

Slide URL

<https://vu5.sfc.keio.ac.jp/slide/>

Web情報システム構成法 第3回 HTML入門(2)

萩野 達也 (hagino@sfc.keio.ac.jp)

前回課題：簡単なHTMLページの作成

- ▶ 自分が好きな海外の町を紹介するトップページを作成しなさい。
 - ▶ HTML5で記述すること.
 - ▶ HTMLだけを用い, CSSやJavascriptなどは用いないこと.
 - ▶ テキストエディタ(テキストエディタ, メモ帳, atom, miなど)で作成しなさい.
 - ▶ マークアップのコピペしないように. 手で打ち込みましょう. その方が覚えられます.
 - ▶ town.html のような名前で作成し, ブラウザで直接ファイルを開いて確認しなさい.
- ▶ トップページに書く内容
 - ▶ 町の名前
 - ▶ 町の紹介
 - ▶ 町の観光名所などを箇条書きにする
- ▶ 利用する要素
 - ▶ h1などのヘッダ要素を使いなさい.
 - ▶ 箇条書きを入れなさい.
 - ▶ emで協調をどこかに入れなさい.
 - ▶ などのHTMLエンティティをどこかに使いなさい.
- ▶ 提出
 - ▶ 作成したHTMLファイルを直接提出してください.
 - ▶ 締め切り: 今週土曜日

テキストエディタ

▶ ファイルにはテキストファイルとバイナリファイルがある

▶ テキストファイル

- ▶ 人が直接中身を見て理解できるファイル
- ▶ 自然言語で使っている文字(アルファベット, 漢字など)で書かれている

▶ バイナリファイル

- ▶ 人が直接中身を見てもよくわからないファイル
- ▶ アプリケーションを経由して中身を見る必要がある
- ▶ 画像や音楽, Word文書, PDF, アプリケーション自身など

▶ テキストエディタ

- ▶ テキストを編集するためのアプリケーション
- ▶ どんなテキストでも編集できる, 汎用のアプリケーション

▶ Windowsの標準

- ▶ メモ帳(notepad)

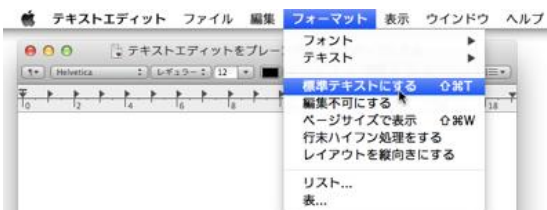
▶ Mac OS Xの標準

- ▶ テキストエディット

▶ 自分でインストール

- ▶ emacs
- ▶ atom
- ▶ vim
- ▶ mi

「フォーマット」を「標準テキストにする」



テキスト形式にもいろいろある

▶ plain text

- ▶ 装飾もなにもないテキスト形式
- ▶ 通常のメールの本文に用いられる
- ▶ Macのテキストエディットでは「標準テキスト」のこと

▶ rich text

- ▶ 装飾などを含んだテキスト形式
- ▶ メールでの簡単な装飾にも使われる
- ▶ Macのテキストエディットのデフォルト

▶ html

- ▶ HTMLもテキスト形式

▶ その他

- ▶ TeX: 文書清書システムで用いられる
- ▶ C, javascript: プログラムもテキスト形式で書く
- ▶ CSV: 表計算アプリで利用

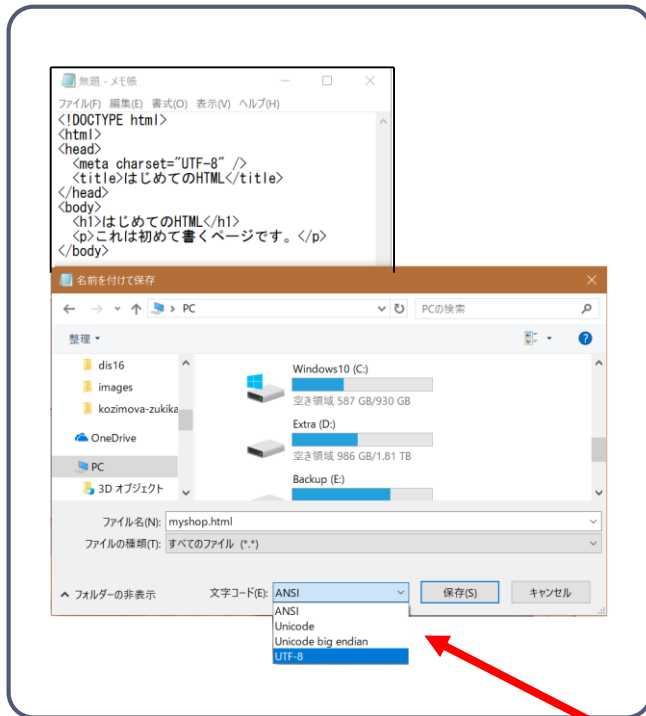
MIME

- データの形式のやり取りで指定
- 電子メールで添付文書の利用に
- Multipurpose Internet Mail Extensions

mimeタイプ	拡張子	意味
text/plain	.txt など	汎用テキスト形式
text/rtf	.rtf	リッチテキスト
text/html	.html	HTMLファイル
text/css	.css	CSSファイル
text/javascript	.js	JavaScriptファイル
text/csv	.csv	CSVファイル
text/xml	.xml	XMLファイル

文字コードとその指定

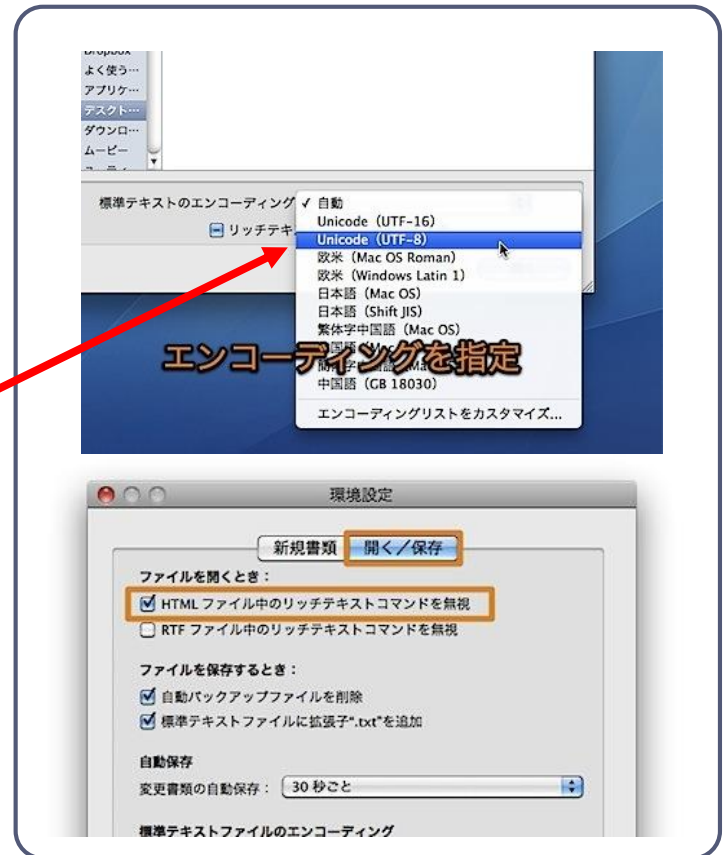
- ▶ テキストエディタでテキスト作成するときに指定する
 - ▶ 保存するときにオプションメニューとしてあることが多い



Windows メモ帳

Mac テキストエディット

UTF-8とUTF-16は
同じUnicodeでも
エンコードが異なる



ANSIとするとShift_JISになる
UnicodeとするとUTF-16になる

文字コードとは

▶ 文字をどのような数字(2進数)で表現するのか

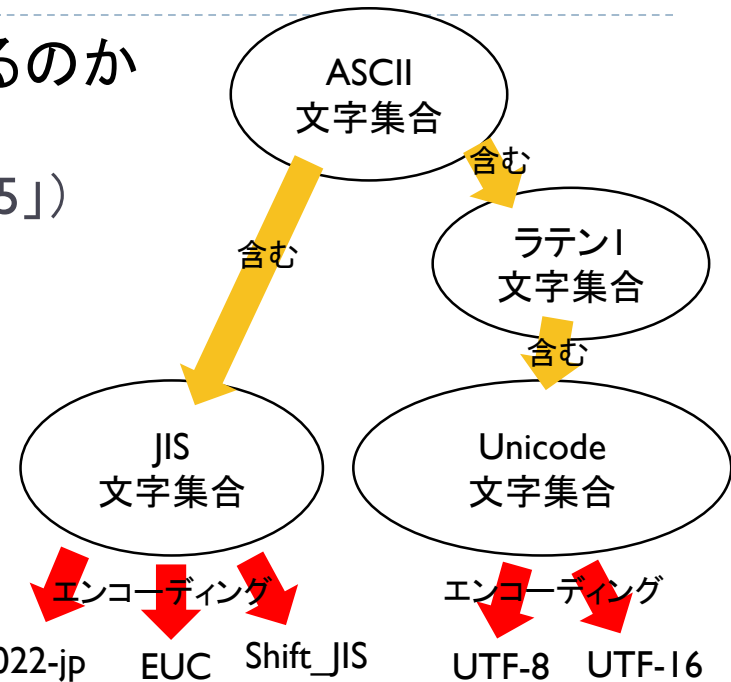
- ▶ コンピュータの中はすべて数字
- ▶ 「A」を「01000001₂」で表す(10進数では「65」)

▶ 文字集合とエンコーディング

- ▶ 文字集合: どのような文字を使うのか
 - ▶ JIS文字集合(第1水準, 第2水準, 外字)
 - ▶ Unicode

▶ エンコーディング: どのように数字で表すのか

- ▶ JIS
 - iso-2022-jp
 - EUC
 - Shift_JIS あるいは CP932
- ▶ Unicode
 - UTF-8
 - UTF-16
 - UTF-32



UTF-8

Unicode	ビット列
0000~007F	0xxxxxxxx
0080~07FF	110yyyyy 10xxxxxx
0800~FFFF	1110zzzz 10yyyyyy 10xxxxxx

Unicodeファイルには先頭にBOM (Byte Order Mark) がついたものがある。
UTF-16では推奨されるが, UTF-8では必須でも推奨でもない。

よくある間違い

- ▶ html は一つだけ
 - ▶ 2つ書いてはいけません.
 - ▶ コピペしてませんか？
 - ▶ body や head も一つだけです.
- ▶ 全角の空白をあまり用いない
 - ▶ タグ内の属性を区切る空白として用いない.
 - ▶ 文字要素を用いてはいけなところにも利用できません.
 - ▶ を用いましょう.
- ▶ 言語指定は正しく
 - ▶ 日本語は ja , 英語は en
 - ▶ jpは国
- ▶ ul 要素の子要素は li のみです.
 - ▶ p, dd, ol など置くことができません.
- ▶ font で文字の大きさなどの指定はしない.
 - ▶ font タグは廃止されています.
- ▶ ヘッダタグを正しく使いましょう.
 - ▶ h1 が最も外側のヘッダです
 - ▶ h1 の下のヘッダは h2 です.
 - ▶ h2 を飛ばして h3 を使ったり, 最も外側が h2 から始まるなどはいけません.

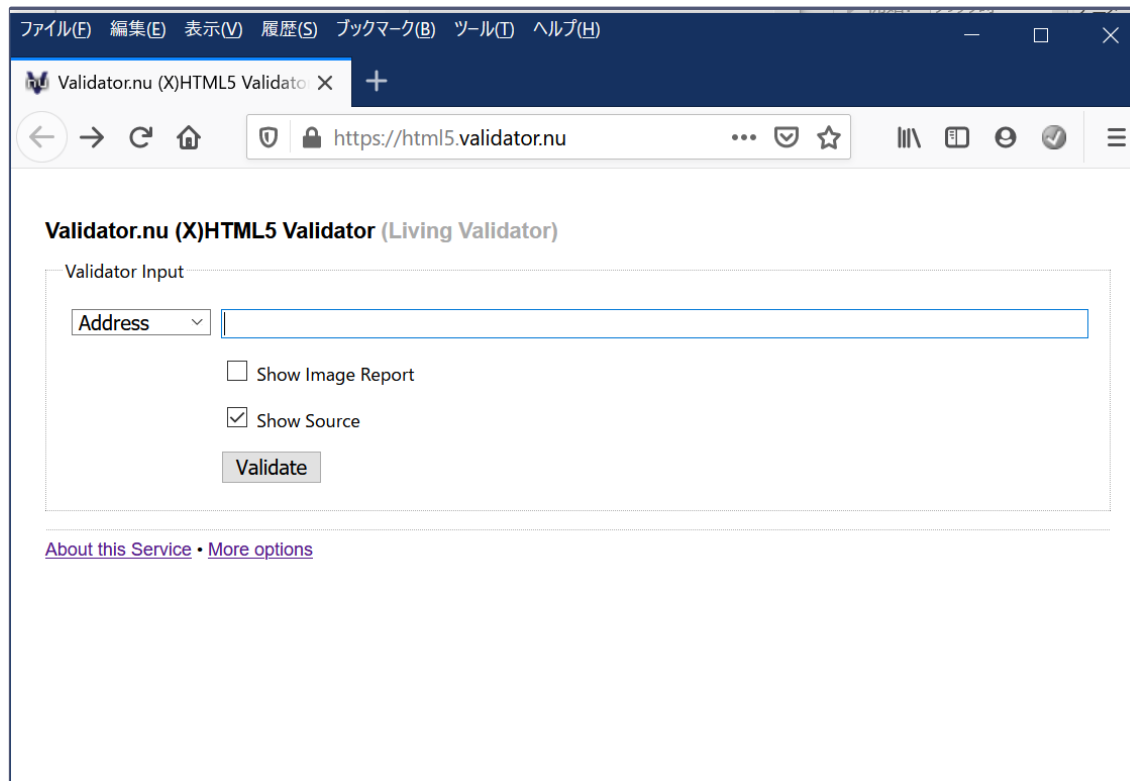
文法違反

```
<p>以下の項目があります.</p>
<ul>以下の項目があります.
  <li>1つ目の項目</li>
  <p>1つ目の説明</p>
  <li>2つ目の項目<br>
    2つ目の説明</li>
</ul>
```

HTMLの文法のチェック

▶ HTML5 Validator

- ▶ URL, ファイル, 直接入力で文法チェック
- ▶ <https://validator.w3.org/nu/>



課題 1：簡単なHTMLページの作成

- ▶ 前回作成した「自分が好きな海外の町」のページを以下の点に注意して正しくしなさい.
 - ▶ 文法的な間違いを修正する。Validatorでチェックすること。
 - ▶ 箇条書きを正しく行う。
 - ▶ ヘッダのレベルを正しく使う。
 - ▶ 言語の指定を正しく行う。
 - ▶ 文字エンコーディングを正しく指定する。
- ▶ 提出
 - ▶ 修正したHTMLファイルを直接提出してください。
 - ▶ 内容の更新を行う必要はありません。
 - ▶ 締め切り：今週土曜日

town.html

```
<!DOCTYPE html>
```

HTML5であることを宣言

```
<html lang="ja">
```

日本語で本文を記述

```
<head>
```

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

文字コードはUTF-8

```
  <title>私の好きな街</title>
```

文書のタイトル

```
</head>
```

```
<body>
```

```
  <h1>エジンバラ</h1>
```

ヘッダ

```
  <p>寒い&amp;暗いところですが, 文化的です. </p>
```

```
  <h2>観光名所</h2>
```

```
  <ul>
```

本文

```
    <li>エジンバラ城</li>
```

```
    <li>ホーリールード宮殿</li>
```

```
  </ul>
```

箇条書き

```
</body>
```

```
</html>
```

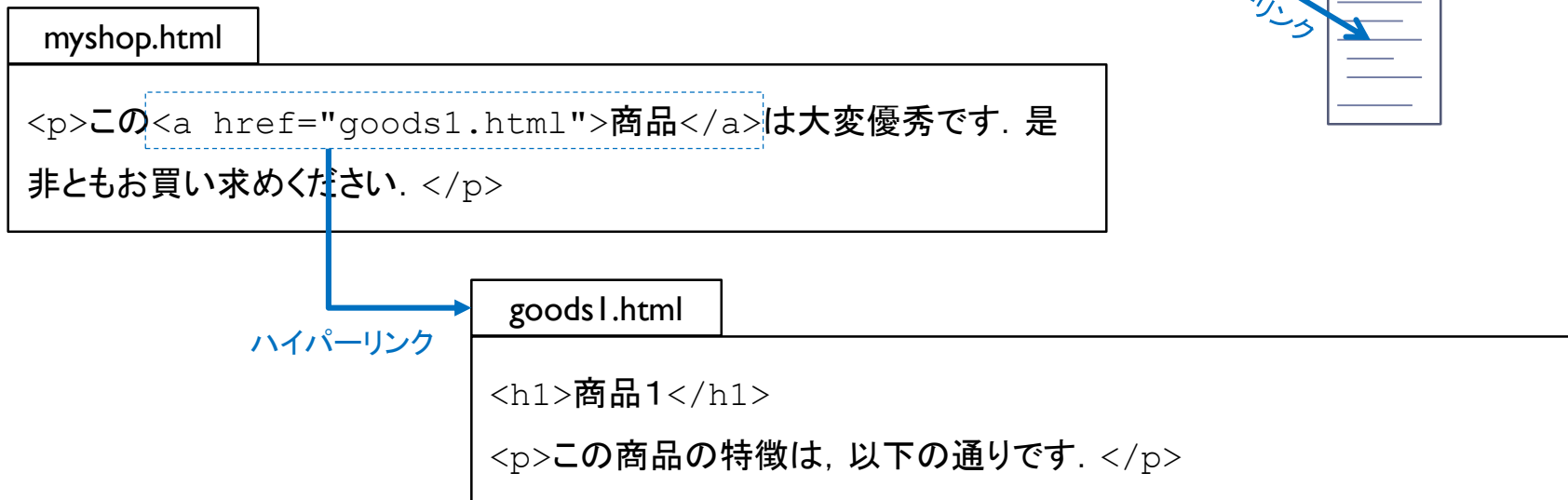
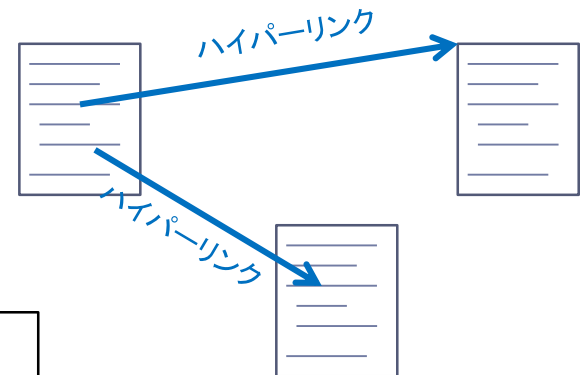
複数のWebページの作成

- ▶ Webサイトは複数のWebページから構成されている。
 - ▶ 全体の構成を考える必要がある。
- ▶ コンテンツマトリックス
 - ▶ どのようなページを用意するのか
 - ▶ 利用者のページの流れを考えて配置する

URL	タイトル	キーワード	内容
town.html	トップページ	概要	街の概要, トップページ
history.html	街の歴史	歴史	街の歴史
parks.html	公園の紹介	公園	街の公園の紹介
stat.html	統計情報	統計	人口, 面積などの統計情報
shops.html	お店	店	街にある店の紹介
help.html	問い合わせ	ヘルプ	問い合わせ
access.html	アクセス	アクセス	街へのアクセス

ページ間の移動

- ▶ ページ間はハイパーリンクにより接続する
 - ▶ リンクをクリックすることで移動する
- ▶ ハイパーリンクの指定
 - ▶ a タグ (要素) を使う
 - ▶ a は anchor (錨) のこと
 - ▶ ブラウザでは青く下線が引かれたりする



メニューもハイパーリンク

▶ メニューは別のページに移動するものなのでハイパーリンクです。

▶ ul を使ったメニューの作成

先頭省略

```
<body>
```

```
  <h1>わが街</h1>
```

```
  <p>街の魅力的な場所を紹介します. </p>
```

```
  <ul>
```

```
    <li><a href="castle.html">城</a></li>
```

```
    <li><a href="mall.html">ショッピングモール</a></li>
```

```
    <li><a href="help.html">ヘルプ</a></li>
```

```
  </ul>
```

```
</body>
```

メニュー

URL

▶ ハイパーリンクで別のページを指すために使っているのがURL

▶ `商品1`

▶ URL

▶ Uniform Resource Locator

▶ Web上の資源の場所を一意的に指し示す

`http://www.sfc.keio.ac.jp/teacher/hagino.html?title=web#lecture`

スキーマ	オーソリティ	パス	問い合わせ	フラグメント
• プロトコル	• ホスト名 • サーバ名	• オーソリティ内での位置 • ファイル名		• 文書内の位置

▶ 絶対URL

▶ スキーマから始まるURL

▶ 相対URL

▶ スキーマやオーソリティが省略されたURL

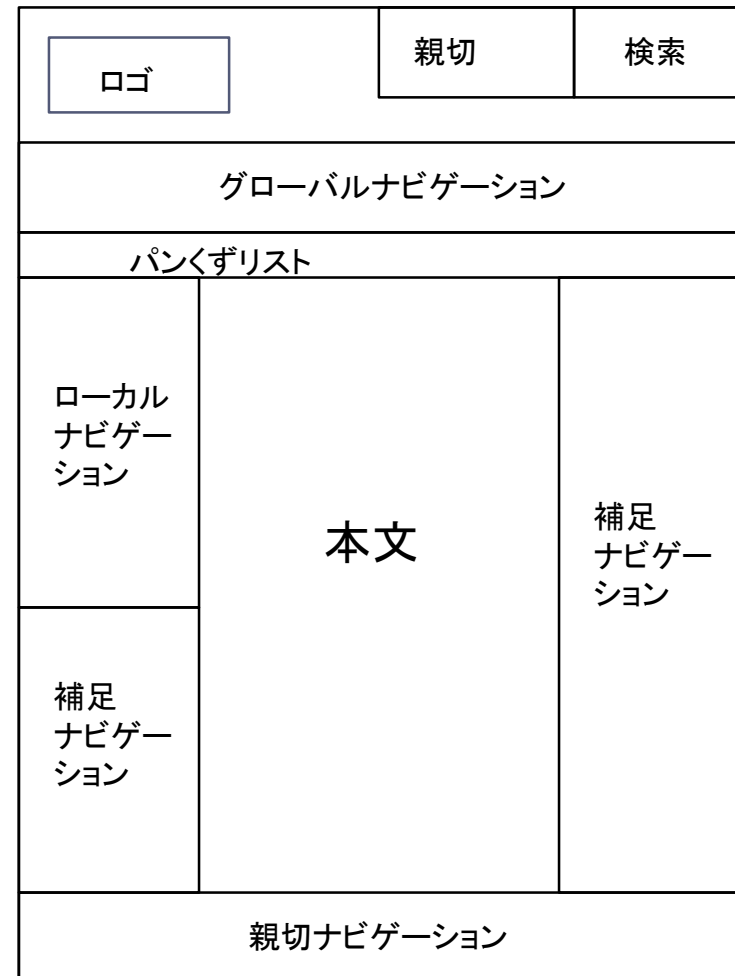
▶ /ではじまれば, 文書と同じオーソリティ内の絶対パス

▶ そうでなければ, 文書と同じ位置からの相対パス

ナビゲーション

- ▶ ナビゲーション
 - ▶ グローバルナビゲーション
 - ▶ ローカルナビゲーション
 - ▶ 補足ナビゲーション
 - ▶ コンテキストナビゲーション
 - ▶ 親切ナビゲーション
 - ▶ パンくずリスト
- ▶ サイトマップ
- ▶ サイトインデックス

典型的なページ



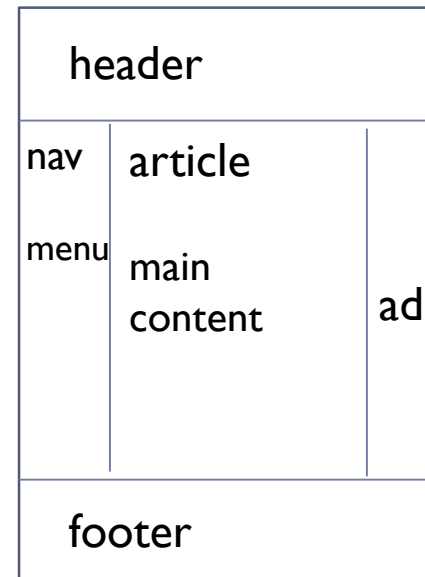
ナビゲーション詳細

- ▶ グローバルナビゲーション
 - ▶ サイト全体を大きく切り替える
 - ▶ すべてのページで共通
 - ▶ 7つ程度まで
- ▶ ローカルナビゲーション
 - ▶ それぞれのカテゴリ内での切替
- ▶ 補足ナビゲーション
 - ▶ ローカルで不足する部分
- ▶ コンテキストナビゲーション
 - ▶ 複数のページの場合に、次のページへ遷移するなど
- ▶ 親切ナビゲーション
 - ▶ ヘルプ
 - ▶ 検索
 - ▶ 問い合わせ先
- ▶ パンくずリスト
 - ▶ サイト内での現在の位置
 - ▶ 迷子にならないようにする
- ▶ サイトマップ
 - ▶ サイト全体の構成を表示

HTML5の基本的なbody構造

- ▶ HTML5では標準のページの構造を表す要素がある

```
<body>
  <header>
    header
  </header>
  <nav>
    menu
  </nav>
  <article>
    main content
  </article>
  <aside>
    advertisement
  </aside>
  <footer>
    about
  </footer>
</body>
```



HTML5の意味的区分要素

- ▶ `<div>～～</div>`
 - ▶ 内容に意味を与えない
- ▶ 意味的な区分を与える要素
 - ▶ `<nav>～～</nav>`
 - ▶ ナビゲーションリンクを入れる
 - ▶ リンク集などを入れるものではない
 - ▶ `<article>～～</article>`
 - ▶ 自己完結型のコンテンツ
 - ▶ その部分だけを取り出しても意味がある
 - ▶ `<section>～～</section>`
 - ▶ 意味的に関連したコンテンツ
 - ▶ 見出しをつけるべき
 - ▶ `<aside>～～</aside>`
 - ▶ 本文に属しないコンテンツ
 - ▶ 広告など
 - ▶ `<header>～～</header>`
 - ▶ ロゴ, タイトル, ナビゲーション
 - ▶ article, section, nav, asideの見出しや著者名
 - ▶ `<footer>～～</footer>`
 - ▶ 著作権情報, 法的事項
 - ▶ article, section, nav, asideの公開日やライセンス情報

HTML5によるtown.html

```
<!DOCTYPE html>
<html lang="ja">
<head>
  <meta charset="UTF-8">
  <title>私の好きな街</title>
</head>
<body>
  <header>
    <h1>エジンバラ</h1>
  </header>
  <nav>
    <ul>
      <li><a href="castle.html">エジンバラ城</a></li>
    </ul>
  </nav>
  <article>
    <h2>観光施設</h2>
    <p>エジンバラの魅力ある観光施設などを紹介します.</p>
  </article>
  <footer>
    <p>問い合わせ: <a href="help.html">ヘルプ</a></p>
  </footer>
</body>
</html>
```

← ページ全体のヘッダ

← メニュー

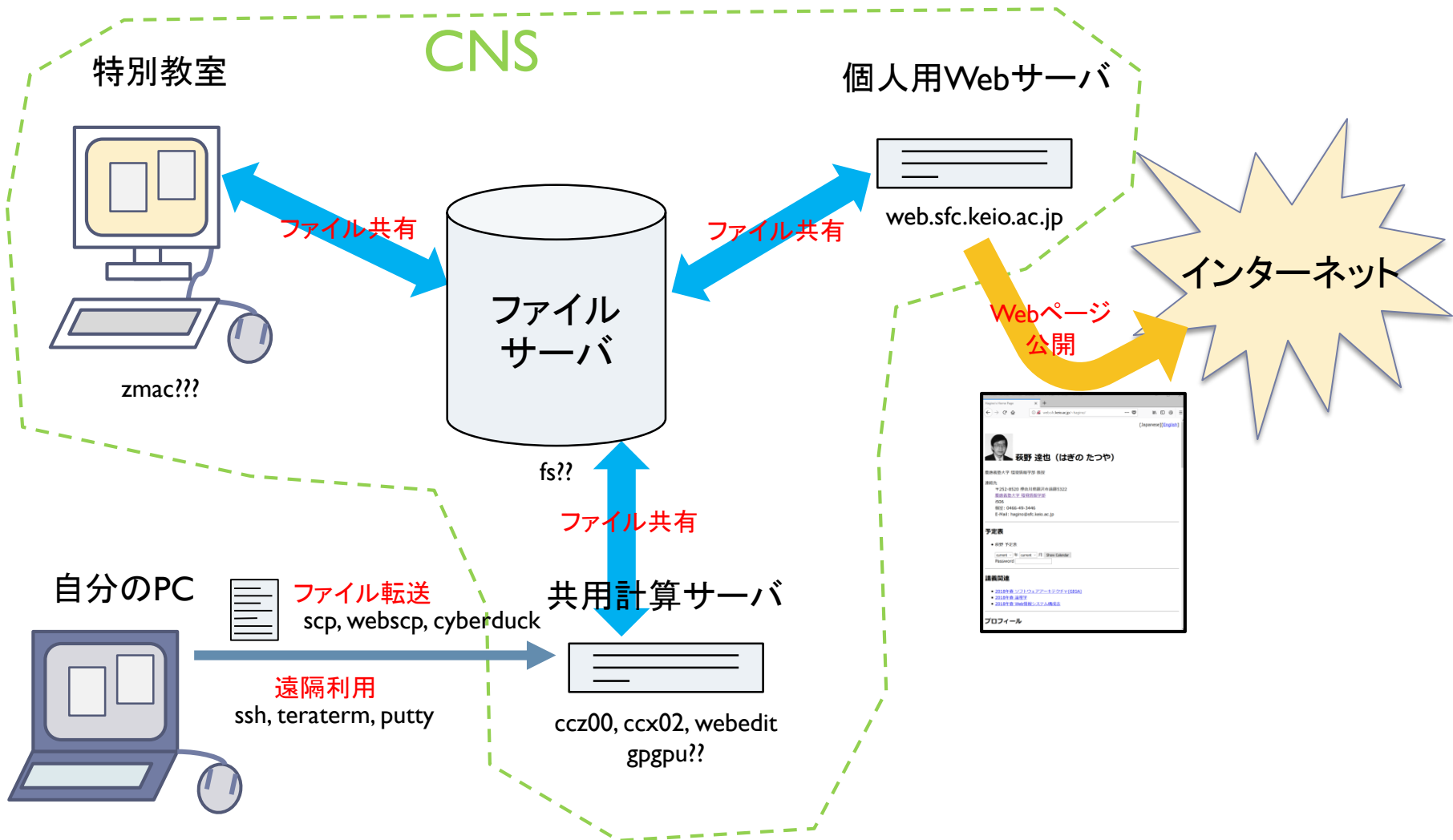
← このページの本文

← フッター
ヘルプナビなど

課題 2 : 複数のHTMLページ

- ▶ 自分の好きな海外の街を紹介するために、複数のページを作成し、メニューでこれらをリンクしなさい。
 - ▶ HTML5で記述すること。
 - ▶ HTML5の基本構造を利用して、ヘッダ、メニュー、本文、フッタに分けなさい。
 - ▶ HTMLだけを用い、CSSやJavascriptなどは用いないこと。
- ▶ 複数ページ
 - ▶ 街の概要を紹介するページ
 - ▶ 街の観光名所を紹介するページを2ページ以上
 - ▶ その他
- ▶ 提出
 - ▶ 作成したHTMLファイルをCNS上に置き、トップページのURLを提出してください。
 - ▶ 締め切り：今週土曜日

CNSのファイルサーバとWebページの関係



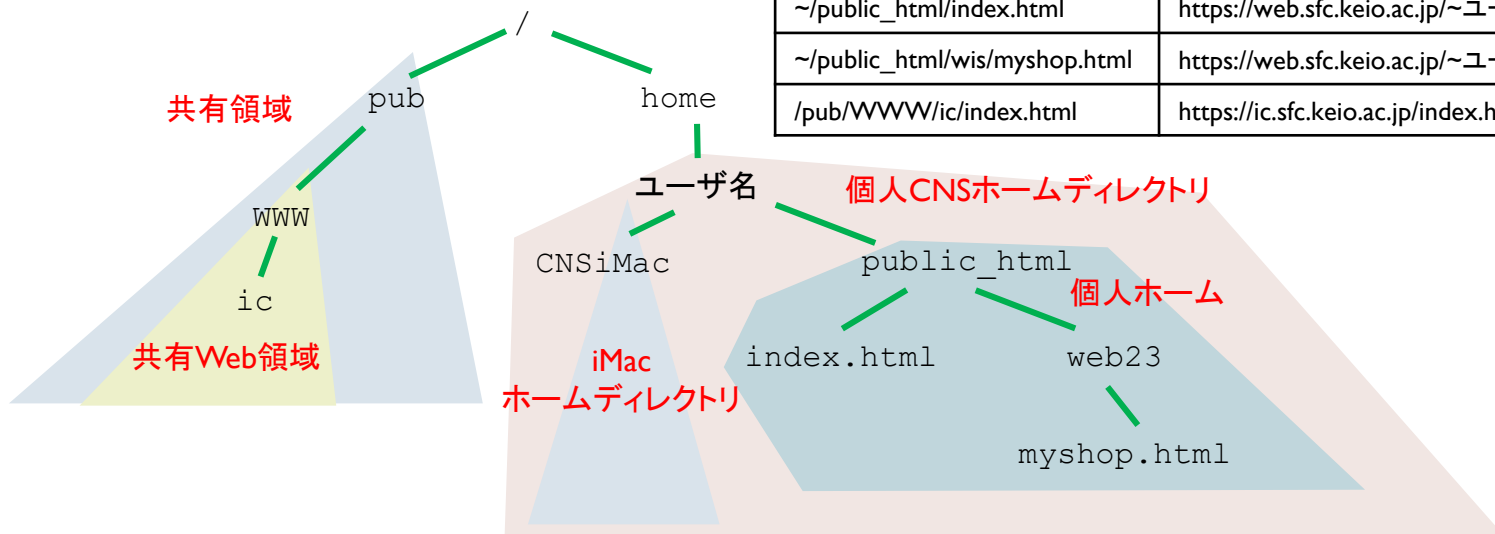
CNSファイルサーバ

- ▶ CNS内でファイルを共有している
 - ▶ CNSのどのPCでも同じ環境, 同じファイルを利用できる
 - ▶ 個人のホームディレクトリ
 - ▶ 利用者ごとに割り当てられる
 - ▶ 一部が個人Webページとして公開される
 - ▶ 共有ファイル領域
 - ▶ プロジェクトなどで共有して利用
 - ▶ プロジェクトなどのWebとして公開可能

~ は個人のホームディレクトリ
 ~ユーザ名
 /home/ユーザ名


ファイルサーバとURLの対応

パス名	Web URL
~/abc	非公開
~/public_html/index.html	https://web.sfc.keio.ac.jp/~ユーザ名/index.html
~/public_html/wis/myshop.html	https://web.sfc.keio.ac.jp/~ユーザ名/web23/myshop.html
/pub/WWW/ic/index.html	https://ic.sfc.keio.ac.jp/index.html



CNSへのWebページの置き方

▶ CNSの個人Webページ

- ▶ ホームディレクトリ下の public_html が公開される
- ▶ ~/public_html/ファイル名 
- ▶ https://web.sfc.keio.ac.jp/~ログイン名/ファイル名

▶ Web情報システム構成法のためにサブディレクトリを作ると便利

- ▶ ~/public_html/web23/town.html 
- ▶ https://web.sfc.keio.ac.jp/~ログイン名/web23/town.html

▶ public_html の設定

- ▶ webedit.sfc.keio.ac.jp に ssh などに入って編集
- ▶ 自分のPCで作成したファイルは scp などCNSに転送

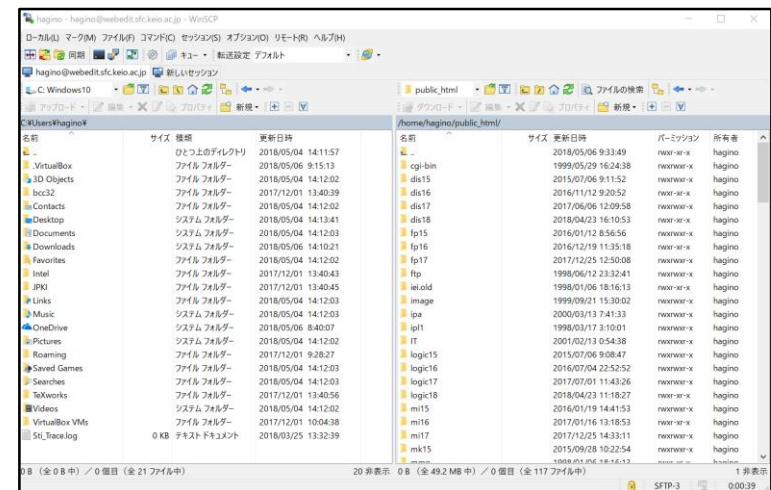
ファイル転送と遠隔利用

▶ ファイル転送

- ▶ 手元のPCのファイルをサーバにアップしたりダウンロードする
- ▶ scp
- ▶ WinSCP
- ▶ Cyberduck

▶ 遠隔利用

- ▶ サーバに入って、設定などを行う
- ▶ TeraTerm, PuTTY
- ▶ ssh



まとめ

- ▶ テキスト文書
 - ▶ テキスト形式
 - ▶ 文字コード

- ▶ ハイパーリンク
 - ▶ URL
 - ▶ ナビゲーション

- ▶ HTML5
 - ▶ 基本構造