

# 船舶海洋情報学

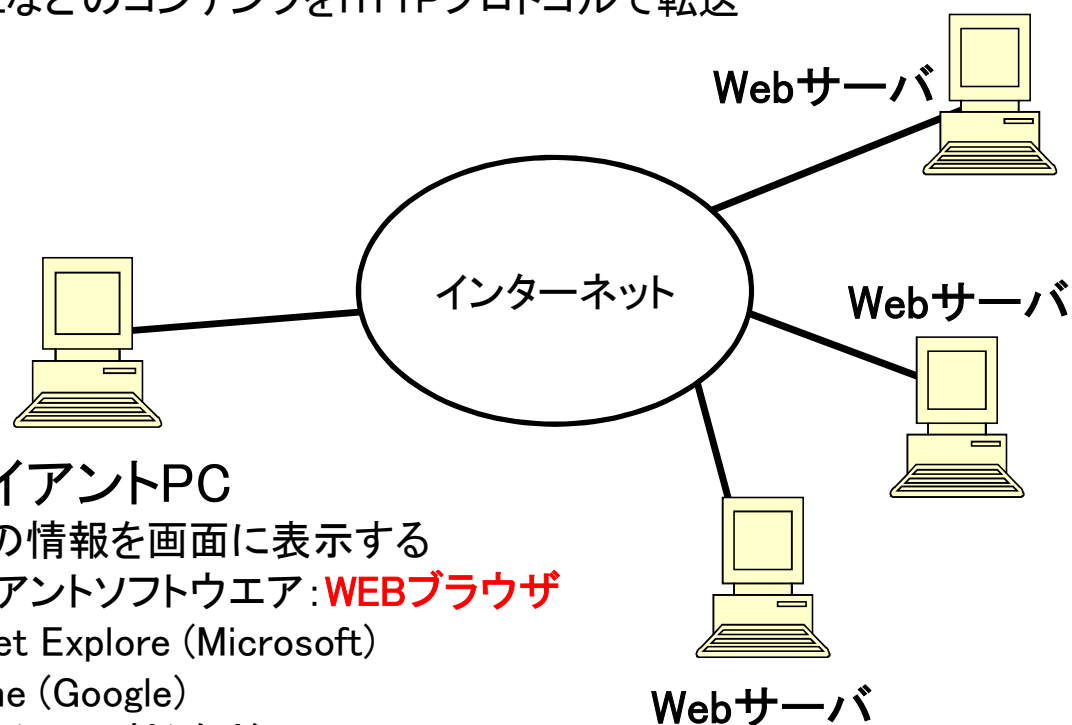
九州大学 工学府海洋システム工学専攻 講義資料 担当:木村

## 04. WWWとHTML



### WWW (World Wide Web)

インターネットの情報をハイパーテキスト形式で参照できる情報提供システム  
HTMLなどのコンテンツをHTTPプロトコルで転送



#### クライアントPC

WWWの情報を画面に表示する

クライアントソフトウェア: **WEBブラウザ**

Internet Explore (Microsoft)

Chrome (Google)

Safari (Apple社)など

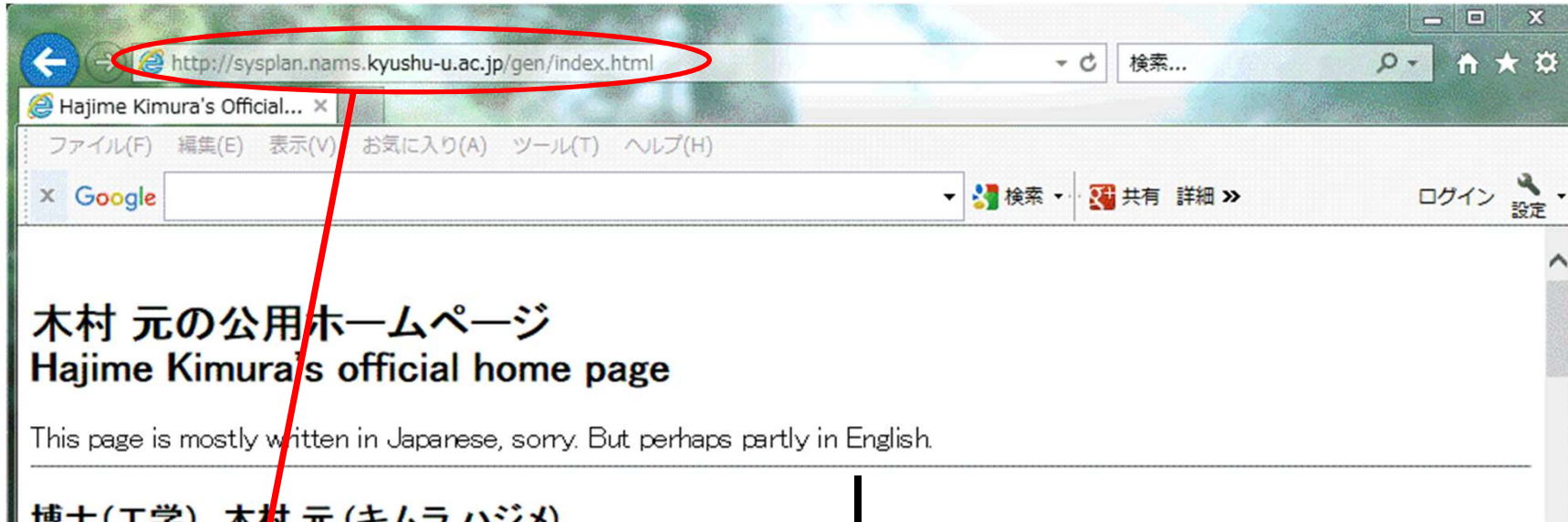
#### WWW で必要な3つの定義:

- 1) 情報のアクセス手段と位置  
**URI** (Uniform Resource Identifier)
- 2) 情報の表現フォーマット  
**HTML** (HyperText Markup Language)
- 3) 情報の転送などの操作の定義 (前回紹介)  
**HTTP** (HyperText Transfer Protocol)

# URI (Uniform Resource Identifier)

リソースを表す表記法(識別子)  
URL や メールアドレスなど

URL (Uniform Resource Locator)



HTTP://sysplan.nams.kyushu-u.ac.jp/gen/index.html

URIスキーム                      ホスト名                      パス                      リソース名

- HTTP://      Webサーバへ接続
- HTTPS://    Webサーバへ接続
- FTP://      サーバへ接続してファイル転送
- FILE://     ローカルホスト(自PC)へのファイルアクセス

IPアドレスでもOK

以下のような書式も可能

HTTP://ホスト名:ポート番号/パス/リソース名

HTTP://ホスト名/パス/?問い合わせ内容#部分情報

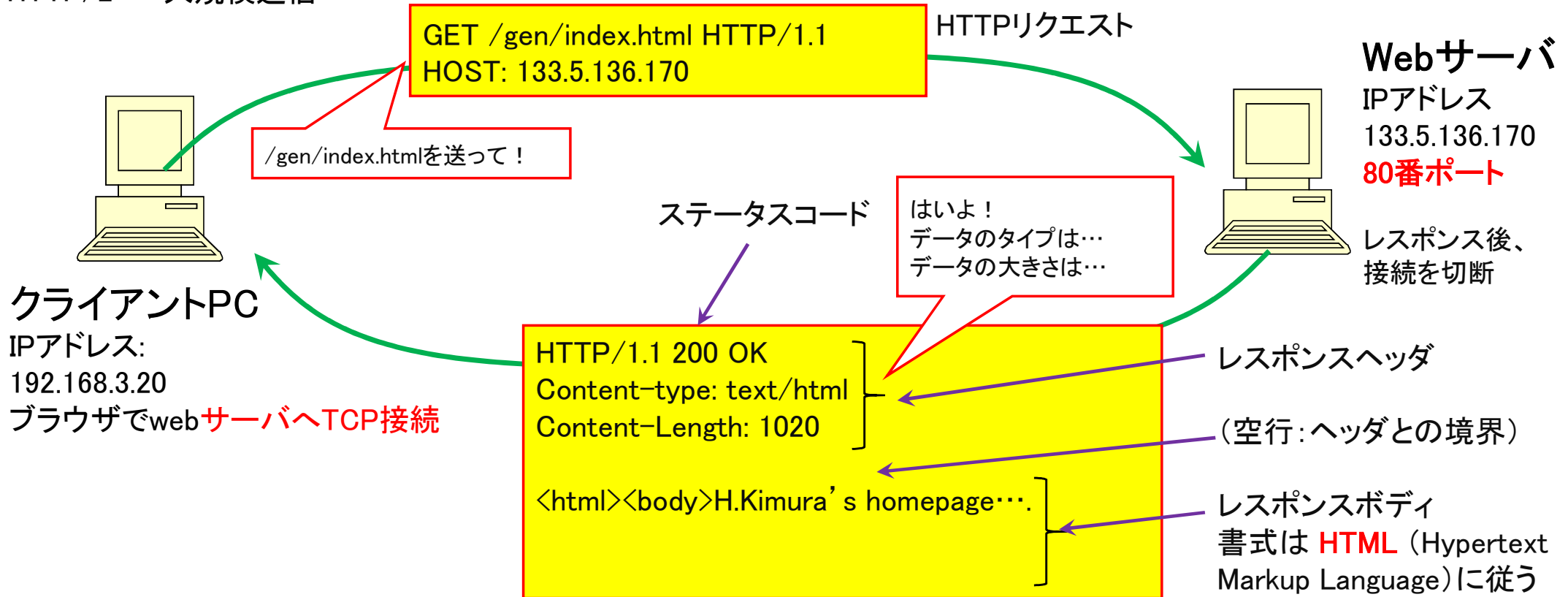
# HTTP (HyperText Transfer Protocol) 前回紹介

HTMLなどのコンテンツの送受信に用いられるリクエスト-レスポンス型の通信プロトコル

ポート番号80をデフォルトとして使用

HTTP/1.1 広く利用されている

HTTP/2 大規模通信



# HTML (HyperText Markup Language)

- ・ Webページを記述するための言語(データ形式)
- ・ 「タグ」を用いて表示する要素やブラウザに対する命令を意味付け
- ・ テキストファイルとして作成し、拡張子を「.html」とする

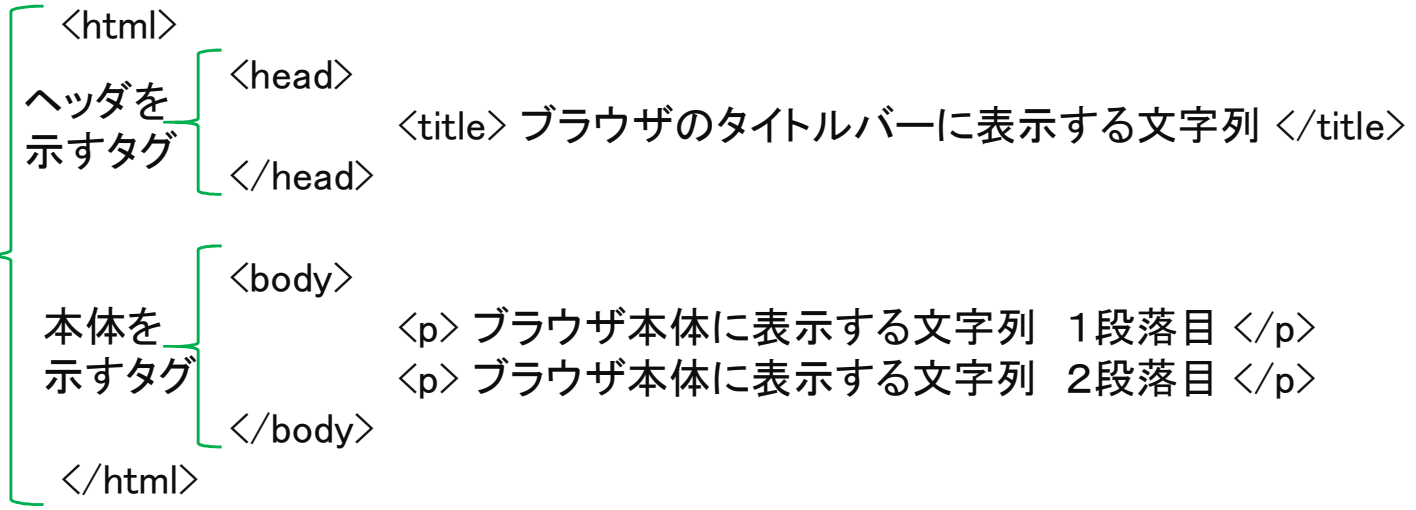
タグには大文字/小文字の区別なし  
タグ中の属性で示すリソースのパスやファイル名には大文字/小文字の区別がある

## HTMLファイルの構成

文書型宣言: HTMLのバージョン等を示す省略可

```
<!DOCTYPE HTML PUBLIC “./W3C//DTD HTML 4.01 Transitional//EN”  
HTTP://www.w3.org/TR/html4/loose.dtd>
```

HTML文書の始まりと終わりを示すタグ



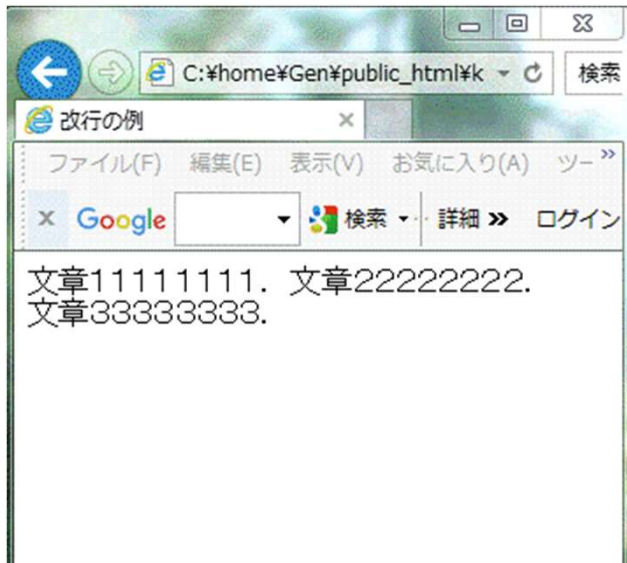
# HTMLのタグ一覧(1)

改行 <br>

単独のタグ

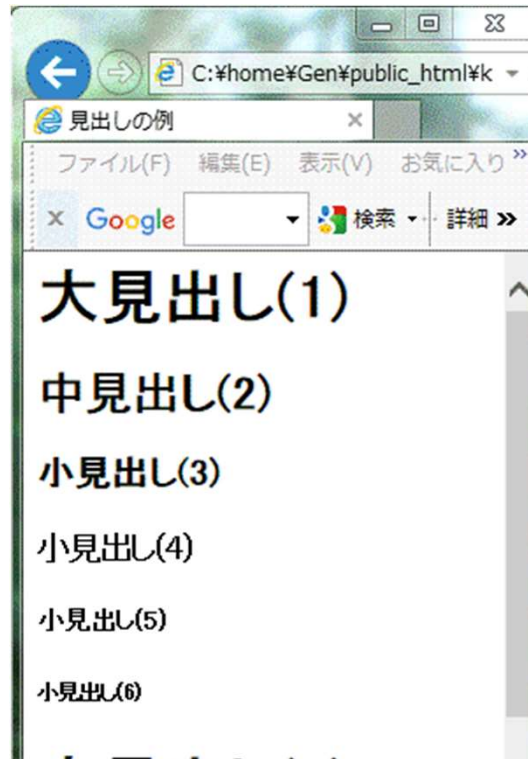
文章11111111.  
文章22222222. <br>  
文章33333333.

Htmlテキスト中の  
改行は無視される



見出し

<h1> 大見出し(1) </h1>  
<h2> 中見出し(2) </h2>  
<h3> 小見出し(3) </h3>  
<h4> 小見出し(4) </h4>  
<h5> 小見出し(5) </h5>  
<h6> 小見出し(6) </h6>



マーク付きリスト

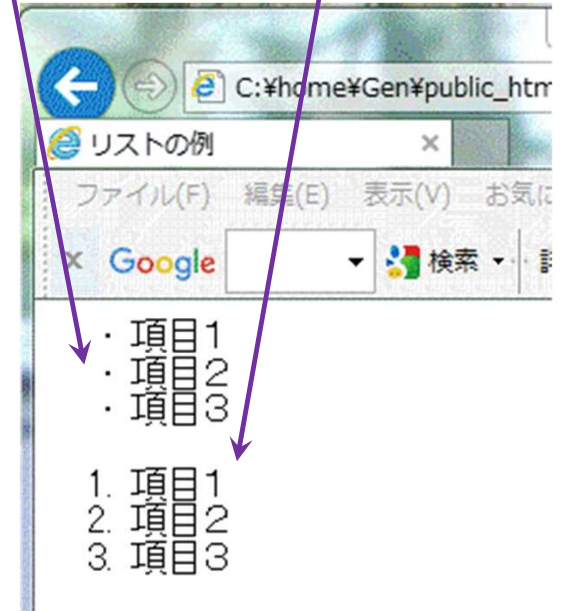
<UL>  
<LI> 項目1  
<LI> 項目2  
<LI> 項目3  
</UL>

「入れ子」構造も可

番号付きリスト

<OL>  
<LI> 項目1  
<LI> 項目2  
<LI> 項目3  
</OL>

「入れ子」構造も可



## HTMLのタグ一覧(2)

### 画像を置く

単独のタグ

``

属性(attribute)

例)

``

相対パス

絶対パス  
他のホストのリソース

``

### 大きさを指定して画像を置く

``

例)

``

``





# HTMLのタグ一覧(3)

## リンクを張る

<a>タグで囲まれた文字列を青色で下線を引き、  
クリックするとhref属性で示したページ(リソース)へ移動する

<a href="/パス/リンク先リソース名"> リンク先に関連する文字列 </a>

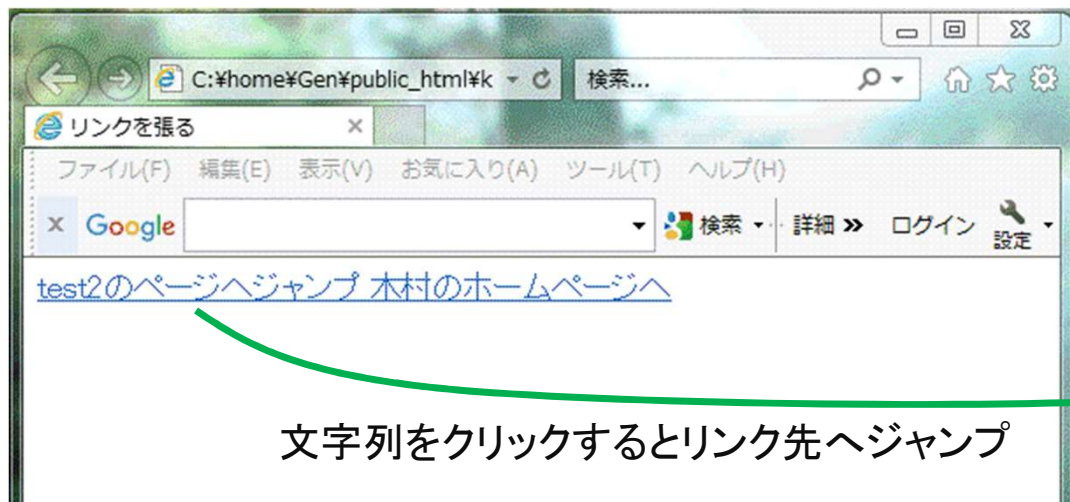
属性(attribute)

相対パス

絶対パス  
他のホストのリソース

例)  
<a href="./abc/test2.html"> test2のページへジャンプ </a>

<a href="http://sysplan.nams.kyushu-u.ac.jp/gen/index.html"> 木村のホームページへ </a>



文字列をクリックするとリンク先へジャンプ

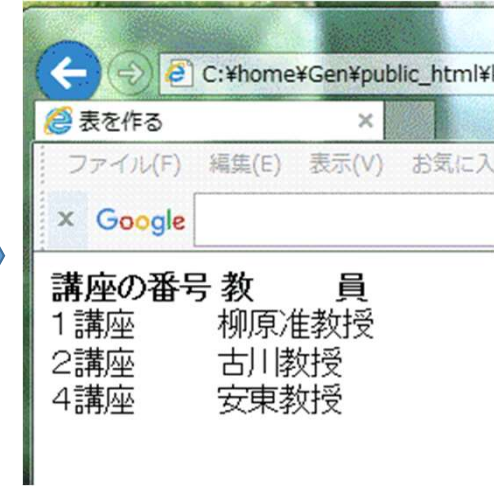
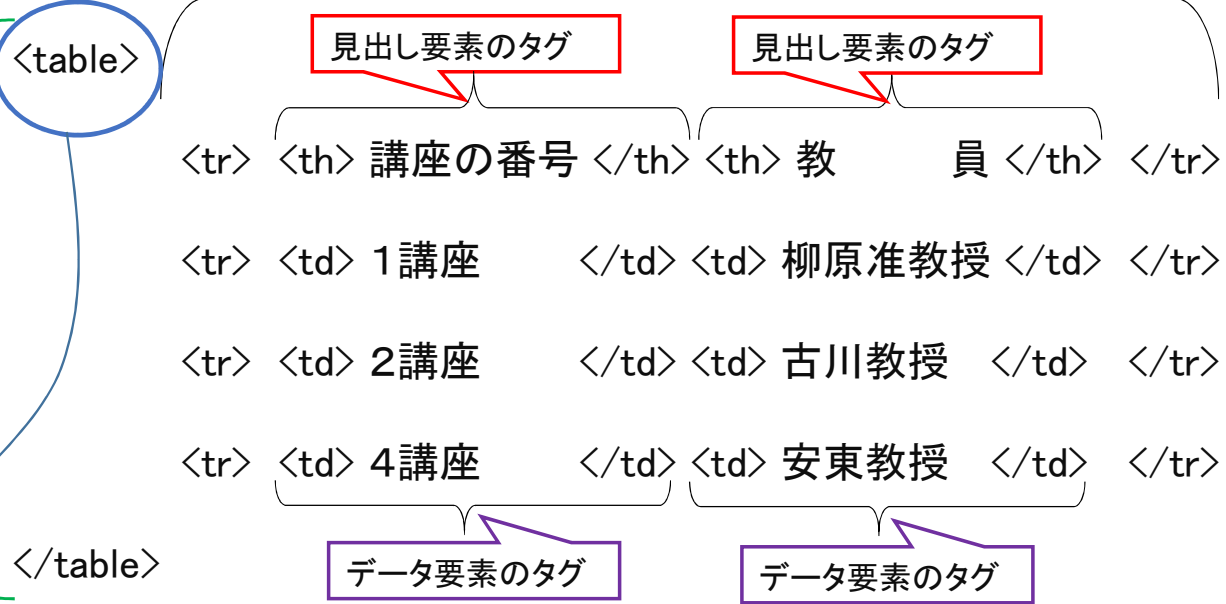


# HTMLのタグ一覧(4)

<tr>...</tr>のタグで囲まれた部分が表の1行分

## 表を作る

表の始まりと  
終わりを示すタグ



<table border> のような属性にすると  
罫線が入る





## HTMLのタグ一覧(5)

### 文字フォントの色や大きさを指定する

指定したい文字列を<font> </font>タグで囲む

<font color="green" size="4"> 文字列文字列文字列 </font>

属性 colorで色を指定  
red, blue, black など

属性 size で大きさを指定  
1から7までの数字、または  
数値の前に+や-を付けて  
相対値で指定することも可

### 水平な線を引く

<hr>

単独のタグ

### 文字列を太字で強調する

<strong> 文字列文字列文字列 </strong>

### 文字列を斜体で強調する

<em> 文字列文字列文字列 </em>

### 物理スタイルタグ

<b> 文字列文字列 </b>	: 太文字
<i> 文字列文字列 </i>	: イタリック(斜体)
<tt> 文字列文字列 </tt>	: 等幅(タイプライタ)
<sup> 文字列文字列 </sup>	: 上付き添え字
<sub> 文字列文字列 </sub>	: 下付き添え字
<u> 文字列文字列 </u>	: アンダーライン
<s> 文字列文字列 </s>	: 取り消し線

# CGI (Common Gateway Interface) ブラウザからwebサーバ側のプログラムを実行

単なる「静的なデータの閲覧」ではなく、**状況に応じた反応**を返す仕組み(1)

HTMLで簡単に記述 (サーバ側はプログラム言語とサーバソフトに依存)

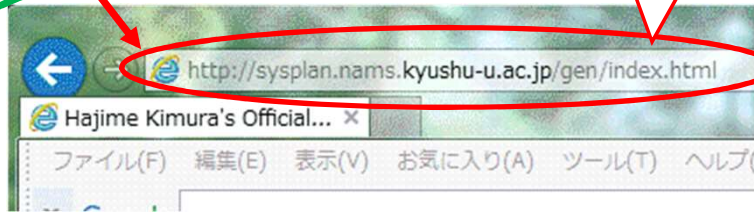
**URL** (Uniform Resource Locator)  
に直接プログラム名を指示  
通常はHTMLの<a>タグのリンク先  
として呼び出す

必要に応じて、プログラム名の後ろに  
?変数名=値&変数名=値&…の形式  
でパラメータを渡す

実際にはブラウザのURLに直接このようなパラメータを  
手入力することは現実的ではない(<a>タグも同様)

ユーザがブラウザ上で簡単にパラメータを入力してCGI  
を呼び出せる **<form></form>**タグがある

HTTPリクエスト文字列



クライアントPC

HTTPレスポンス文字列

WEBサーバ

指示されたプログラムを実行  
(**サーバ上で動作する**)  
プログラムの標準出力(文字列)が  
そのままクライアントPCへの応答となる

HTMLテキスト  
画像データ(MIMEでエンコードされたテキストデータ)  
など

webページにアクセスカウンタの数字画像表示を埋め込む場合  
HTML文書の埋め込みたい場所に例えば

```

```

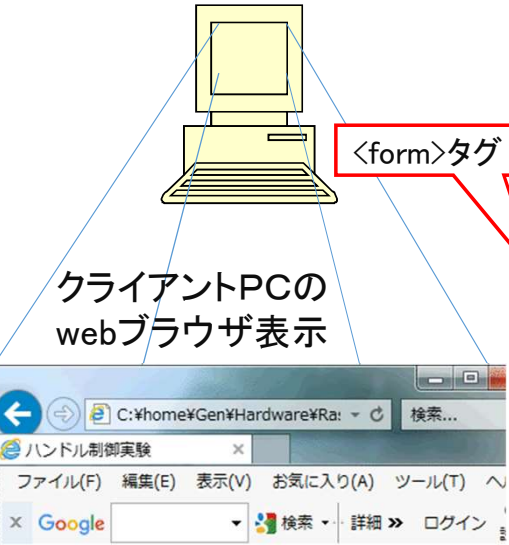
のようにすればよい(cgi実行結果の画像データが埋め込まれる)

# CGI (Common Gateway Interface) の例

フォーム入力とCGIプログラム起動を実現するweb画面のHTML

```
<html>
<head>
<meta http-equiv=content-type content="text/html; charset=UTF-8">
<title> ハンドル制御実験 </title>
</head>
<body>
<h1> 台車のハンドル制御実験装置 WEB操作パネル </h1>
<form method="POST" action="start.py" >
  ハンドル角度制御フィードバックゲインの設定<Br>
  <input type="text" name="xGain" size="6" value="0" style="font-size:200%;"> 位置偏差フィードバックゲイン k1<Br>
  <input type="text" name="thetaGain" size="6" value="0" style="font-size:200%;"> 角度偏差フィードバックゲイン k2<Br>
  <Br>
  <select name="moveDirection" size="2">
    <option value="forward" selected> 前進
    <option value="backward"> 後進
  </select> 台車の進行方向 (ステアリング車輪がある方が前) <Br>
  <BR>
  <input type="submit" value="START">
</form>
</body>
</html>
```

HTTPのPOSTリクエストを使いサーバーのCGIプログラム start.py を起動する



クライアントPCのwebブラウザ表示  
台車のハンドル制御実験装置WEB操作パネル

ハンドル角度制御フィードバックゲインの設定

0	位置偏差フィードバックゲイン k1
0	角度偏差フィードバックゲイン k2

前進  
後進 台車の進行方向(ステアリング車輪がある方が前)

START

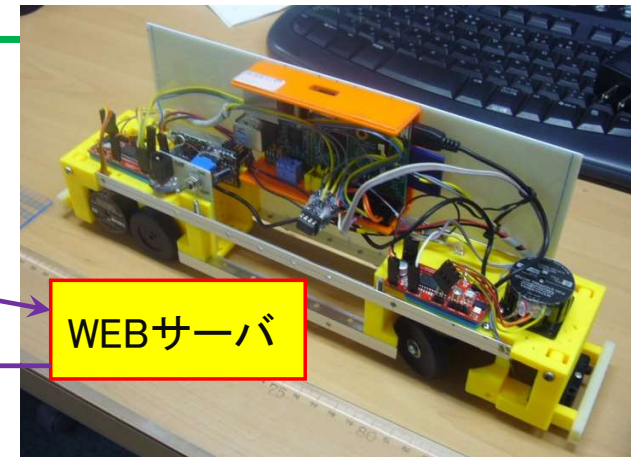
フォーム入力後、ボタンを押すとCGIプログラム起動

HTTPリクエスト文字列

クライアントPC  
ブラウザに表示

CGIプログラム  
実行結果  
HTMLテキスト

WEBサーバ



# JavaScript

HTMLにプログラムを埋め込んで状況に応じて反応する仕組み(2)

**クライアントのブラウザ上でプログラムを実行**

(CGI はサーバ側で実行している違いに注意)

```
<html>
<head>
<meta http-equiv=content-type content="text/html; charset=UTF-8">
</head>
<body>

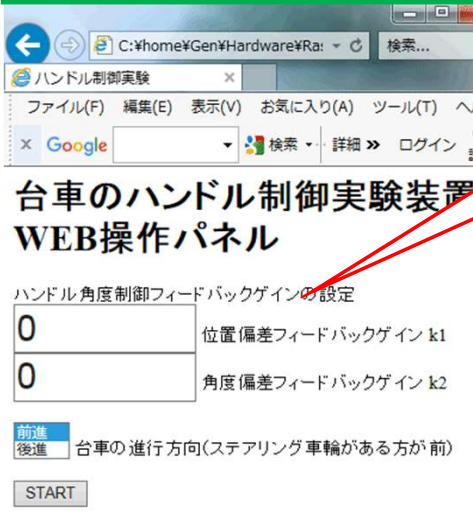
<script type="text/javascript">
  //ここにjavascriptを書きます。
</script>

</body>
</html>
```

インラインスクリプト:  
HTMLに直接埋め込む場合

```
<html>
<head>
  <script type="text/javascript" src="sample.js"></script>
</head>
<body>
</body>
</html>
```

外部スクリプトをHTML内で読み込む



ボタンを押したり  
マウスを合わせたり  
文字をクリックしたときに  
JavaScriptを実行できる

フォームに入力されたデータの計算やチェック  
サーバと通信しながら**リアルタイムに情報表示**  
(通常は「更新」ボタンを押したときのみ情報更新)  
ユーザのマウスやキーボード操作に合わせて画像を動かすなどが可能

## まとめ

- (1) 情報のアクセス手段と位置  
**URI** (Uniform Resource Identifier)
- (2) 情報の表現フォーマット  
**HTML** (HyperText Markup Language)  
タグいろいろ
- (3) **CGI**: ブラウザから**サーバのプログラム**を実行する仕組み
- (4) javascript: HTMLに埋め込んで**ブラウザ上でプログラム**を実行

## レポート課題：

自分のホームページをイメージしたHTMLを作成し、ブラウザで表示して確認せよ。  
少なくとも2つ以上のHTMLファイルで構成し、互いのページにリンクしてジャンプできること。  
ページには画像を埋め込むこと。(CGIはサーバが無いと無理なので考えなくて良い)  
各自なるべく魅力的なコンテンツを用意すること。**ページや画像へのリンクは相対パスを使うこと。**

【提出方法】 上記のHTMLファイルと画像データなどをディレクトリ(フォルダ)ごとまとめてzipで圧縮し、  
下記の課題提出用フォルダへ、課題の番号と提出者が分かるようにファイル名を以下のようにしてアップロードせよ  
第4回2TE19xxxZ名前.zip

<https://share.iii.kyushu-u.ac.jp/public/hROwAAqIPi5ATI4BUXJtIJeJbMLJzVszfitNI89GHcMK>

上記フォルダへのリンクは下記ホームページから

<http://sysplan.nams.kyushu-u.ac.jp/gen/edu/NavallInformationProcessing/2019/index.html>