor(上級者向け)のアイコンと間違えない



BlueGriffonでのHTML編集のTips(2/2)

- CSSウィンドウでCSSによる細かな装飾が可能
 - CSSおよび適用対象にID/classを設定可能
 - classは複数に適用可能
- フォントはこだわらなければ、sans-serifかserifのフォントファミリーを利用すると良い



HTMLのTips(1/2)

- 無属性のタグはそのままでは何もおきないので、CSS(class)と組み合わせる
 - ブロック単位: <div>...</div>
 - 下記のブロック要素なので、文中で使うと前後で強制的に改段落
 - 文字の部分単位: ...
- HTMLタグの挙動はブロック要素とインライン要素に別れる
 - ブロック要素の物を使うと前後で強制的に改段落が入ったような形になります
 - ブロック要素の例: h, ul, table, div, ...
 - インライン要素の例: img, span, font, ...

HTMLのTips(2/2)

- CGI(Common Gateway Interface)という仕組みを使い、ウェブサーバ側でプログラムを動かすことができる
 - Light Weight Languageと呼ばれるシステムの言語(Perl, PHP, Rubyなど)がよく使われる
- ブラウザ内で動作するJavaScriptというプログラミング言語がある
 - 入力内容(キー入力、マウス操作)によって表示内容が変化したりするのがたいていコソレ
 - Ajax(Asynchronous JavaScript + XML)方式は画面遷移せずにデータ通信も可能
- 他にも、アプレットとかプラグインと呼ばれる方式でブラウザ側でプログラムを実行可能
 - JavaアプレットやFlashプラグインとか
 - HTML5でHTML+JavaScriptに取り込まれつつあるが

Content Management System(CMS)の利用

24

- 編集したファイルをウェブサーバに転送するのが面倒
→ブラウザ上で編集作業までできない?
- 自由に編集できると編集が必要な部分が多くて面倒
- 携帯(スマートフォン)用のページも作りたいけど、一度の編集で両方作れない?



CMSの利用
(例: WordPress, Pukiwiki, tDiary)

- ブラウザ上から編集可能
- 各種テンプレート利用可能(テンプレート別ページも同時作成)
- バックアップや巻き戻しも自由自在
- 前述のCGI等で実装されている

各種CMS(1/4)

25

- 汎用型CMS
 - 見た目は普通のウェブページと変わらない
 - 管理者用ページから更新
 - WordPressが汎用型CMSの80%ぐらいを占めている
 - ただし、無駄にDataBaseと直結していてセキュリティ問題が多い
WordPressを使っているページの例



各種CMS(2/4)

26

- Blog型CMS
 - 日付単位でコンテンツを管理したい場合
 - コメントをつけてもらうことも可能

Blog型CMSの例(tDiary)



各種CMS(3/4)

27

- Wiki型CMS
 - 編集性を重視し、各ページから編集/ファイルアップロードができる
 - 多人数による編集を前提としているので、編集履歴を前面に押し出していることが多い



各種CMS

28

- グループウェア型CMS
 - ユーザ分けをしっかりとっており、コンテンツを操作したユーザの記録がしっかり残る
 - 共有カレンダーや会議予定調整などの機能もあつたりする



Wiki型CMSの利用のすすめ

29

- 個人管理が緩い形で情報集積に便利
 - 研究室内、サークル内、など
- 情報共有にも便利
 - 全員へのアナウンス、後輩に残したいTips、など
- ということで、Wiki型CMSのPukiwikiを使ってみましょう
 - おそらく、もっとよく使われているWiki型CMS

PukiWikiの基本のページ構成

- 上部に編集用のリンク
- 左に最新の更新情報
- 右に閲覧中のページ



PukiWiki上での既存ページの編集(1/3)

1. 編集したいページを開く
2. ページ情報の「編集」のリンクをクリック
 - 下のような編集ページが開く



PukiWiki上での既存ページの編集(2/3)

The screenshot shows the edit page for 'FrontPage' with several callout boxes explaining editing steps:

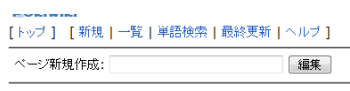
- 既存ページをコピーして編集する時**: Points to the top of the page.
- 編集のテキストボックス**: Points to the main content area.
- 編集結果をプレビューしたい場合**: Points to the 'プレビュー' (Preview) button.
- プレビューなしで更新する場合**: Points to the '更新' (Update) button.
- 書式のルール確認 (別タブで開くと便利)**: Points to the '書式のルールを確認' (Check the style rules) link.
- 直近の更新リストに載せたくない場合**: Points to the '直近の更新リストに載せたくない' (Don't show in recent updates) checkbox.

PukiWiki上での既存ページの編集(3/3)

3. 適当に編集してプレビューした後にページを更新
 - 慣れたら直接ページを更新しても良い
- 「テキスト整形ルールの表示する」は別タブで開いておくの良い
 - ルールは、編集用のテキストボックスの下に来るので、ページをスクロールさせないと出現しない点に注意
 - 外部のWikiの使い方ページを別タブで開いておくのも良い
 - 誤記修正などの細かな修正で、直近の更新リストに載せたくない場合、「タイムスタンプを更新しない」をチェック

PukiWiki上での新規ページ作成

1. 「新規」のリンクを選択
2. 新規ページ名の入力を求められるので入力
 - 「既存のページ名/新規ページ名」の形で入力すると、既存ページの下にページ扱いになる
3. あとは既存のページの編集と同様



PukiWiki上でのファイルのアップロード

1. 添付ファイルをアップロードしたいページを選択
 - 添付ファイルはアップロードしたページに紐付けられる点に注意
2. 「添付」のリンクを選択
3. 添付ファイルを選択して「アップロード」を押す
 - 添付したファイル一覧は、そのページの下方に表示されます
 - 本文中から「&ref(ファイル名);」の書式で呼び出せます



Wikiの文法とHTMLタグの対応例(1/3)

- 見出し(<h>)
 - アスタリスクを冒頭に置く
 - 例: *見出し → <h1>見出し</h1>
 - アスタリスクの数が多いほどより下位の見出しになる
 - 見出しに対しては<a name>要素が付与されるため、見出しに対してリンクを張ることができる
- ボールド(): シングルクォーテーション2つで囲む
 - 例: 『強調』 → 強調
- 表組み(<table>)
 - 「|」で表の要素となる文字を区切る
 - 「>」だけを書くと、そのマスは右のマスと合体する
 - 「~」だけを書くと、そのマスは上のマスと合体する

Wikiの文法とHTMLタグの対応例(2/3)

- (数値付き)箇条書き()
 - +や-を冒頭に置く
 - 例: - 1個目 → 1個目
 - 数が増えるほど下位のものになる点は見出しに同じ
- ハイパーリンク<a href>
 - URLの場合はURLを書くだけでもリンクを貼れる
 - ただし、URL自体が表示される
 - リンクに名前をつけたい場合は、「[[URL:リンク名]]」の書式
 - 間に「:」が入る点に注意
 - Wiki内ページは [[ページ名]] のようにページ名を大括弧2つで囲むだけでOK

Wikiの文法とHTMLタグの対応例(3/3)

- (左/中央/右)寄せ (<... align="left/center/right">)
 - 冒頭に「LEFT: / CENTER: / RIGHT: 」を書く
- 水平線 <hr>
 - 「----」とハイフンを4つ重ねる
- その他、色々テキストの装飾方法があるので、テキストの整形ルールを見ましょう
 - いきなり「ブロック要素の説明」が出て心を折りますが、まずは、気にせず先に読み進めましょう
 - 「ブロック要素」で指定できる内容と「インライン要素」で指定できる内容を見比べた方が、理解しやすいと思います

WikiのTips

- うっかり編集されて困る場合は、「凍結」のコマンドで編集禁止にすることが可能
 - 要管理者パスワード
 - 編集状態にして、そのページの書き方を見ることができなくなるのが難点
 - 実習用ページのトップページは凍結してあります
- バックアップ、リストア、ロールバックなど、入力したデータを保全する機能があります
 - バックアップと現在の差分を見ることができます

CMSのTips

- 無料でCMSを使えるサーバがあるので、利用すると便利
 - もちろん、強制的に広告が表示されます
 - 例: @wiki, wikiwiki.jp, 各種blogサイト
 - 自分でレンタルサーバを借りて、CMSを設置するのもあり
- ちゃんと最新バージョンのCMSソフトウェアを使っていないと、サイバー攻撃を受けることもあります
 - 最近では、WordPressを使っている所が攻撃されることが多い
 - 攻撃されるだけで済めばまだましで、攻撃の踏み台にされたら目も当てられないことに...
 - 学内でやらかすと、「証拠保全や安全性の確認」のために、しばらくそのサーバは使えなくなります