

2016年度 メディアプロジェクト演習1

HTML 講座 基礎編

立命館大学情報理工学部 メディア情報学科

1 はじめに

本演習の最初の4回では、HTML・JavaScriptを用いたWebページ作成の基礎知識を学びます。Webページを作成するには、次の2通りの方法が考えられます。

- (1) Webのパブリッシングソフトを使う。
- (2) テキストエディタを用いて直接HTML文書を作成する。

本演習では、2つ目の、直接HTML文書を書いてWebページを作るという方法を採用して進めます。これは、Webページがどのように作られているか、その原理を知ることが目的とされているためです。この知識を習得しておくことで、Webページ作成用ソフトウェアを使用するときにも、その利用方法の理解やトラブル解決などに、必ず役に立つことと思います。

本演習を行うにあたり、テキストおよびサンプルデータ、Webページを作成する上で参考となるページへのリンクなどを以下のWebページに置いておきますので、参考にしてください。

<http://www.rm.is.ritsumei.ac.jp/lecture/mediaP2016/>

2 インターネットに関する基礎知識

2.1 インターネットとは

インターネットとは、ひとことで言うと、「ネットワークのネットワーク」であると言えます。たとえば、ある学校にある複数のコンピュータが、互いに接続されていると仮定してみましょう。この状態では、学校内のコンピュータに限って、お互いに情報をやりとりすることができます。このようなネットワークを、LAN (Local Area Network)と言います。さて、隣の学校でも、同じようにネットワークを組んでいました。

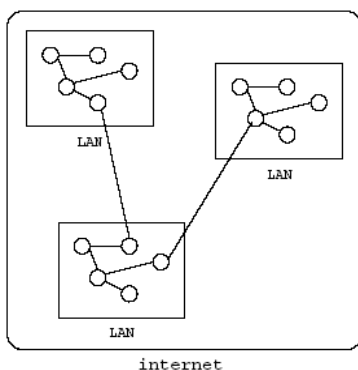


図1 小さいインターネット

そこで、この2つの学校の間をネットワークで繋ぐことにしました。これで、この2つの学校の間では、自由に情報のやりとりができるようになります。このようにできたネットワークは、小さいながらも、「インターネット」の1つです(図1)。

通常「インターネット」と呼ばれているものは、このように相互に接続されたネットワークのうち、アメリカの国防省の高度研究計画局(Advanced Research Projects Agency: ARPA)の実験ネットワークであった、ARPANETが母体となって誕生した、世界最大の相互接続ネットワークを指します。英語では、これらはきちんと区別できるようになっており、前者が"internet"と表記されるのに対し、後者は"The Internet"と、定冠詞付きで表現されます。以後、「インターネット」と表記した場合は、"The Internet"を指すものとします。

2.2 インターネットで何ができるか

このように、インターネットとは相互に接続されたネットワークの総体を表す言葉です。インターネットに接続された機器は、IP (Internet Protocol) という規約(プロトコル)に基づいて情報をやりとりします。実際には、IPを用いたさらに高度なプロトコルである、TCP/IP (Transmission Control Protocol/Internet Protocol)を用いて通信が行われることがほとんどです。そこで、インターネットのプロトコルはTCP/IPである、と言われることもあります。インターネット上で、TCP/IPに基づいて情報をやりとりすることにより、様々なアプリケーションが実現されています。代表的なものとしては、

- ・ 電子メール
- ・ ファイル転送 (FTP)
- ・ World Wide Web (WWW)

などが挙げられます。

3 WWWに関する基礎知識

3.1 World Wide Webとは?

World Wide Web (WWW, W3, Webなどと略されます)は、1989年に欧州原子核研究機構(CERN)のTim Berners-Leeにより提案された、広域情報閲覧システムです。WWWは、"Web"と呼ばれるように、様々な情報が蜘蛛の巣のように相互に繋がったもののことです。WWWでは、様々な情報の間を自由にジャンプ

して閲覧することができます。このように、他の情報にジャンプすること、あるいはジャンプするためのインデックスのことを、WWWの世界ではリンクと呼びます。次々にリンクを辿って様々な情報にアクセスしている様子を、WebとWave(波)を引っ掛けて、ネットサーフィンと呼んだりもします。WWW上で閲覧できる情報のことを、WebコンテンツやWebページ、あるいは単にページなどと呼びます。

3.2 Web ページはこのようにできている

WWWを閲覧しているとき、実に多彩な見かけを見ることができます。これらは体裁こそ様々ですが、すべてのWebページは「HTML(Hyper Text Markup Language)」という言葉を使って書かれた「HTMLファイル」でできています。言語というと難しいものようですが、ルールは決して難解なものではありません。

試しに、どこかのWebサイトを表示させてHTMLファイルの内容を見てみましょう。Firefoxを起動して、朝日新聞のWebページ(<http://www.asahi.com>)を開いてみてください。Firefoxの「ツール」メニューから「Web開発」を選択し、さらに「ページのソース」を選択すると、現在表示されているWebページを構成しているHTMLの内容が表示されます(図2)。WebブラウザはこのHTMLの内容を解釈し、文章や表などを指定されたとおりに表示しているのです。

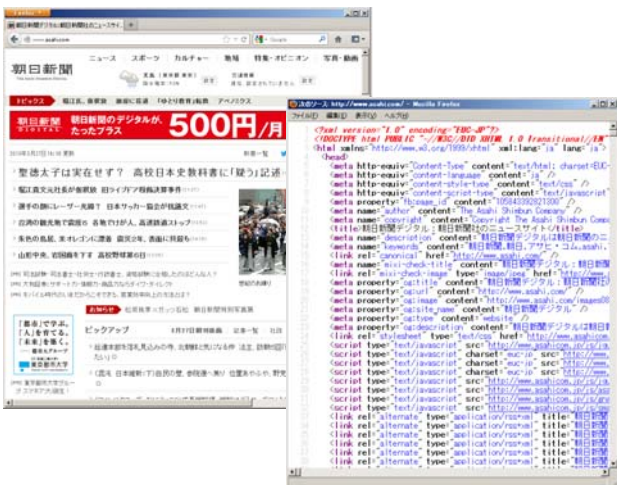


図2 メニューから「ソース」を選択すると、Webページを作るHTMLファイルの内容が表示される

HTMLのルールは何度か改正されてきています。2014年にはHTML5が勧告されましたが、現在は、HTML4.01が主に利用されているので、この演習では、HTML4.01 Transitionalに従って演習を進めていくことにします。

3.3 HTML タグ

先ほど表示させたHTMLの内容を見てみると、「<>」で囲まれた文字がたくさんあることに気づくでしょう。「<html>」や「<head>」「<p>」など、「<>」で囲まれた文字列のことをタグといいます。

既にLaTeXの講義で習ったように、タグは、文書に様々な意味づけをするために使われます。

レポートや書類を作るときのことを考えて見ましょう。見出しは太字にして大きくし、リストは番号を振るなど、見た目を調整してそれぞれの部分がどういう意味を持っているのかが分かるようにするでしょう。HTMLのタグはこの「意味づけ」のために使われるのです。

HTMLのタグは数十種類あり、タグを使うことで「ここからここまでが本文」「ここに画像を挿入する」「これはリストの項目」「これは表の項目」といった意味づけができます。

4 Web ページ作成に必要なソフト

4.1 Web ブラウザ

Webページを閲覧するためには、Webブラウザ(正しくはUser Agent)と呼ばれる閲覧ソフトウェアが必要になります。ブラウザにはInternet ExplorerやFirefox, Chrome, Safari, Operaなど、様々な種類があり、その表示能力(どんな字体が表示できるか、グラフィックが表示できるか等)は、千差万別です。

演習室のコンピュータには、Linuxで起動するとFirefoxが、Windowsで起動するとInternet ExplorerとFirefoxが入っています。

4.2 テキストエディタ

Webページを構成するファイルのうちHTMLファイルは文字情報だけが入っている「テキストファイル」の一種です。テキストファイルの作成や編集には、「テキストエディタ」というソフトウェアが向いています。

Linuxでは、演習などで既に使用しているemacsを使うとよいでしょう。本演習では、Linux上でemacsを用いてHTMLファイルの作成・編集を行います。

Windowsの場合は、標準添付されている「メモ帳」といったテキストエディタを使うとよいでしょう。メモ帳などのテキストエディタは、文字だけを対象としており、その属性を扱うことはできません。

MS-Wordや一太郎などのワープロソフトでも、テキストファイルの編集はできますが、単に文字を扱うだけではなく、文字の位置、形(フォント)、大きさ、色、その他の装飾(網掛けなど)といった属性も同時に扱うので、不向きです。

「文字を扱う」という視点から、一見ワープロもテキストエディタも同じように見えますが、

属性を扱うか否かという点で大きな違いがあります。

4.4 FTP ソフト

コンピュータ上で作成・編集した HTML ファイルや画像ファイルをインターネットで公開するためには、Web サーバ（HTTP サーバともいう）から見える位置にそのファイルを置かなければいけません。

Web サーバに Linux のような UNIX 系の OS が入っている場合には、各ユーザがファイルを置いておく場所が用意されています。このような場所を、ユーザのホームディレクトリと呼びます。立命館大学の RAINBOW システムでは、皆さんが Linux を使った演習でいつも作業しているディレクトリがホームディレクトリに当たります。公開する HTML ファイルや画像ファイルの置き場所は、通常ホームディレクトリ下の“public_html”という名前のディレクトリに割り当てられます。“public_html”ディレクトリの下に“index.html”という名前の HTML ファイルを置くことで、それが自分の Web サイトのトップページになります。

立命館大学の RAINBOW システムでは、ホームディレクトリは、皆さんが今使っているコンピュータとは別の場所に置かれたサーバ上にあります。RAINBOW では、Linux で起動したときには、サーバのホームディレクトリ上で簡単に作業ができるよう設定されていますが、Windows の場合、そのような設定はされていません。このため Windows 上で HTML を編集する場合は、ホームディレクトリにファイルを転送する必要があります。このときネットワークを介してファイルを転送するソフトを使わなければいけません。このようなソフトを FTP ソフトと呼びます。Windows で利用できる代表的な FTP ソフトには「FFFTP」があります。

ここから
start

5 Web ページのひな形作成と閲覧

5.1 最も簡単な HTML ファイル

基本課題 1: 以下の手順に従って、ひな形となる HTML ファイルを作成してください。

(1) UNIX でログイン，端末の起動

UNIX でログインし、GNOME 端末を起動します (RAINBOW GUIDE 2016 Linux 操作入門編 p.22~24)。

(2) emacs の起動

GNOME 端末から emacs を起動します。

(3) HTML ファイルの作成

そして、次のように打ち込んで下さい。日本

語の部分は、UNIX 上の日本語入力システム、Anthy, iBus などを使って入力して下さい (RAINBOW GUIDE 2016 Linux 操作入門編 p.25, 35, 67~69)。

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.01 Transitional//EN">
<html>
<head>
<title>タイトル</title>
</head>
<body>
<p>本文</p>
</body>
</html>
```

そして、このファイルを各自のホームディレクトリ以下の「public_html」ディレクトリに保存します。ファイル名は「index.html」とします。

(4) 作った HTML ファイルの確認

それでは早速、実際に WWW ブラウザを用いて、作成した HTML 文書を確認してみましょう。まずは、Firefox を起動します。作成したファイルをドラッグ&ドロップするか、ファイルメニューから、作成したファイルを開いてください。以下のような画面が出てくるとと思います。

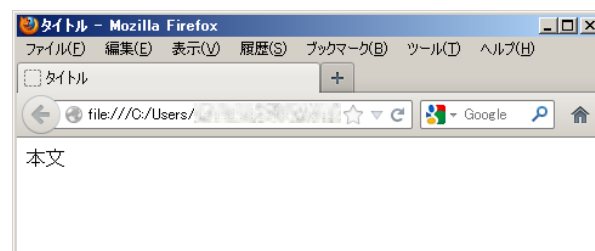


図 3 ひな形ページ

(5) 自分の Web ページを閲覧

次に、Web サーバを通して皆さんの作ったページを閲覧することができるか確認してみましょう。

Web ブラウザを立ち上げて、URL <http://www.ritsumei.ac.jp/~アカウント> を開きます。

もし、エラーメッセージが出て開くことができなかつたら、ファイル名が間違っていないか、ディレクトリが間違っていないかをチェックしてみてください。

なお、個人の Web ページは学内からのみ閲覧が可能で、学外に公開するためには申請が必要です。

6 HTML の基本

6.1 開始タグと終了タグ

ここで、さきほど作成したひな型ページのソ

ースをもう一度見てみましょう。<html>と</html>など、似たタグが対になっていることに気づいたでしょうか。

HTML では、「ここからここまでが HTML 文書である」といった指定を、<html>などのタグで行います。指定する範囲の前後に「開始タグ」と「終了タグ」を記述すれば、指定は完了です。開始タグの<html>に「/」を足した</html>が<html>の終了タグです。

また、開始タグから終了タグまでの範囲を「要素 (エレメント)」と呼び、タグで囲まれた範囲を「内容」と呼びます。たとえば、「<title>HTML 講座</title>」は全体が「title の要素」「HTML 講座」の部分が「title 要素の内容」にあたります。

一方、開始タグを書くだけで指定が完了するタグもあります。例えば画像を表示するタグには終了タグはなく、などと記述する必要はありません。この場合、タグのみが要素のすべてであり、「内容」はないこととなります。

HTML では、開始タグ、終了タグを入れ子にするのがルールです。たとえば、「<html>～<head>～」と 2 つの開始タグを記述したら、終了タグは「～</head>～</html>」の順で記述します。「<html>～<head>～</html>～</head>」という順でタグを書くことはできません。また、特定のタグはあるタグの範囲にのみ記述できる、といったルールや特定のタグは HTML ファイルの中に一回のみ記述できる、といったルールもあります。これらのルールは、それぞれのタグの説明に合わせて紹介します。

6.2 属性と属性値

タグの中には、のように「img」という要素と、それに関する細かな指定 (src="ファイル名"以降) から成り立っているものもあります。

このように、タグの中に書かれる細かい指定は「属性」と「属性値」といいます。「src="ファイル名"」という部分は、「src」が属性、「ファイル名」が属性値です。属性値は、「"」(ダブルコーテーション) でくくります。

ここでは、「src」と「alt」という 2 つの属性が指定されています。このように属性を複数同時に指定する場合は、「属性="属性値"」どうしを半角スペースで区切って記述します。属性を記述する順序にはルールはなく、どの順に記述してもかまいません。

タグは、10 章で説明します。

6.3 基本のタグ

まず、先ほどひな型ページで書いた HTML 文書の内容を解説していきます。

(1) HTML のバージョン宣言

HTML で記述する文書では、その文書がどのバージョンの HTML によって記述されたものかを記述しなくてはなりません。そこで、文書の最初には DTD (Document Type Definition) 宣言というおまじないを記述します。例えば、HTML 4.01 Transitional を用いて文書を記述する場合、文書の最初に、以下のような DTD 宣言を入れなくてはなりません。

(sample1.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

(2) HTML 要素

次に、HTML で書かれた文書であることを示すために、文章全体を<html>というタグで囲み、html の要素にします。

(sample2.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>

</html>
```

(3) head 要素と body 要素

html 要素は、head と body の、2 つの要素を含まなくてはなりません。これらの要素は<head>と<body>というタグで囲んで示します。

head 要素にはその HTML 文書についての情報を記述し、body 要素には本文を記述します。<head>タグも<body>タグも、1 つの HTML ファイルに 1 度のみ<html>と</html>の間に記述します。

(sample3.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
</head>
<body>

</body>
</html>
```

(4) title 要素

head 要素には、title 要素を含まなくてはなりません。title 要素は、<title>タグで囲んで、その文書の内容を端的に表す表題を示します。ここに示された内容は、Web ブラウザのタイトルバーの文字列や、ブックマークやお気に入り、あ

るいはホットリストなどと呼ばれる個人のリンク集などの見出しとして使用されます。ですから、それを見ただけでページの内容が想像できるような表題を付けることが肝心です。

(sample4.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Transitional//EN">
<html>
<head>
<title>タイトル</title>
</head>
<body>

</body>
</html>
```

7 文章構造

7.1 見出し

見出しを付けるためには、h1, h2, h3, h4, h5, h6 (Heading) 要素を使用します。

通常、文章を書くときには、大節、中節、小節、小小節などのように、内容が細分化されるに従って小さな見出しを使います。h 要素も同様に、h の後に続く数字によって見出しの大きさを指示します。この数字は見出しとしての大きさを示すものであって、物理的なフォントの大きさを指定するものではないことに注意してください。h1 が最も大きい見出しで、h6 が最も小さい見出しとなります。

タグ	<h1><h2><h3><h4><h5><h6>~</h1~</h6>
機能	h1~h6 で、見出しの大きさを指定します。

(sample5.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Transitional//EN">
<html>
<head>
<title>見出しタグ</title>
</head>

<body>
以下の様に、見出しタグを用いることで見出しの大きさを指定できます。
<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>
</body>
</html>
```

基本課題 2： ひな形ページに、見出しタグを使って、名前、出身地、出身高校を書き加え、自己紹介のページを作ってください。

7.2 段落

段落を指定するには p (Paragraph) 要素を使用します。ひとまとまりの文章を 1 つの p 要素にすると、WWW ブラウザは段落間に適度なスペースを挿入するなどして意味の区切りが解り易くなるようなレイアウトを施します。

タグ	<p>~</p>
機能	段落の範囲を示します

(sample6.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Transitional//EN">
<html>
<head>
<title>段落タグ</title>
</head>

<body>
このように文章中に段落タグを入れることで<p>その部分は段落で表示され</p>本文中にも関わらず段落化されます。
</body>
</html>
```

7.3 改行

HTML のソースリストで改行を入れても、ブラウザで表示する際には改行されません。そこで、強制的に改行を行いたい場合には br (BReak) 要素を用います。ブラウザ上で改行して表示するには、改行したい位置に
タグを入れます。
タグは強制的に改行する位置を示すためのタグです。
タグには終了タグはなく、単に
と記述すればその位置で改行されます。

タグ	
機能	改行を入れる位置を示します。

(sample7.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Transitional//EN">
<html>
<head>
<title>改行タグ</title>
</head>
<body>
HTML では改行キーを入力しても
その部分で改行されず続けて文章が表示されます。<br>
このように、HTML では文章中に改行タグを入れることで<br>
その部分で改行され次の行に文章が表示されま
```

す。
 </body>
 </html>

7.4 水平線

<hr>タグは、Web ページに水平線を引くためのタグです。文書の区切りによく使われます。
 <hr>タグは
タグ同様終了タグがありません。
 <hr>と記述すれば指定は終了し、その位置に水平線が表示されます。

タグ	<hr>
機能	指定した場所に水平線を引きます。

(sample8.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Transitional//EN">
<html>
  <head>
    <title>水平線タグ</title>
  </head>

  <body>
    以下の様に、水平線タグを用いることで<hr>
    上の様に表示されます。
  </body>
</html>
```

基本課題 3： 自己紹介のページの、名前、出身地、出身校の下に水平線を引き、その下に段落タグと改行タグを使って自分の趣味を複数個並べて書いてください。

8 文字の修飾

8.1 強調, イタリック体

文字(列)を強調する要素として、em 要素と strong 要素があります。

タグ	~
機能	強調

タグ	~
機能	さらに強い強調

ただし、これらの要素の内容中に h 要素や p 要素を入れることはできません。

(sample9.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Transitional//EN">
<html>
  <head>
    <title>強調タグ</title>
  </head>

  <body>
```

```
文字を強調したいときは、<br>
<em>強調</em><br>
<strong>さらに強い強調</strong><br>
など指定できます。
</body>
</html>
```

8.2 色

文字、または文字列の修飾のために、直接フォントの大きさや色を指定する要素があります。これが font 要素です。

タグ	
機能	フォントサイズと文字色を指定します。

size 属性の属性値は、絶対値または相対値でフォントサイズを指定します。絶対値による指定の場合は、1 から 7 の自然数で指定します。相対値による指定の場合、その時のフォントのサイズに対して「-3」や「+2」等のように指定します。

color 属性の属性値としては、以下にあげる色の中から 1 色を指定します。

black, silver, gray, white, maroon, red, purple, fuchsia, green, lime, olive, yellow, navy, blue, teal, aqua

この色指定方法では 16 色種類の中から選択するしかありません。より多くの色を指定するための方法については、発展編 1 で学びます。

(sample10.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Transitional//EN">
<html>
  <head>
    <title>フォントタグ</title>
  </head>

  <body>
    文字色を変えたい場合は
    <font color="RED">このようにフォントタグで
    色を指定することで</font>
    色を変更することができる。<br>
    色の指定は色々できますが、
    <font color="RED">赤</font>,
    <font color="BLUE">青</font>,
    <font color="GREEN">緑</font>,
    <font color="YELLOW">黄</font>,
    <font color="PURPLE">紫</font>,
    など指定できます。
  </body>
</html>
```

基本課題 4： 自己紹介ページのテキストの重要箇所を強調タグで強調したり、フォントタグを使って色づけしたりして目立つようにしてください。

9 リスト（箇条書き）

箇条書きのリストを作るには、~と~タグを使います。~タグの中に、各項目を~タグで囲んで記述すると、各項目のまえに「・」が表示されます。

なお、リストは、<p>~</p>要素の内容としては記述できないというルールがありますので注意してください。

タグ	~
機能	箇条書きのリストを定義します。

タグ	~
機能	リストの各行を定義します。

(sample11.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <head>
    <title>リストタグ</title>
  </head>

  <body>
    幾つかの項目をリスト表示するには、以下の様に記述します。
    <ul>
      <li>リスト 1</li>
      <li>リスト 2</li>
      <li>リスト 3</li>
    </ul>
  </body>
</html>
```

基本課題 5： 自己紹介のページの趣味を箇条書きに変えてください。

10 画像

文書中に画像を埋め込むための要素は img (IMaGe) 要素です。img 要素には終了タグはありません。単独のタグだけの要素です。img 要素にはいくつかの属性があり、これにより img 要素に情報を与えます。

img 要素の属性で必ず与えなければならないものとして、src (SouRCe) 属性と、alt 属性があります。src 属性は、img 要素に、文書に埋め込む画像のファイル名を指示するもので、ファイル名、もしくは、インターネット上の別の場所にあるファイルの場合は、そのファイルの URL を指定します。alt 属性は、画像が表示できない

環境、例えばテキストベースの WWW ブラウザやグラフィック表示をオフにしてある WWW ブラウザの場合に画像の代わりに表示する文字列を指定します。また、目が不自由で読み上げソフトを使用して Web をブラウズしている人の助けにもなります。

タグ	
機能	「ファイル名」で指定された画像を表示します。

(sample12.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
  <head>
    <title>イメージタグ</title>
  </head>

  <body>
    HTML では画像を表示させることもできます。
    <br>
    
  </body>
</html>
```

基本課題 6： 自己紹介のページに画像を貼り付けてください。画像は、以下の URL にありますので、好きなものをダウンロードして使ってください。
<http://www.rm.is.ritsumeai.ac.jp/lecture/mediaP2016/img/>
 注) 貼り付けた画像は、「public_html」ディレクトリに保存する必要があります。

11 表

表を作るには、最低でも<table>~</table>、<tr>~</tr>、<td>~</td>の 3 つのタグを使います。<th>~</th>は表の項目が見出しである場合に、<td>~</td>の代わりに使うタグです。

HTML による表は、「表全体 (table 要素)」「それぞれの行 (tr 要素)」「行の中の項や見出し (td 要素や th 要素)」からなります。

表を構成する項目の 1 つ 1 つを<td>~</td>や<th>~</th>で定義し、それを 1 行分まとめて<tr>~</tr>で囲みます。表の行数だけ<tr>~</tr>を集めて<table>~</table>で囲めば、それが 1 つの表となって表示されるというしくみです。

```
<table>
<tr>
  <td> 項 1 </td>
  <td> 項 2 </td>
</tr>
<tr>
  <td> 項 3 </td>
  <td> 項 4 </td>
</tr>
</table>
```

図 4 表の書き方

タグ	<table>～</table>
機能	表を定義します。

タグ	<tr>～</tr>
機能	表の行を定義します。

タグ	<td>～</td>
機能	表の各項目を定義します。

タグ	<th>～</th>
機能	表の各項目が見出しである場合に使用します。

(sample13.html)

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML
4.0 Transitional//EN">
<html>
  <head>
    <title>テーブルタグ</title>
  </head>

  <body>
    表を定義するには、以下の様に記述します。
    <table>
      <tr>
        <td>一行一列目</td>
        <td>一行二列目</td>
      </tr>
      <tr>
        <td>二行一列目</td>
        <td>二行二列目</td>
      </tr>
    </table>
  </body>
</html>
```

基本課題 7： 自己紹介のページに 2016 年度前期の皆さんの時間割表を追加してください。

12 HTML のルール

HTML を記述する際には、いくつかのルールがあります。ここでは、HTML を記述する上で気にするべきところ、気にしなくてもよいところを紹介します。

(1) タグは大文字で書いても小文字でもよい

タグと属性はともにアルファベットで記述しますが、大文字で書いても小文字で書いても同じものとして解釈されます。

(2) タグの前後の改行は無視される

タグの前後で改行しても、表示結果は変わりません。改行を連続して入れ、なにも記述しない行があっても同じです。

(3) 本文中の改行はスペースに置き換えて表示

される

文書の途中で改行した場合は、半角スペースに置き換えられて表示されます。これは、英文を行の途中で改行した場合に、単語と単語の間が開いて表示されるようにするためです。

(4) ウィンドウの端で改行する指定は不要

Web ブラウザではウィンドウの端で自動的に行が折り返して表示されるため、1 行が長い段落であっても改行を指示する必要はありません。

(5) 複数の半角スペースは 1 つのスペースとして表示される

ブラウザに表示される文章の中に複数の半角スペースを入れると、1 つの半角スペースに置き換えて表示されます。一方漢字入力モードで入力したスペース（全角スペース）はいくつ入力しても置き換えられることはなく、すべて表示されます。

(6) ブラウザに表示されないメモを入れるには

HTML ファイルの中に、ブラウザには表示されない文を入れることができます。ページ作成に当たってのメモを記入しておくとう便利です。

表示させたくない部分を「<!--」と「-->」で囲めば、その部分はブラウザが解釈しないコメントとして扱われ、ブラウザのウィンドウにも表示されなくなります。コメントはソースリストのどこにあってもかまいません。また、複数行にわたるコメントを記入することもできます。

(7) ブラウザがサポートしていないタグは無視される

ブラウザが未対応のタグや属性があった場合、ブラウザはそれらのタグや属性をまったく無視して表示します。

(8) HTML 文書はブラウザごとに見え方が異なる

HTML 文書はブラウザによって見え方は異なる場合があります。どのようなブラウザでも快適に見られるように配慮することが、沢山の人に見てもらおうための重要なファクターとなります。ですから、ページを書いたらできるだけ沢山の WWW ブラウザで見栄えのテストをすることをお勧めします。

参考文献

[1] Raggett, D., H. L. Araud., and Ian Jacobs, eds., HTML 4.01 Specification, W3C Recommendation, World Wide Web Consortium, 1999.

[2] HTMLとスタイルシートによる最新Webサイト作成術—ホームページでなにを伝える? どう作る?, エクスナレッジ, 2002.