情報教育演習II 第2回

このページのアドレスはこちら。

- http://arena.hyogo-dai.ac.jp/~kawano/?Lecture/JouhouC2010

「HTML(1)」

Webの中心的な技術「HTML」について、今回から数回程度、実習をします。

授業の内容

1. World Wide Webとハイパーテキスト
2. World Wide WebとHTML
3. Webページの作成
4. HTMLの書き方
5. HTMLファイルの「骨組み」をつくる
6. 「見出し」をつくる
7. 「段落」と「改行」をつくる
8. 「リンク」をつくる
9. 「水平線」をつくる

今回の課題

1. インターネット上の記事へのコメントをHTMLファイルにする

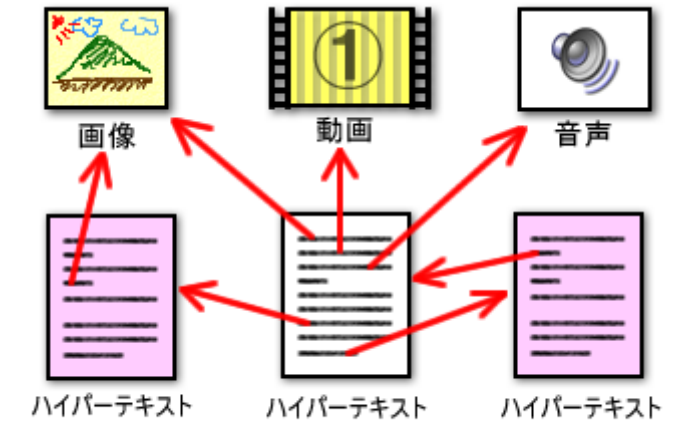
World Wide Webとハイパーテキスト

「ハイパーテキスト」と「ハイパーリンク」

「本」という文書（テキスト）は、１ページ目から、順番に、書かれ読まれます。

「**ハイパーテキスト**」(Hyper Text)とは、ある文書の中に他の文書への「位置情報」を埋め込むことで、文書同士を相互に結びつけることができる、コンピュータを利用した文書のひとつです。テキスト(Text)を超える(Hyper)、というところから名づけられました。

「**ハイパーリンク**」(Hyper Link)とは他の文書や様々な情報（画像、音声、動画など）への位置情報のことです。リンク(Link)ともいいます。



ハイパーテキストの概念（赤い矢印：ハイパーリンク）

ハイパーテキストを使えば、膨大な情報を、順番に読む必要はなく、ハイパーリンクを利用して効率的に情報にアクセスすることができます。

ハイパーテキストの特徴

それぞれの情報は，リンクで結びついている

ハイパーテキストの情報は、本のように、必ずしも順番に並んでいるとは限りません。ハイパーリンクを使って、（人間の頭の中と同じように）自由に情報同士を結び付けています。

それぞれの情報は，（ある程度は）内容が独立している

ハイパーテキストの情報は、順番に並んでいるとは限らないので、本のように「前に書いてあること」を考慮した内容にはなっていません。それぞれの情報は、その中である程度完結したものになっています。

それぞれの情報は，独立して作成・管理される

ハイパーテキストの情報は、本のように、紙（メディア）の上に情報が張り付いてはいません。本の情報は、内容が変更したり新しい情報を追加するためには、本そのものを作り直す必要があります。しかし、ハイパーテキストは、そのテキストだけ変更・追加すればいいので、リンク全体を作成する必要がありません。

World Wide Web

Web (World Wide Web) は、1989年に**ティム・バーナーズ＝リー**が作った、インターネット上の情報を共有するためのしくみです。

このWebは、ハイパーテキストをインターネット上で現実化したものです。共通のフォーマットで作った文書（テキスト）をインターネット上に公開し、それらを、ハイパーリンクを使って、相互に参照できるようにしました。

ハイパーテキストを、インターネット上で、利用者のコンピュータ環境に関係なく、誰もが利用できるようにするため、Webでは次のような技術を使っています。

- URL: ネットワーク上の情報の位置を特定するための技術
- HTTP: Webのシステムで情報をやりとりするための技術
- HTML: コンピュータ環境に関係なく情報を表現するための技術

Webに関する技術開発や普及活動は、1994年に設立された、WWWコンソーシアム（通称W3C)が行っています。

次へ進んでください。

World Wide WebとHTML

HTML: Hyper Text Markup Language

HTML(Hyper Text Markup Language)は、WWW(World Wide Web)上で、ハイパーテキストを利用して情報を発信するための、コンピュータ言語のひとつです。W3C(World Wide Web Consortium)で改良が続けられています。

HTMLでは、定められたフォーマットにしたがって記述することで、インターネット上で情報を共有することができます。主に、次のような特徴があります。

テキスト情報である

特定のソフトで作った情報は共有することができません。しかしHTMLは、テキスト（文字）だけで書かれている情報です。テキストだけの情報はどのようなコンピュータ環境であっても、内容を読み書きすることができます。

ハイパーテキストの機能をもつ

インターネット上にある、他の文書や情報へのリンクを作ることができます。リンクを作成するには、リンクしたい情報へのURLを指定します。

情報のかたちを表現することができる

「情報のかたち」とは、文字の色やサイズを指定することではありません。「見出し」「段落」などの要素（文書を組み立てる部品）のことです。文書中に要素を示すマークをしておくことで、文書の構造（かたち）を表現することできます。

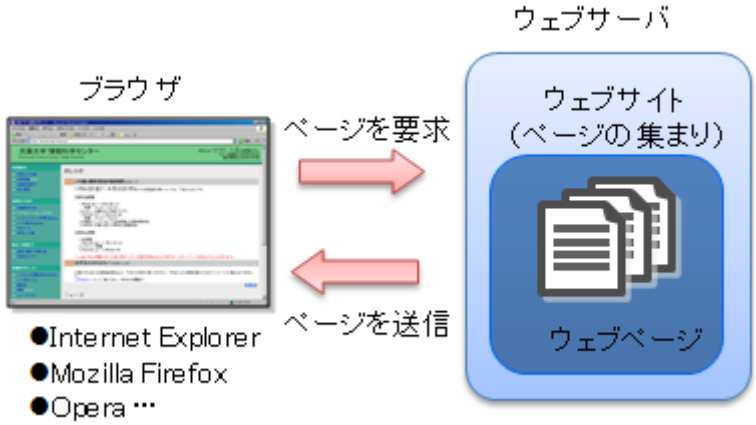
HTMLの歴史

HTMLは、登場して以来、数年おきに、新しいバージョン（版）に更新されています。現在のバージョンは、HTML4.01です。また、1998年に登場した「XML」(eXtensible Markup Language)に基づいて考案された、XHTML(eXtensible Hypertext Markup Language)もあります。

1993年	HTML 1.0
1995年	HTML 2.0
1997年	HTML 3.2
1997年	HTML 4.0
1997,999年	HTML 4.0、HTML 4.01
2000,2002年	XHTML 1.0
2001年	XHTML 1.1
2010年（予定）	HTML 5.0

HTML文書とブラウザ

HTML文書とは、HTMLで書かれた文書のことです。（HTMLファイルともいう） ファイル名は必ず半角文字で、拡張子（ファイル名の最後）は「**.html**」か「**.html**」になります。



Webサーバ上にあるHTML文書を、ブラウザが内容を解析して、Webページとして適切な表示をします。

ただし、HTML文書に書かれた、文書構造の目印の表現のしかたが、ブラウザによって多少異なります。また、ブラウザがどのHTMLのバージョンに対応しているかによっても、表示のできる範囲が変わってしまいます。現在の主なブラウザは、次のとおりです。

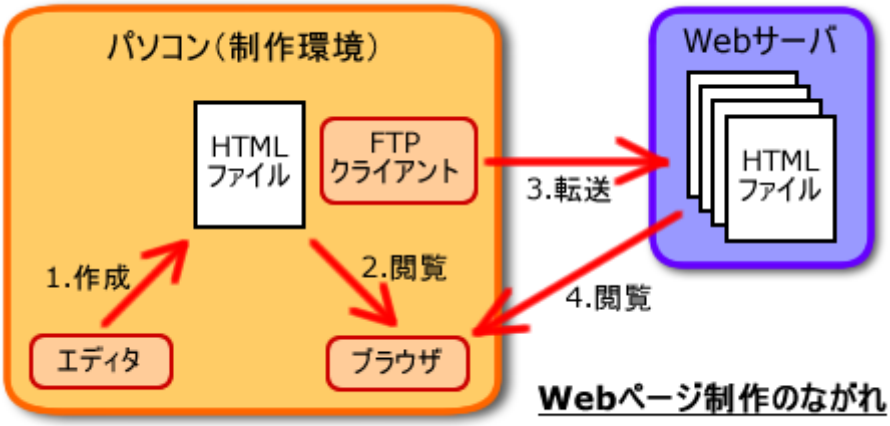
- Internet Explorer (マイクロソフト)
- Mozilla Firefox (Mozilla)
- Opera (Opera Software)
- Safari (アップル)
- Google Chrome (Google)

次へ進んでください。

Webページの作成

Webページを作成するながれ

Webページを作成するには、いくつかのソフトウェアを利用します。そのながれを見てみましょう。



HTMLファイルの作成

Webページの元になる、HTMLファイルを作成するには「**エディタ**」を使用します。エディタには、いくつかの種類があります。

アプリケーションソフト

ワープロソフトなどには、作成したファイルを、HTMLファイルとして出力する機能があるものがあります。

ホームページ作成ソフト

Webページ作成専用ソフトで、編集だけでなく、サーバへの転送や画像の編集機能がある、高機能なものもあります。
主なものには、ホームページビルダー （日本IBM）やDreamweaver （アドビ） があります。

テキストエディタ

もっとも、手軽にHTMLファイルを作成できる方法です。Windows付属の「メモ帳」やMacintoshの「Simple Text」で、直接HTMLを記述します。

Webページの閲覧

「**ブラウザ**」は、HTMLファイルを読み込んで、Webページとして表示してくれます。

作成中のHTMLファイルを読み込んで、でき具合を確認します。なお、同じWebページでもブラウザによって見栄えが異なる場合があるので、注意が必要です。

Webサーバへの転送

作成したWebページをインターネットに公開するには、Webサーバに転送する必要があります。

「**FTPクライアント**」を使って、作成したWebページをWebサーバへ転送したり、逆にWebサーバから現在のデータを取り寄せたりします。

テキストエディタを使うメリット・デメリット

メリット

- HTMLなどのWWWに関する技術の勉強になる
- 余計なデータが勝手に書き込まれない（見やすいファイル）
- ブラウザによる表示の違いなどを意識して作成できる

デメリット

- HTMLなどのWWWに関する技術をある程度理解する必要がある
- 作業に慣れるまで、作るのに時間がかかる
- 専用ソフトとは違い…
  - HTMLを自分で入力しなければならない
  - 入力中に出来上がりを確認できない（ソフトの切り替えが必要）

次へ進んでください。

HTMLの書き方

ここでは、HTMLの書き方について解説します。

要素と属性

要素のマークアップ

文章は、「見出し」「段落」「改行」「箇条書き」などの、いくつかの「要素」で構成されています。しかし、文書をひと目見ただけでは、その文書の構成要素は、人間にはだいたい見当がつきますが、コンピュータには簡単にはわかりません。

文書に目印（マーク）をつけることで、どの部分がどのような要素なのかをわかるようにすれば、コンピュータは簡単に判断することができます。

このように、文書の「**要素**」（element）を示すためのマークを、「**タグ**」といいます。下の例にある、<h1>や のように、「**山括弧<>で囲んだもの**」がタグです。<h1>～</h1>が見出し、 ～ が段落をあらわしています。

```
<h1>要素のマークアップ</h1>

<p>文章は、「見出し」「段落」「改行」「箇条書き」などの、
いくつかの「要素」で構成されています。</p>
```

要素は、たいていは「開始タグ」（<と>で囲んだもの）と「終了タグ」（</と>で囲んだもの）で囲みます。一部の要素は、開始タグだけで表します。タグは、原則として小文字で書くと良いでしょう。

属性

要素にさまざまな設定をする場合には、開始タグに「**属性**」(attribute)を設定します。

```
<タグ名 属性1="値1" 属性2="値2" ...>
```

開始タグのタグ名のあとに、スペースを空けて、「属性="値"」と記述します。値は引用符（""）で囲みます。

スペースで区切ることで、複数の属性を設定することができます。

タグの入れ子構造

ある開始タグと終了タグの中に、別の開始タグと終了タグを入れることができます。しかし、内側の要素の終了タグを記述する前に、外側の要素の終了タグを述することはできません。

```
正しい例：<aa>きょうは<bb>いい天気</bb>です</aa>
誤った例：<aa>きょうは<bb>いい天気</aa>です</bb>
```

次へ進んでください。

HTMLファイルの「骨組み」をつくる

HTMLファイルの作成

それでは、HTMLファイルを作成していきます。実際に入力して練習しましょう。それには、まずテキストエディタを起動します。

- 「TeraPad」を利用する場合
  a. 「スタート」ボタンをクリックし、「すべてのプログラム」→「TeraPad」→「TeraPad」を選択

  b. Tera Padが起動

- 「メモ帳」を利用する場合
  a. 「スタート」ボタンをクリックし、「すべてのプログラム」→「アクセサリ」→「メモ帳」を選択

  b. メモ帳が起動

なお、基本的に、**この授業では「TeraPad」を使用します**。

HTML文書の骨組み

HTML文書の構造は、次の３つの要素によって、内容を組み立てられています。

- **html要素**: HTML文書全体 (<html>...</html>)
- **head要素**: HTMLファイルの基本情報 (<head>...</head>)
- **body要素**: Webページとして表示される内容 (<body>...</body>)

head要素は、HTML文書に関する情報や設定を記述する場所です。HTML文書のタイトルもその中に記述します。

- **title要素**: Webページのタイトル
  - title要素は、head要素のなかで1つだけ記述できる
  - 設定したタイトルは、ブラウザのタイトルバーに表示される

例

```
<html>
<head>
  <title>Webページのタイトル</title>
</head>

<body>
  （Webページとして表示される内容）
</body>
</html>
```

ファイルの保存とブラウザでの確認

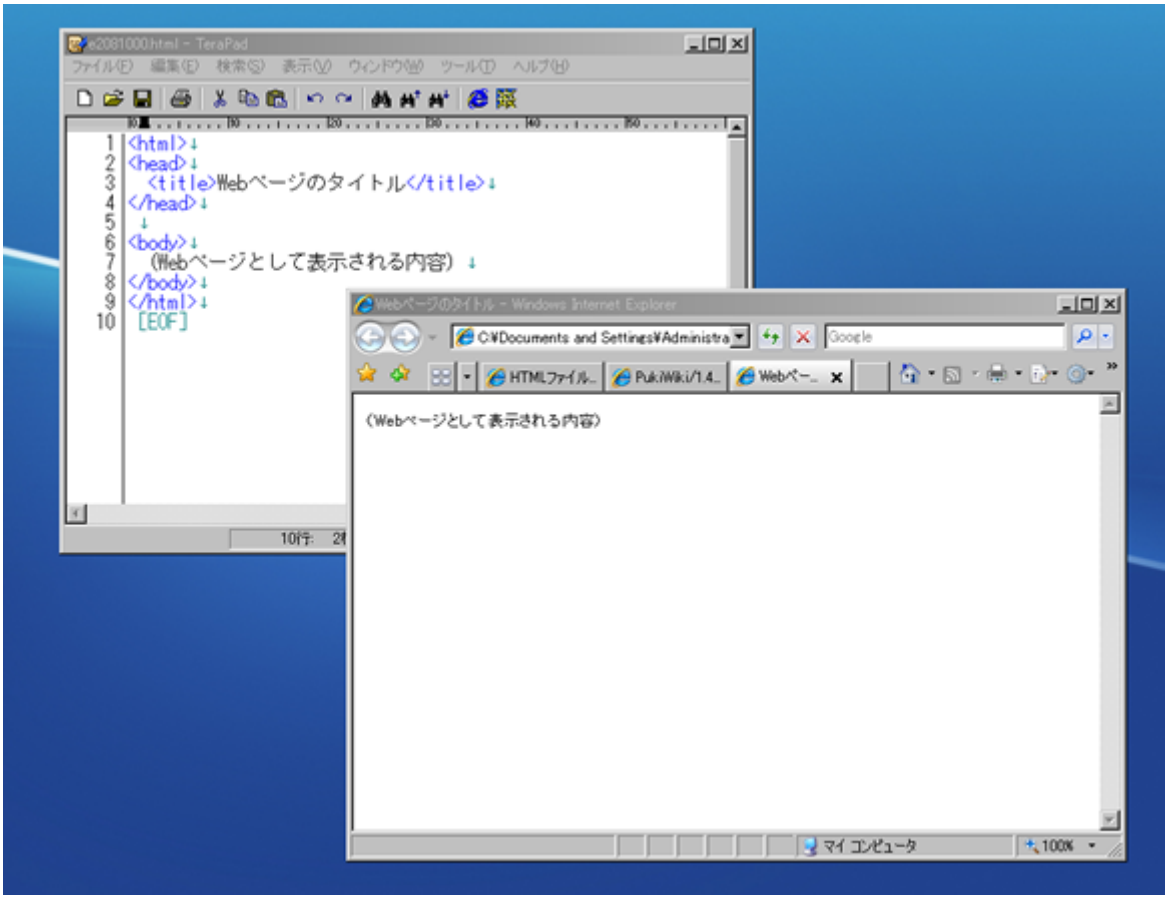
ひとまず、ファイルを保存してみましょう。

1. TeraPadのメニューから、「ファイル」→「名前をつけて保存」を選択
2. 「保存する場所」を「マイドキュメント」に設定
3. 「ファイル名」を「**自分の学籍番号**」+「**.html**」と入力
   (例: 学籍番号がC2092000の場合、c2092000.html)
4. 「保存」ボタンをクリック
5. HTMLファイルとして、内容が保存される。

TeraPadの場合、HTMLファイルとして保存すると、**タグや属性の部分が色分けして表示**されます。

HTMLファイルが保存できたら、ブラウザでWebページとして確認してみましょう。次のいずれかの方法で確認してください。

- Internet Explorerを起動
  a. 「マイドキュメント」を開いて、作成したHTMLファイルのアイコンを探す
  b. HTMLファイルのアイコンを、Internet Explorerのなかにドラッグする
- TeraPadのツールバーにある「Internet Explorer」のアイコンをクリック
- Webページが表示される



このあと編集作業は、次のように進めていくことになります。

1. HTMLファイルを編集したら、上書き保存
2. ブラウザのツールバーにある、「更新」ボタンをクリック

3. HTMLファイルが上書きした内容で読み込まれ、Webページが再表示される

次の説明から、いくつかのタグを紹介します。実際に入力してみて、どのように表示されるか、よく確認してください。

次へ進んでください。

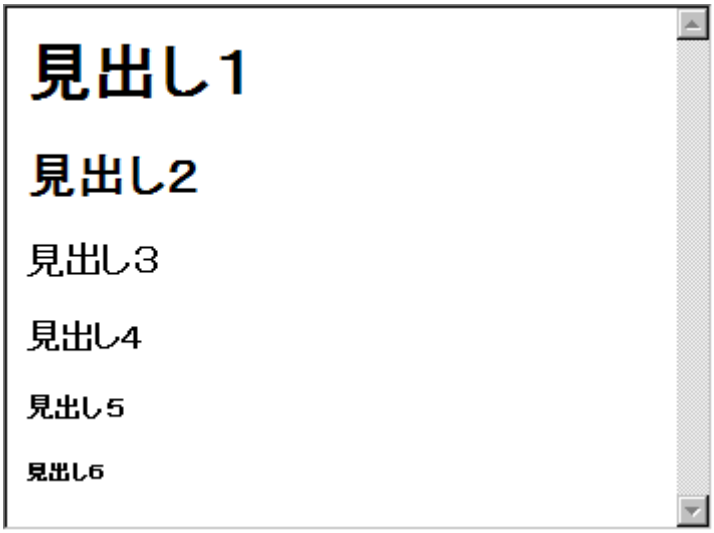
「見出し」をつくる

見出しを作るには、**hn要素**（nは1～6の数字）を使います。

```
<h1>...</h1>
<h2>...</h2>
<h3>...</h3>
<h4>...</h4>
<h5>...</h5>
<h6>...</h6>
```

見出しは、<h1>が大見出し、<h2>が中見出し、<h6>が最小の小見出しのように、6段階の見出しのレベルに分かれており、レベルが小さくなるほど、文字サイズが小さくなります。

ブラウザでの表示例



HTMLファイルでの例

```
<h1>自己紹介</h1>
<h2>プロフィール</h2>
<h3>リンク集</h3>
```

次へ進んでください。

「段落」と「改行」をつくる

空白や改行の扱い

HTML文書では、空白や改行の扱い方が変わっています。

- 空白
  - 半角スペース: 何文字入力しても、ブラウザでは一文字の空白文字として表示
  - 全角スペース: 入力した数だけの空白文字（全角）としてて表示
- 改行
  - 改行しても、ブラウザでは一文字の空白文字として表示

例えば、次のようなHTMLファイルは、

```
これは　　空白のテストです。
途中で改行しても
つづけて表示されます。
```

ブラウザでは、次のように表示されます。

```
これは 空白のテストです。 途中で改行しても つづけて表示されます。
```

「段落」をつくる

段落を作るには、**p要素**を使います。

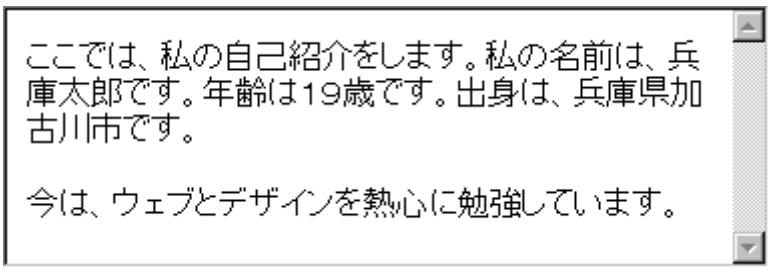
```
<p>...</p>
```

ここでの「段落」（パラグラフ）とは、「１つの考えや意味を説明する文章の集まり」と考えてください。日本語の段落とは違い、最初の文字が１文字字下げされません。

段落と段落の間には、大き目の空白ができます。

HTMLとブラウザでの表示例

```
例：
<p>
ここでは、私の自己紹介をします。
私の名前は、兵庫太郎です。年齢は１９歳です。
出身は、兵庫県加古川市です。
</p>
<p>今は、ウェブとデザインを熱心に勉強しています。</p>
```



「改行」をつくる

要素の中で、強制的に改行を作るには、**br要素**を使います。

```
...<br>
```

段落などの要素の中で、文章を強制的に改行させたい場合は、br要素をを使います。この要素には、終了タグがありません。

なお、この要素を使って、段落を表現することはやめてください。

```
<h2>プロフィール</h2>
名前は、○○です。<br>
出身地は○○です。<br>
所属は○○学科です。
```

次へ進んでください。

「リンク」をつくる

「リンク」をつくる

ほかのWebページにジャンプする、リンクを作るには、**a要素**と**href属性**を使います。

```
<a href="リンク先のURL">...</a>
```

a要素のhref属性に、リンク先のWebページのURLを指定します。また、ファイルを指定することもできます。

タグの間の内容には、リンク先のWebページのタイトルなど、リンクであることをあらわす具体的な言葉を設定します。「ここ」や「ここをクリックしてください」のような、抽象的な言葉は避けてください。

HTMLとブラウザでの表示例

```
<a href="http://arena.hyogo-dai.ac.jp/~kawano/">ウェブデザイン担当者のサイト</a><br>
<a href="http://www.hyogo-dai.ac.jp/">兵庫大学</a><br>
<a href="index.html">トップページへ</a>
```

次へ進んでください。

「水平線」をつくる

「水平線」をつくる

水平線（横罫線）を作るには、**hr要素**を使います。

```
<hr>
```

hr要素を入れた場所に、凹凸のある水平線をひきます。

さらに、**size, width属性**を使うと、線の太さや長さも設定できます。

```
<hr size="太さ" width="長さ">
```

size属性に、数値を設定すると、線の太さが変わります。単位は「ピクセル」（画面を構成する点）です。

width属性に、数値を設定すると、線の長さが変わります。こちらも、単位は「ピクセル」です。また、長さに「%」を設定すると、ウィンドウの横幅に対する割合で、長さが設定されます。

HTMLとブラウザでの表示例

```
<hr>
<hr size="10">
<hr size="10" width="20">
<hr size="10" width="80%">
```

次へ進んでください。

課題: ネット上の記事へのコメントでHTMLファイルを書く

今回から、HTMLファイルを作成していきます。HTMLファイルの内容は、**「インターネット上のあらゆる記事へのコメント」**とします。

1. インターネット上の記事にコメントをする

(1) 情報を集める

インターネットから**自分が興味を持ったニュース等の記事**を1つ探し出しなさい。

テーマは自由ですが、次の点に注意してください。

- 「ニュース」または「ブログ・日記」サイトからを取り上げる（掲示板は×）
- その記事は1つのURLで閲覧できる（1つのWebページにつき、記事は1つ）
- 例:
    - ニュース系: アサヒ・コム、NHKオンライン、Yahoo!ニュース、PC Watch、ESPN F1など
    - ブログ・日記系: アメーバブログ、エキサイトブログ、ココログ、livedoor Blog、はてなダイアリーなど

情報が見つけられたら、次の3つの情報をメモしておきなさい。

- WebページのURL
- Webページのタイトル
- Webページが作成（公開）された日時

(2) 情報にコメントをつける

記事を探したら、次の順に自分のコメントを作りなさい。なお、文字数は、**目安として「200文字以上」**としますが、内容はささいなことでかまいません。

1. 今日の日付を書く
2. **コメントにふさわしいタイトル**を自分で考える
3. コメントの内容を書く
    - 記事を書いた個人名または会社名、サイト名と作成（公開）された日付を書く
    - 記事のURLへのリンクを作る
    - 記事の内容を、**他人に内容がわかるように要約**する
    - 記事に対して、**自分の意見・批評・思ったことなど**を書く

サンプル：
2010年4月20日
中国GPをマクラーレンが制す

4月18日付けのFPSN F1の記事によると、
18日（日）日本時間16時から行われた、2010年F1世界選手権第4戦中国GPの決勝で、
マクラーレンのジェンソン・バトンが優勝した。
日本人ドライバーの小林可夢偉（BMWザウバー）は1週目に接触事故でリタイアとなった。

予選では、レッドブルの2人のドライバーがポールポジションと2番グリッドを獲得し、
フロントローを独占した。しかし、決勝では雨に見舞われたため、
接触事故の後セーフティーカーが2回導入されるなど、荒れたレースとなった。
結果として、天候の変化をうまくつかんだバトンがレース中盤からトップに立ち、
そのままフィニッシュした。チームメートのルイス・ハミルトンは2位に入り、
マクラーレンが1-2フィニッシュを飾った。

バトンは今季2勝目を挙げたが、いずれのレースも荒れたレースを制した形になっている。
昨年の経験がここに来て活かされているような気がする。
しかし、車の出来ではレッドブルやフェラーリのほうが上なので、これからまだまだ息を抜くことはできない。

2. HTMLファイルの作成

今回説明しているHTMLの技術を使って、考えたコメントをHTMLファイルを作成してください。

注意点は、次のとおりです。

- 情報源へのURLは、その情報源にアクセスできるようにリンクにすること
- 次にあげるHTMLの要素を、使用すること
    - html
    - head
    - title（注：課題のタイトル、h1と同じ内容）
    - body
    - h1（注：課題のタイトル、titleと同じ内容）
    - h2（コメントの日付）
    - h3（自分で考えたコメントのタイトル）
    - p（コメントの各段落）
    - br
    - a href
    - hr

3. 課題のタイトルとファイル名

- タイトル:「4月20日の課題」
- ファイル名: 次のとおりとします。
    - 「0420」＋「学籍番号」＋「.html」（半角文字で！）
      （学籍番号がC2092000の場合、ファイル名は 0420c2092000.html」）

提出方法

作成したファイルを、次の場所にコピーして提出してください。

- 「マイコンピュータ」→「Fsの資料・課題」→「kawano」→「提出」

提出期限

- 原則として、**平成22年4月27日9時30分**までとします。
- 期限以後は、特別な理由がない限り、提出を受け付けません。
- 課題のやりなおしは、提出期限まで受け付けます。