

情報処理演習

WWWとホームページ

内容

- WWWとは
- ブラウザ
- 検索エンジン
- HTML
- 自分のホームページ作成

WWWの生い立ち

- CERN : Conseil Europeen pour la Recherche Nucleaire
- 加速器を使った研究成果としてのデータを世界中で共用する仕組みが必要
- 1990年頃に提案、実装 (NeXT)
- 1990年中頃から急速に普及

WWWの特徴

- 使い易いので、誰でも手軽に情報を入手できる
- サーバの立ち上げを簡単にできる
- テキストに加えて、図形、イメージ、動画、音などを含められる
- 誰でも見栄えのするページを作成できる
- 各種のブラウザを使用できる

キーワード

- URL : Uniform Resource Locator
 - 情報の在処
- ドメイン名 : コンピュータの名前
- HTML : Hyper Text Markup Language
 - Webページを書く言語
- プロトコル : 通信手順
 - HTTP : Hyper Text Transfer Protocol
 - HTTPS : Hypertext Transfer Protocol Security

ブラウザ

- Safari : Mac OSに標準で搭載
- Internet Explorer (IE) : Microsoft の Windows に標準で搭載
- Firefox : 急速に普及してきたブラウザ、Linuxなどでは標準
- Netscape Navigator : IE に対抗するブラウザで、自分で他所から入手してインストールする必要
- OPERA : LinuxやMac OS 用のものがあり、愛好者は多い

Safariの基本機能

- 戻る
- 進む
- 更新
- 中止
- ブックマークバー
- ブックマーク
- 検索

Safariの設定について

- 起動時のページ
- 一時ファイル(キャッシュ)の削除
- Cookiesの削除
- 履歴の削除
- パスワードの記録

知っていると便利な機能

- 文字列のコピーとペースト
- 文字列の検索
- ソースコードの表示
- 文字の大きさの変更

保存

- ページの保存
 - Webページはいつまでもあるとは限らない
 - 後から見たい場合には保存すべし
- イメージの保存
 - イメージだけ保存することができる
 - 著作権には注意すべし

インターネット上の情報の特徴

- 国境がない。(国内、国外を問わず情報を発信、入手できる)
- 情報の入手に時間がかからない。
- 発信も受信もコストが安い。
- オンラインで検索できる。したがって、再利用も容易である。
- 最新のデータを自分で保持する必要がない(各種統計データなど)。ただしものによっては、その情報がいつなくなるか分からない。
- 情報は必ずしも正しいとは限らない。

検索エンジン

- 検索の仕方
 - 電話帳型 = 検索のキーワードが明確な場合
 - カテゴリ型 = 曖昧な検索をしたい場合
- 検索エンジンの種類
 - ディレクトリ型
 - 人間が分類
 - Yahoo
 - ロボット型
 - 自動的に情報収集
 - Google

検索エンジン使用上の注意

- キーワードを空白で区切って並べる
- 除きたい場合にはマイナスをつける
- 特定のフレーズを検索したい時は“”で挟む

引用の注意

- 引用する部分を明確に区別
- 出所を明示
- 著作権に十分に注意
- 情報がいつまであるか不明なので配慮
- 課題でそのままペーストするなどということはないように（読んだらすぐに分かる！）

Webページの構造

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head> <title> ... </title> </head>
  <body> ... </body>
</html>
```

metaタグの中に文字
コードを記述する

Webページで使用する文字コードを明示する場合

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

- Htmlタグ
- Headタグ
- Bodyタグ
- 見出しのタグ

箇条書き(1)

行の先頭に黒丸などをつける場合

基本形 : ` `

``

` 1番目の項目`

` 2番目の項目`

``

` 1番目の項目`

` 2番目の項目`

``

` 3番目の項目`

``

- 1番目の項目
- 2番目の項目
 - 1番目の項目
 - 2番目の項目
- 3番目の項目

箇条書き(2)

行の先頭に数字などをつける場合

```
<ol>
```

```
<li> 赤いリンゴ</li>
```

```
<li> 青いリンゴ</li>
```

```
<ol>
```

```
<li> 青い海</li>
```

```
<li> 白い空</li>
```

```
</ol>
```

```
<li> 黄色いバナナ</li>
```

```
</ol>
```

1. 赤いリンゴ

2. 青いリンゴ

1. 青い海

2. 白い空

3. 黄色いバナナ

箇条書き(3)

<dl>

<dt> 1番目の項目</dt>

<dd> 1番目の説明</dd>

<dt> 2番目の項目</dt>

<dd> 2番目の説明</dd>

<dt> 3番目の項目</dt>

<dd> 3番目の説明</dd>

</dl>

1番目の項目

1番目の説明

2番目の項目

2番目の説明

3番目の項目

3番目の説明

テーブル

```
<table border=1 width=60%>
```

```
<tr><th>1番目の見出し</th><td align=left>1番目の内容</td></tr>
```

```
<tr><th>2番目の見出し</th><td align=center>2番目の内容</td></tr>
```

```
<tr><th>3番目の見出し</th><td align=right>3番目の内容</td></tr>
```

```
</table>
```

1番目の見出し	1番目の内容
2番目の見出し	2番目の内容
3番目の見出し	3番目の内容

文字飾り

ソース	表示
<code><i>Italic</i></code>	<i>Italic</i>
<code>Bold</code>	Bold
<code><tt>Typewriter</tt></code>	Typewriter
<code><S>Strike String</S></code>	Strike String
<code>青色</code>	赤色
<code>X<sup>2</sup></code>	X ²
<code>H<sub>2</sub>O</code>	H ₂ O
<code><marquee>SCROLL</marquee></code>	SCROLL

文字の大きさ

ソース	表示
 大きさ= 1 	大きさ= 1
 大きさ= 2 	大きさ= 2
 大きさ= 3 	大きさ= 3
 大きさ= 4 	大きさ= 4
 大きさ= 5 	大きさ= 5
 大きさ= 6 	大きさ= 6
 大きさ= 7 	大きさ= 7

見出し

ソース	表示
<pre><h1> 大きさ 1 の見出し </h1>
 <h2> 大きさ 2 の見出し </h2>
 <h3> 大きさ 3 の見出し</h3>
 <h4> 大きさ 4 の見出し</h4>
 <h5> 大きさ 5 の見出し</h5>
 <h6> 大きさ 6 の見出し</h6>
</pre>	<p>大きさ 1 の見出し</p> <p>大きさ 2 の見出し</p> <p>大きさ 3 の見出し</p> <p>大きさ 4 の見出し</p> <p>大きさ 5 の見出し</p> <p>大きさ 6 の見出し</p>

改行と段落

ソース	表示
<pre><p align=left> テキストを左詰めにする。 </p> <p align=center> テキストを中央揃えにする。 </p> <p align=right> テキストを右詰めにする。 </p></pre>	<p>テキストを左詰めにする。</p> <p>テキストを中央揃えにする。</p> <p>テキストを右詰めにする。</p>

もろもろ

- 線を引く
 - `<hr>`
 - `<hr size=10pt>`
 - `<hr width=50% align=left>`
- バックグラウンドの色
 - `<body bgcolor="#FFFFFF">`
 - `<body bgcolor="red">`

リンク

基本形 : ` 他のファイルへのリンク `

`HTML用の別の窓<p>`

` 他のコンピュータへのリンク `

` 別窓を開いて表示`

` イメージ(写真)の張り付け`

``

そのまま表示

<pre>の有無	表示
なし	<pre>public class app { public static void main(String[] args) { System.out.println("Hello from Java!"); } }</pre>
あり	<pre>public class app { public static void main(String[] args) { System.out.println("Hello from Java!"); } }</pre>

自分のページの作成

- Finderでpublic_htmlを確認
- Index.htmlを作成
- <http://www.s.kyushu-u.ac.jp/~1AB09123X/>
- 中身は簡単で良い

自分のページの見本

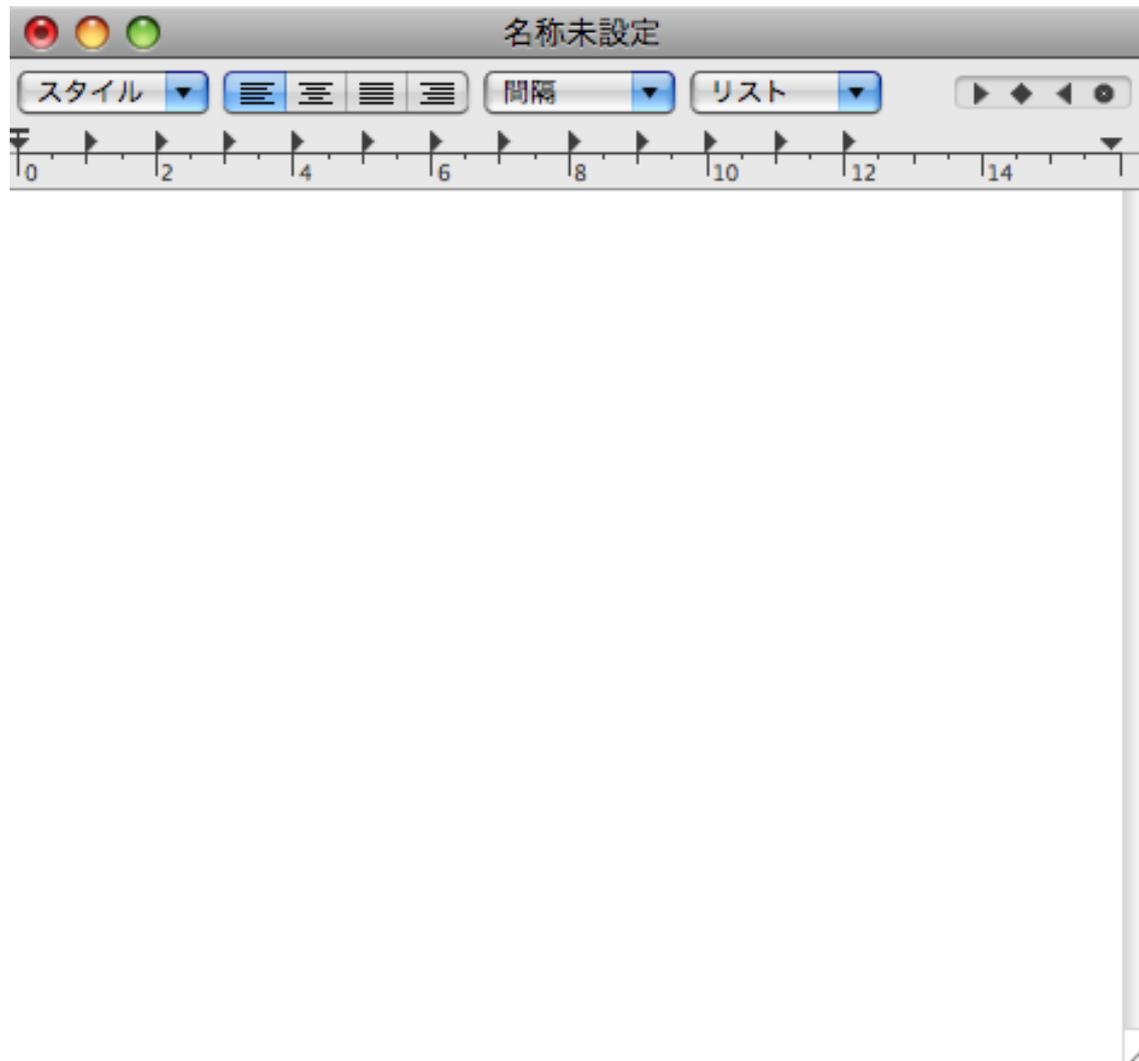
```
<html>
<head>
<meta http-equiv="content-type" content="text/html;charset=UTF-8">
<title>HTMLの見本</title>
</head>
<BODY BGCOLOR="#ffffc0">
<font size=4>
<h2>藤村直美</h2>
私の趣味は久住登山です。
<p>
</body>
</html>
```

テキストエディタを使用する方法

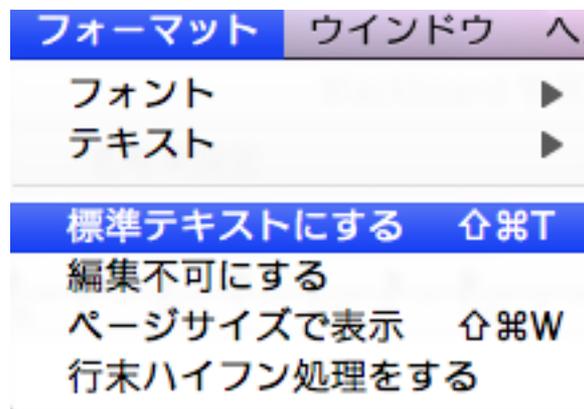
自分のHPの作成手順

- Dockからテキストエディタを起動
- 中身を作成する
- Fileから別名で保存を選択
- Index.htmlという名前と、public_htmlという場所を選択する
- 「保存」で保存する
- 次からは「ファイル」→「開く」で読み出すか、index.htmlをダブルクリックする

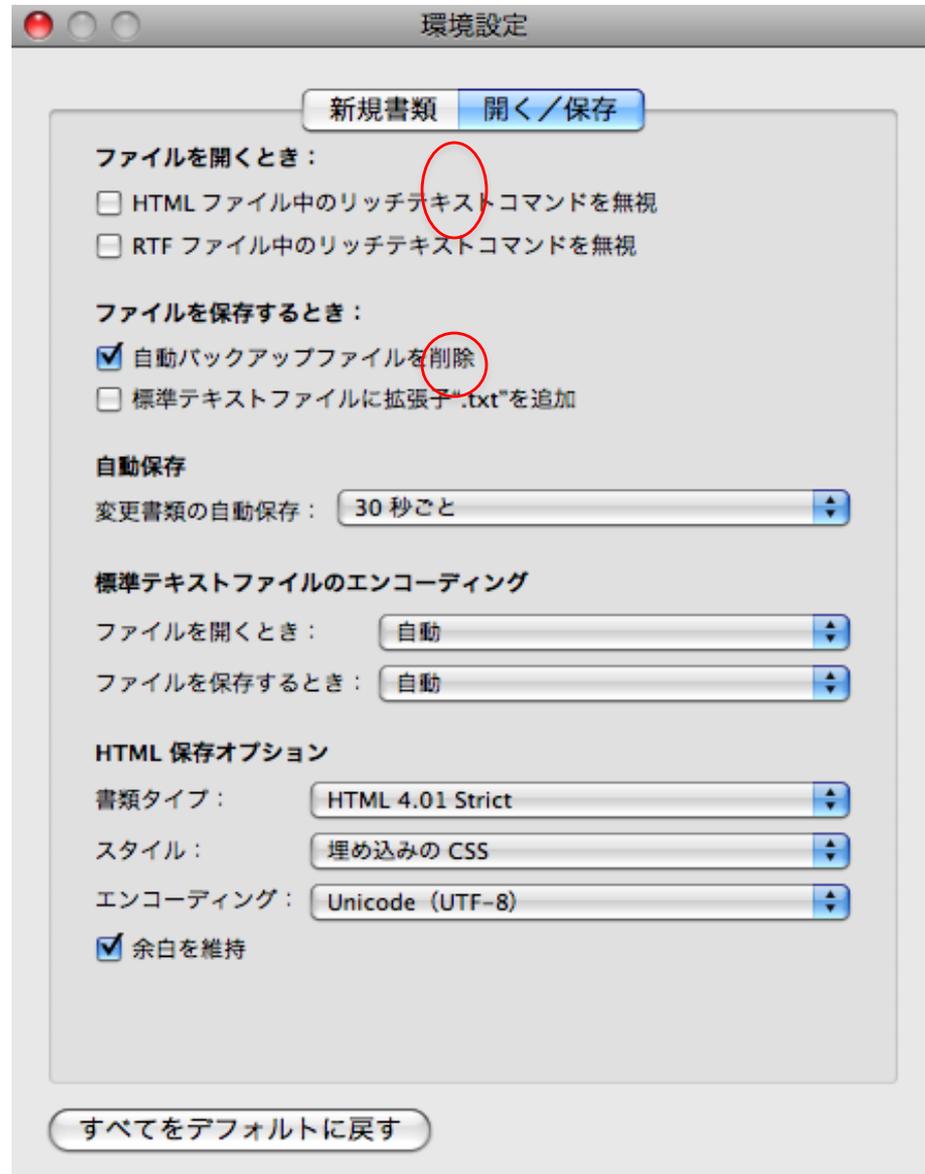
テキストエディタを起動



リッチテキストから標準テキストへ



HTMLのタグを無視する設定



中身を入力

```
名称未設定
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>課題のみ本</title>
</head>

<body bgcolor="pink">

<h2>藤村のホームページ</h2>

<table border=1>
<tr><th>項目</th><td>内容</td></tr>
<tr><th>趣味</th><td>久住登山</td></tr>
<tr><th>好きなこと</th><td>仕事</td></tr>
</table>

<ol>
<li> 赤いリンゴ
<li> 青いみかん
</ol>

<a href="http://www.google.co.jp">Googleのページ</a>

</body>
</html>
```

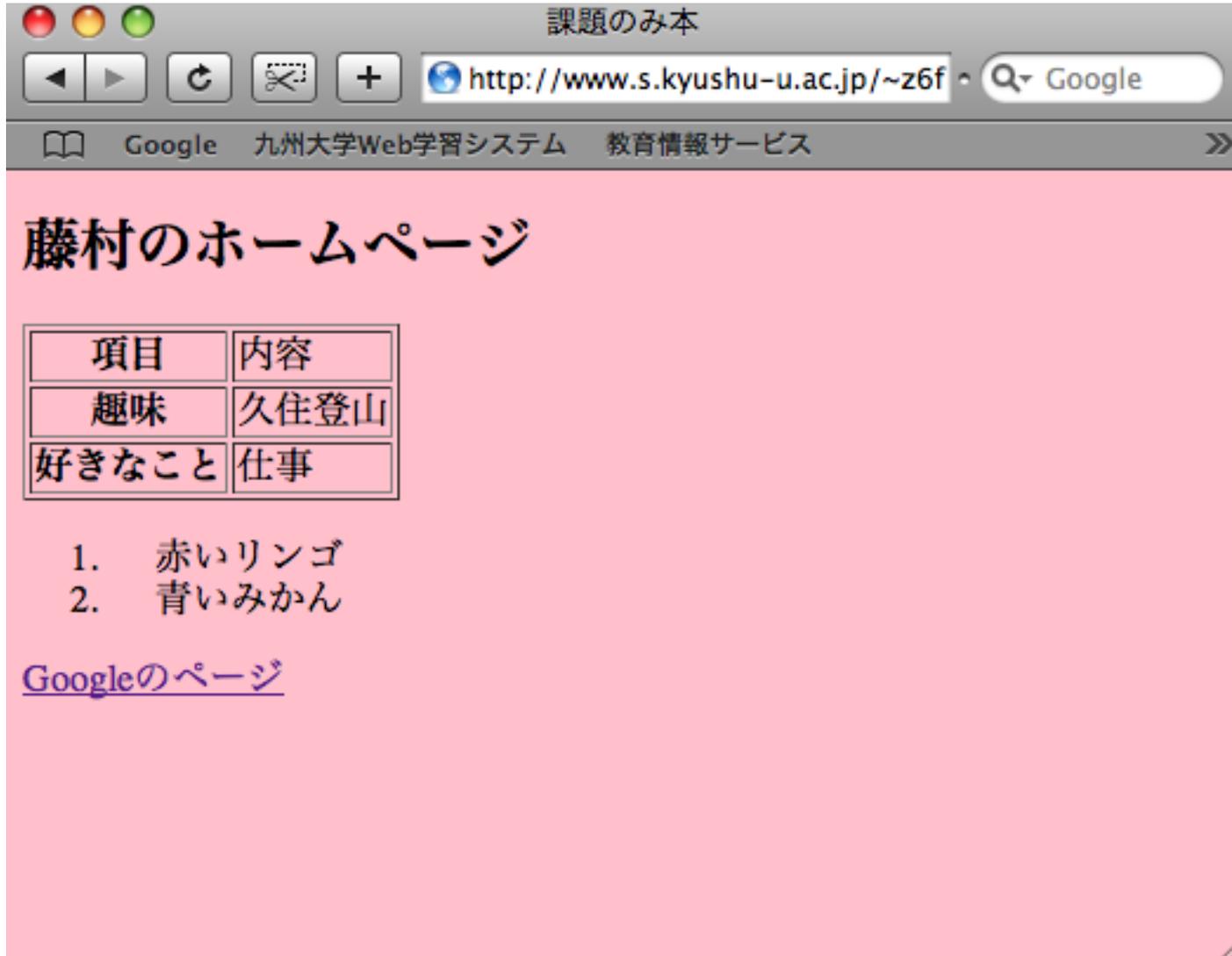
保存

ファイル	編集	フォーマット
新規		⌘N
開く...		⌘O
最近使った項目を開く		▶
閉じる		⌘W
保存		⌘S
別名で保存...		⇧⌘S
すべてを保存		
最後に保存した状態に戻す		
プロパティを表示		⇧⌘P
ページ設定...		⇧⌘P
プリント...		⌘P

保存場所を指定



ブラウザで表示



The screenshot shows a web browser window with the following elements:

- Window title: 課題のみ本
- Address bar: <http://www.s.kyushu-u.ac.jp/~z6f> with a Google search button.
- Navigation bar: Google, 九州大学Web学習システム, 教育情報サービス.
- Page title: 藤村のホームページ
- Table:

項目	内容
趣味	久住登山
好きなこと	仕事

- 1. 赤いリンゴ
- 2. 青いみかん

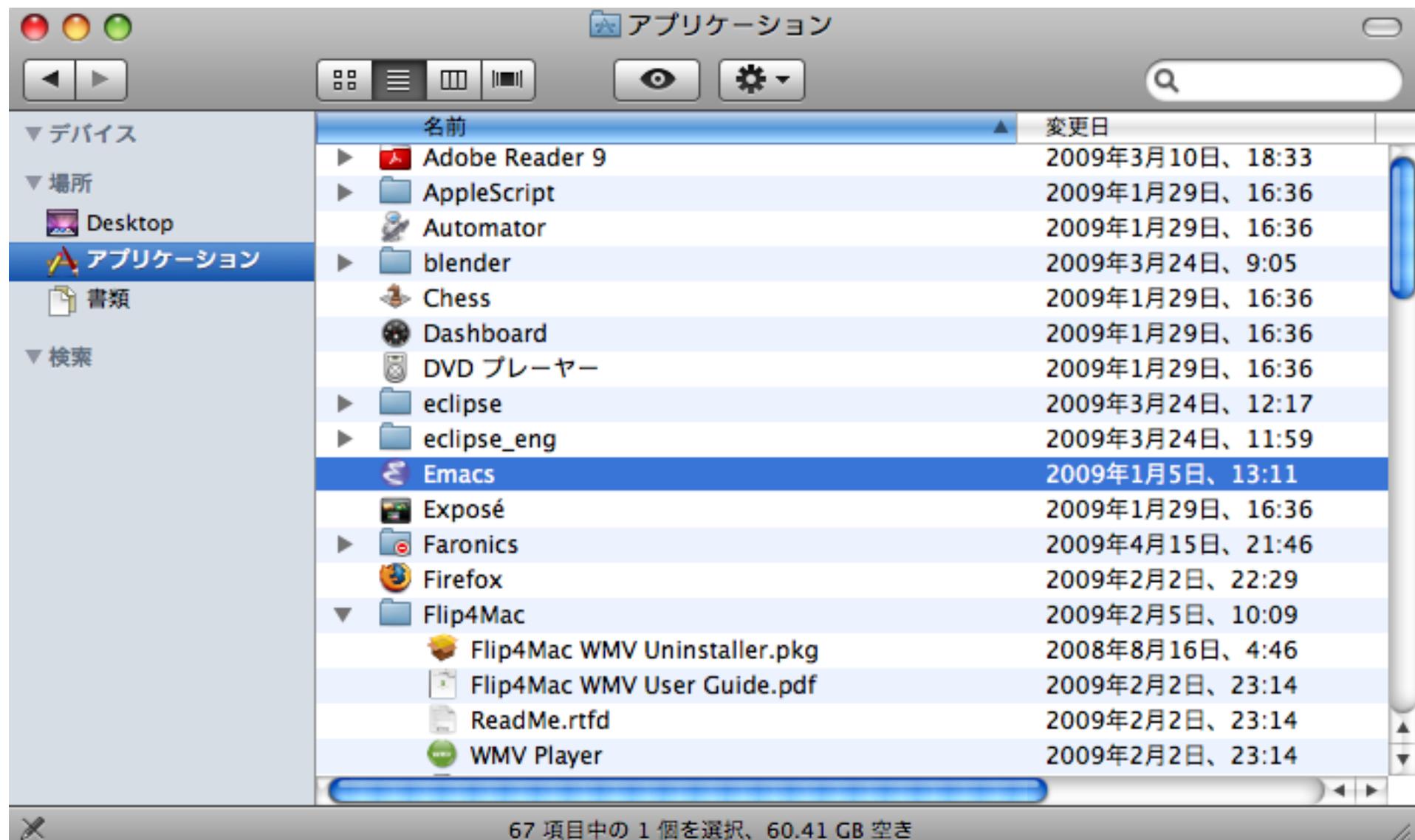
[Googleのページ](#)

emacsを使用する方法

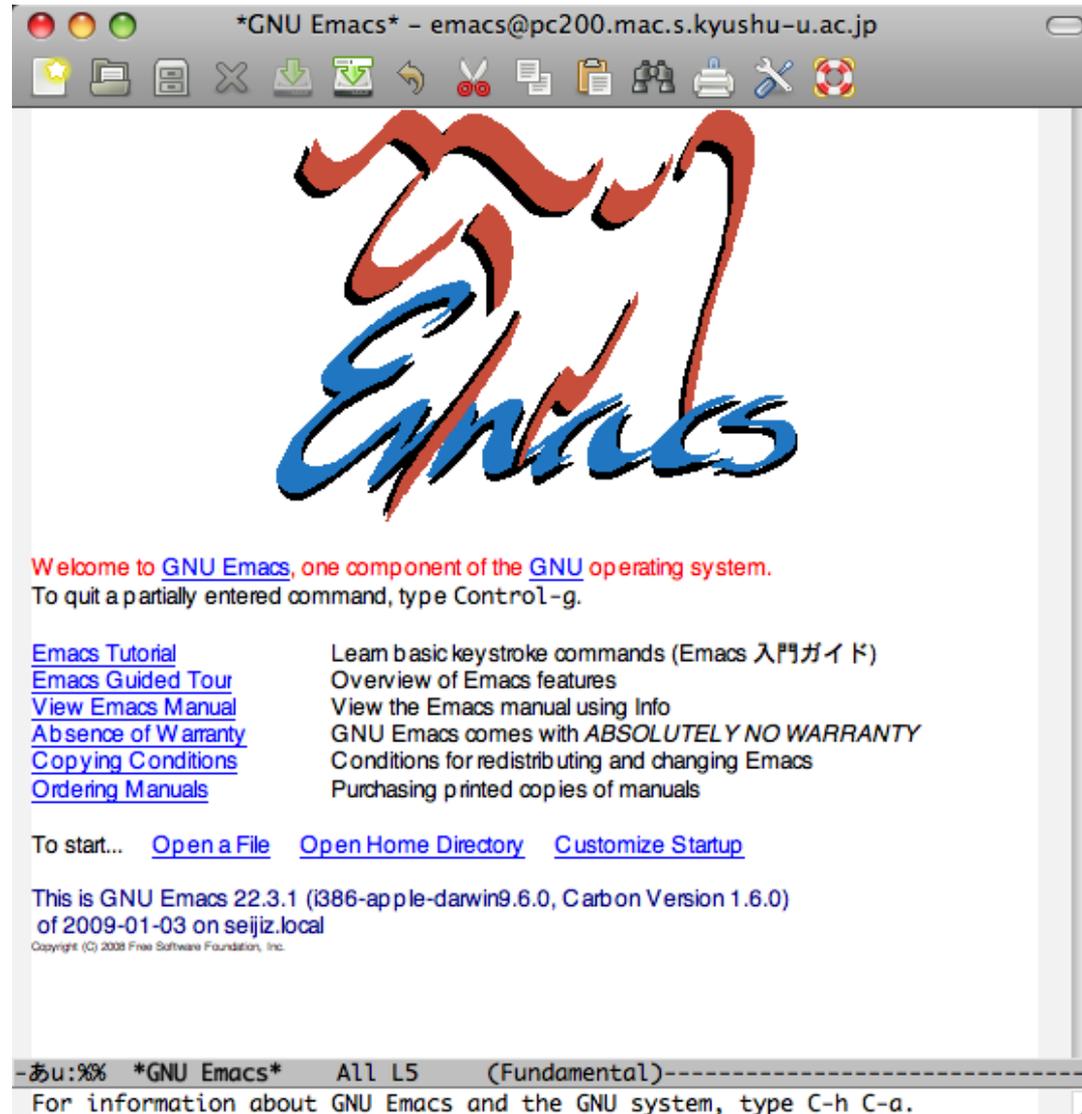
自分のHPの作成手順

- Finderのアプリケーションからemacsを起動
 - EmacsをDockに入れておくとこれから楽
- FileからVisit New File
- Index.htmlという名前と、public_htmlという場所を選択する
- 中身を作成する
- Save で保存する
- 次からはFile → Open Fileで読み出す

Emacsの起動



Emacsの起動画面



File Edit Options Buffers

Visit New File... (C-x C-f)

Open File...

Open Directory... (C-x d)

Insert File... (C-x i)

Close

Save (C-x C-s)

Save As... (C-x C-w)

Revert Buffer

Recover Crashed Session

Print Buffer

Print Region

Postscript Print Buffer

Postscript Print Region

Postscript Print Buffer (B+W)

Postscript Print Region (B+W)

Split Window (C-x 2)

Remove Splits (C-x 1)

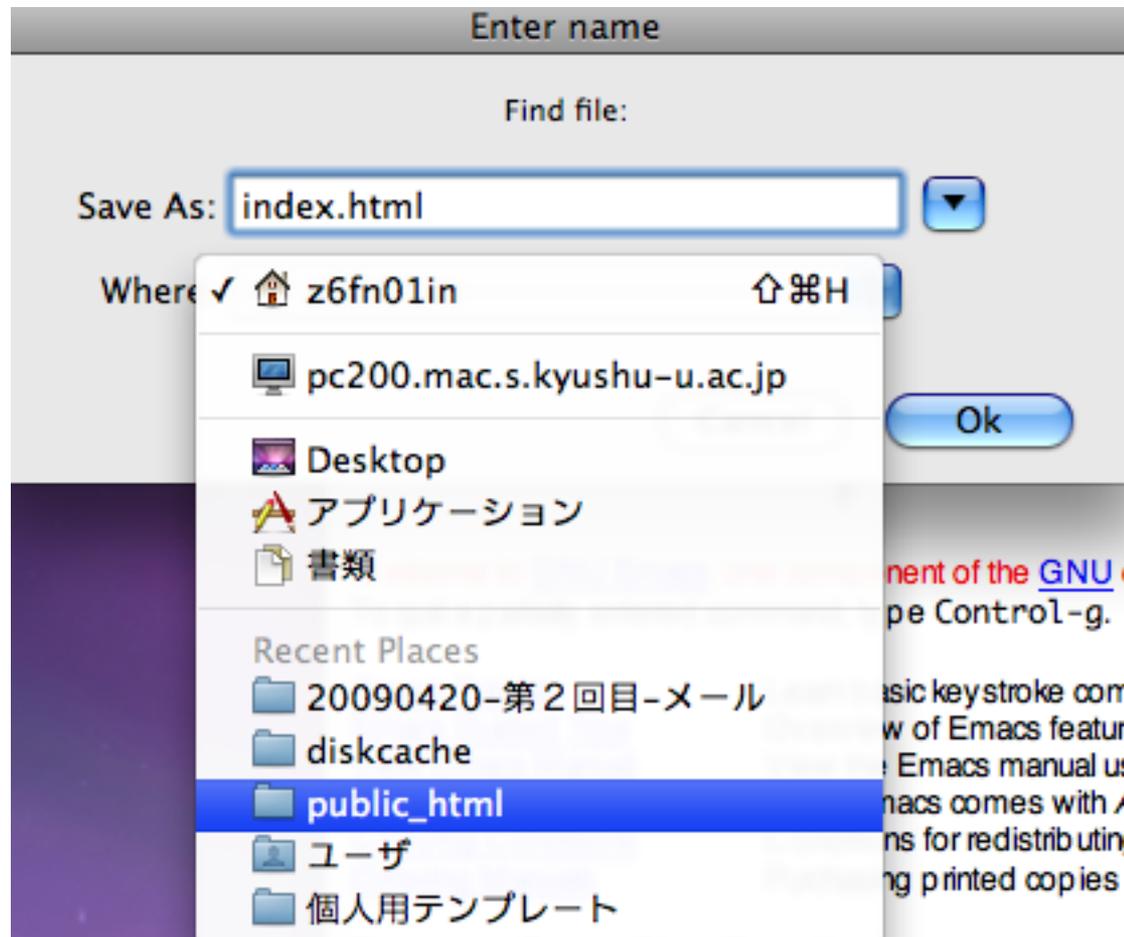
New Frame (C-x 5 2)

New Frame on Display...

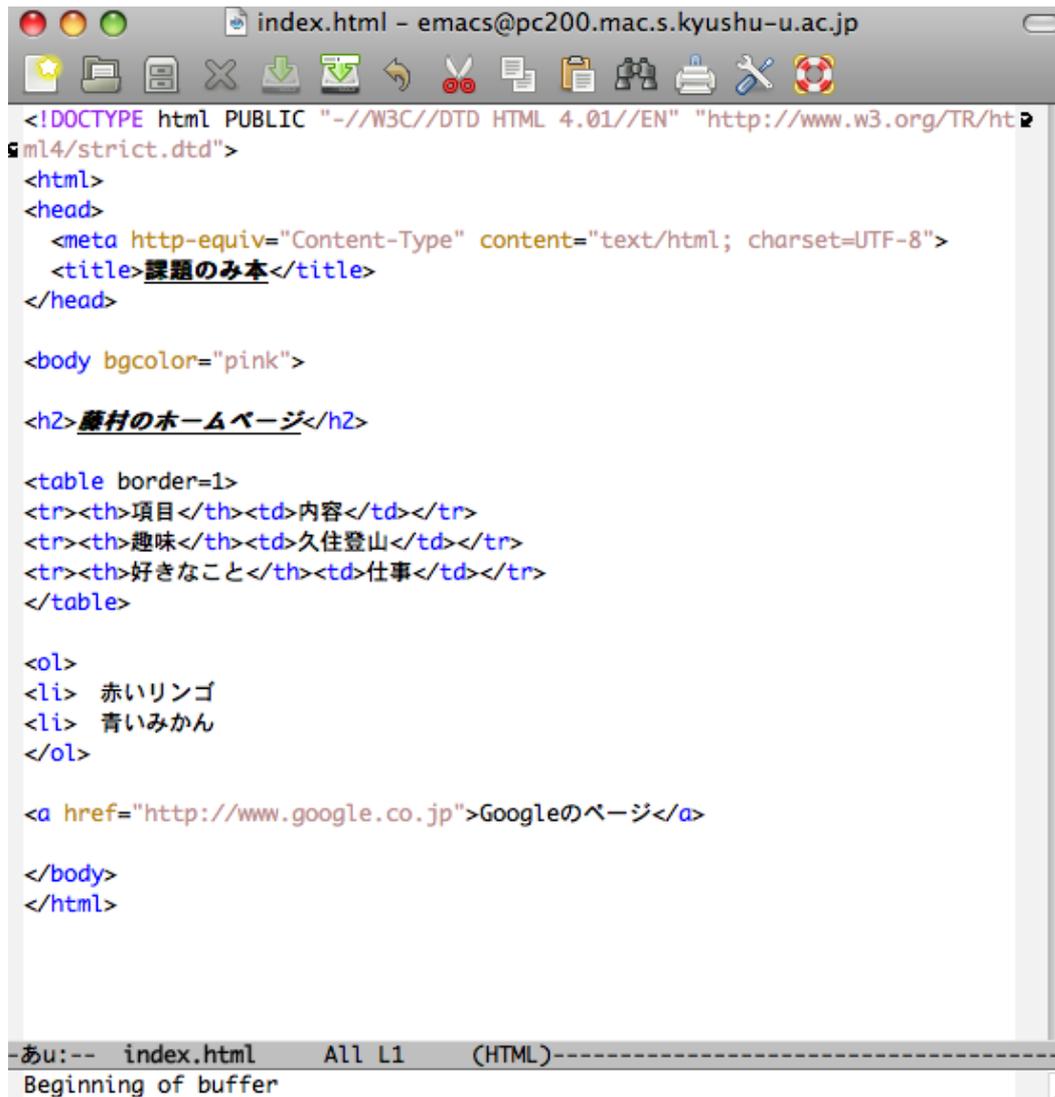
Delete Frame (C-x 5 0)

Exit Emacs (C-x C-c)

ファイルと場所の指定



中身を入力

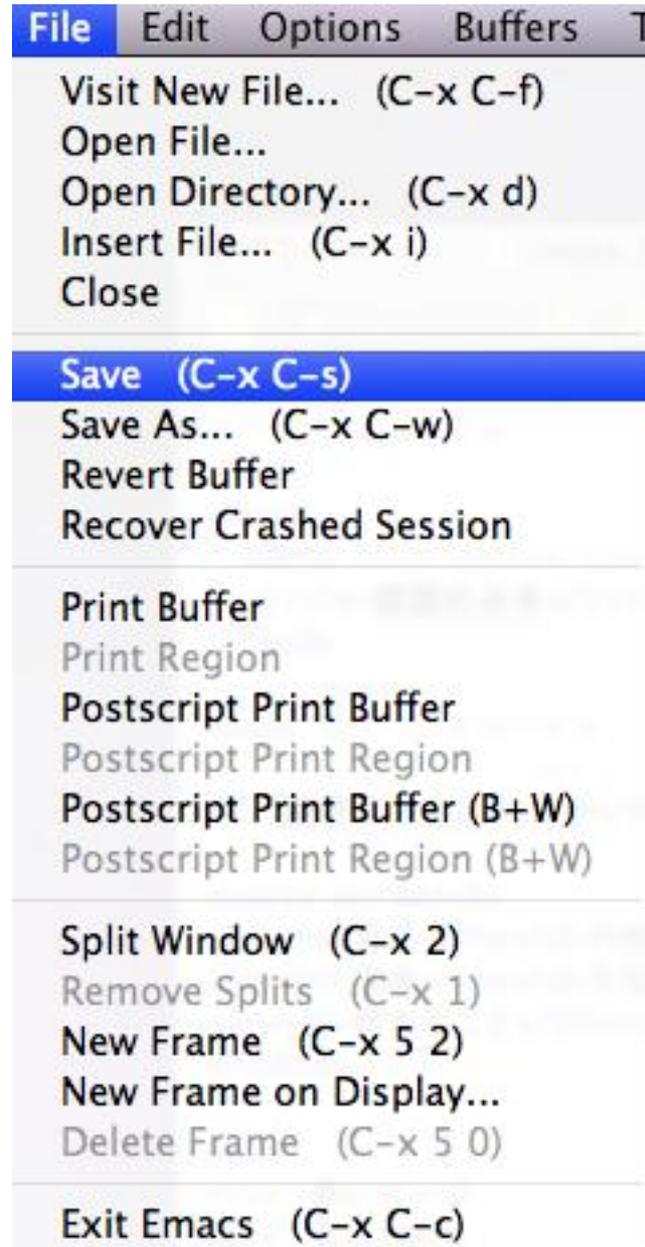


The image shows a screenshot of an Emacs editor window titled "index.html - emacs@pc200.mac.s.kyushu-u.ac.jp". The window displays HTML code for a web page. The code includes a DOCTYPE declaration, a head section with a meta tag for content type and a title "課題のみ本", and a body section with a pink background color. The body contains an h2 heading "藤村のホームページ", a table with three rows (Item, Content, Interest, Hobby, Favorite things, Job), an ordered list with two items (Red apple, Green mandarin), and a link to Google's page.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>課題のみ本</title>
</head>
<body bgcolor="pink">
  <h2>藤村のホームページ</h2>
  <table border=1>
    <tr><th>項目</th><td>内容</td></tr>
    <tr><th>趣味</th><td>久住登山</td></tr>
    <tr><th>好きなこと</th><td>仕事</td></tr>
  </table>
  <ol>
    <li> 赤いリンゴ
    <li> 青いみかん
  </ol>
  <a href="http://www.google.co.jp">Googleのページ</a>
</body>
</html>
```

あう:-- index.html All L1 (HTML)-----
Beginning of buffer

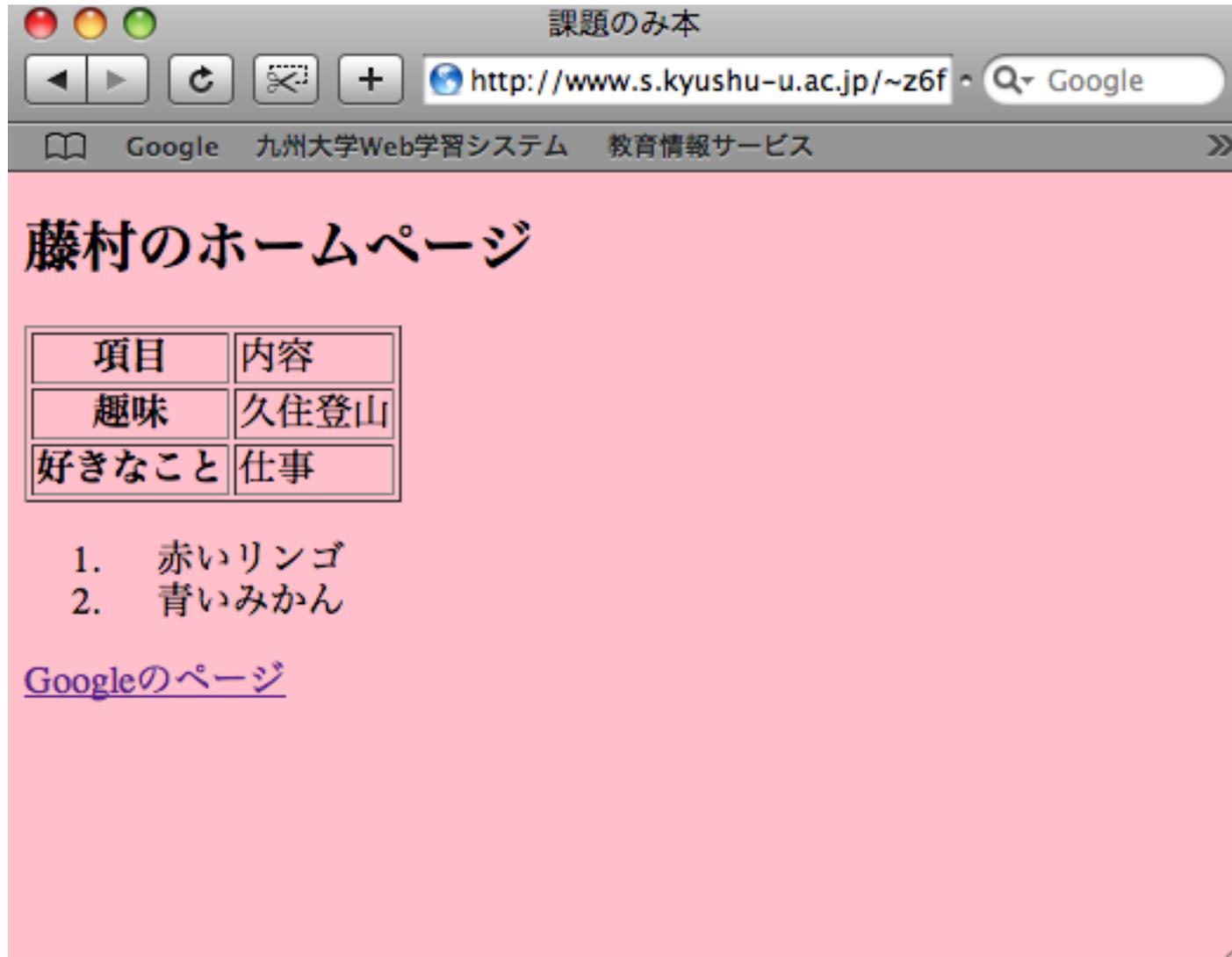
保存



Safariで表示するURL

<http://www.s.kyushu-u.ac.jp/~1AB09123X/>

表示例



The screenshot shows a web browser window with a title bar that says "課題のみ本". The address bar contains the URL "http://www.s.kyushu-u.ac.jp/~z6f" and a search button labeled "Google". The browser's bookmark bar shows "Google", "九州大学Web学習システム", and "教育情報サービス". The main content area has a pink background and features the title "藤村のホームページ". Below the title is a table with two columns: "項目" (Item) and "内容" (Content). The table contains three rows: "趣味" (Hobby) with "久住登山" (Kuzumi登山), and "好きなこと" (Favorite things) with "仕事" (Work). Below the table is a numbered list: "1. 赤いリンゴ" (1. Red apple) and "2. 青いみかん" (2. Green mandarin). At the bottom, there is a link labeled "Googleのページ" (Google's page).

課題のみ本

http://www.s.kyushu-u.ac.jp/~z6f Google

Google 九州大学Web学習システム 教育情報サービス

藤村のホームページ

項目	内容
趣味	久住登山
好きなこと	仕事

1. 赤いリンゴ
2. 青いみかん

[Googleのページ](#)

6. WWW とホームページ作成

6.1. WWW とは

WWW (World Wide Web) が最初に実現されたのはスイスのジュネーブ郊外に本拠地がある CERN (Organisation europeenne pour la recherche nucleaire (仏語表記)、European Organization for Nuclear Research (英語表記)) です。この組織を設立するための準備会の正式名称がフランス語で Conseil Europeen pour la Recherche Nucleaire だったので CERN と呼ばれているそうです。CERN は世界最大規模のヨーロッパ合同原子核研究機構の別名で、1 周が 27km の加速器を使って素粒子物理学および原子核物理学の研究を行っている組織です。

WWW ができたきっかけは、核素粒子の実験結果を全世界の研究者で共有する際に、当時、利用可能だった電子メールやファイル転送 (ftp) では使い勝手が悪く、何らかの改善が必要だったためとされています。歴史的な流れはおよそ次のようでした。

- 1989 年 3 月にプロポーザル
- 1990 年 10 月にプロトタイプ
- 1991 年 3 月に最初のブラウザ、最初は NeXT というワークステーションで開発
- 1991 年 10 月に専用のメーリングリスト
- 1993 年 1 月に Mosaic などのブラウザが出現
- 1992 年 9 月 30 日に日本 で最初のホームページ (<http://www.tsukuba.org/www/>)
- 1993 年秋頃から日本国内で WWW が普及し始め
- 1994 年 5 月に九州芸術工科大学で WWW サーバを設置、早い時期から動くアニメーションを置いたりしたので、世界的に有名な時期があった！

WWW には次のような特徴があります。

- 使いやすいので、利用者が手軽に情報を入手することができる。
- サーバの立ち上げが簡単にできる。
- テキストに加えて、図形、イメージ、動画、音などを含めることができる。
- 誰でも見栄えのするページを作ることができる。
- 昔は CERN と NCSA のブラウザが双頭だったが、今や Microsoft、Netscape、Firefox、Mozilla など、各種のブラウザを使用できる。

WWW ではいくつかのキーワードが出てきます。

URL : Uniform Resource Locator の略で情報資源が何処にあるかを表示するための記述方法です。通常は

プロトコル名://ドメイン名/場所/ファイル名

といった書き方になります。

プロトコル：コンピュータ同士が通信する時の手順を規定するものです。

ドメイン名：みなさんが日頃使っている特定のコンピュータを表す名前です。WWW で始まることが多いのですが、これは WWW で始まらないといけない訳ではなく、WWW で始まっているのは多くの場合に、これが WWW サーバであることを明示したいからです。ファイル名を省略することができ、その場合にはサーバの設定にもよりますが、index.html や index.php などが指定されていると推定されて、処理されます。省略するとエラーになる設定もあります。見本をいくつか、例として次に示しておきます。

`http://www.design.kyushu-u.ac.jp/`

`ftp://ftp.kyushu-u.ac.jp/pub/GNU/prep/emacs-21.3.tar.gz`

HTML：Hyper Text Markup Language の略です。ハイパーテキストを記述する言語の詳細はホームページを作成するところで勉強します。基本的には文字列をどのように修飾するかを示すタグと呼ばれる修飾子を付け加える形式で記述します。

HTTP：Hyper Text Transfer Protocol の略で、WWW のサーバとクライアント間のプロトコルです。ブラウザから WWW サーバにリクエスト(皆さんがクリックすると)を送ると、受け取った情報に応じて WWW サーバが処理して、結果を戻します。ブラウザはサーバから戻ってきた HTML の内容を解釈して、画面に表示します。情報を請求して、それに対して反応している実際の例を図 6-1 に示すので、参考にしてください(青地の文字がユーザが端末から入力する部分)。ただ telnet コマンドやポートのことなど、色々なバックグラウンドを理解していないと意味が通じないかもしれないので、今後の勉強に期待したいと思います。みなさんがクリックするとこのような情報が裏で飛び交うということです。

HTTPS：Hypertext Transfer Protocol Security の略で、WWW サーバとクライアントの間の通信に SSL による暗号化機能を付加したプロトコルです。これを使用すると、氏名や住所、電話番号など、あるいはクレジットカードの番号や有効期限などの重要な情報を安全に通信することができます。

```

[fujimura@SAMPLE]$ telnet sample.design.kyushu-u.ac.jp http

Trying 133.5.123.45...
Connected to sample.design.kyushu-u.ac.jp (133.5.123.104).
Escape character is '^]'.
GET /class/index.html HTTP/1.1
host: sample.design.kyushu-u.ac.jp

HTTP/1.1 200 OK
Date: Sun, 10 May 2009 09:00:23 GMT
Server: Apache/2.2.3 (CentOS)
Last-Modified: Thu, 16 Apr 2009 08:55:52 GMT
ETag: "1b181cb-3e6-3b2fd600"
Accept-Ranges: bytes
Content-Length: 998
Connection: close
Content-Type: text/html

<html>
<head>
<meta http-equiv="CONTENT-TYPE" CONTENT="text/html;CHARSET=EUC-JP">
<title>List of Lecture materials</title>
</head>

<body>
Updated on 2009/04/16<br>
Copyright : fujimura@design.kyushu-u.ac.jp<br>
<hr>

<H2>授業用教材 他</H2>
<HR>
<P>

途中を省略します。。。。。

<hr>

</font>

</body>

Connection closed by foreign host.
[fujimura@SAMPLE]$ exit

```

図 6-1 HTTP の通信例

6.2. ブラウザについて

WWW とは World Wide Web の略です。俗に「インターネットする」という言い方をする人がおり、その本当の意味は WWW の仕組みを利用して、ブラウザで色々な情報を検索・表示することのようです。

世界で最初に開発されたグラフィカルな Web ブラウザは NCSA によって開発された Mosaic です。最近では、Microsoft 社の Internet Explorer、Mosaic 開発チームが興した Netscape Communications 社の Netscape Communicator (Navigator)、Netscape 社が立ち上げたオープンソースプロジェクト「Mozilla.org」による Mozilla、その後継の Firefox、Opera Software 社の Opera などが広く使われています。また、テキストしか表示できない (しない) 「テキストベース Web ブラウザ」として、w3m や lynx が有名です。Macintosh では Safari が標準で入っています。最近、世間で広く使われている WWW ブラウザとしては次のようなものがあります。

- Internet Explorer (IE) : Microsoft の Windows に標準で含まれているので広く使われています。
- Firefox : 急速に普及してきたブラウザです。Linux などでは標準で入っています。
- Netscape Navigator : IE に対抗するブラウザですが、Windows には標準で含まれていないので、他所から入手して使えるようにしないとイケません。
- OPERA : これも標準で使えるようにはなっていませんが、Linux や Mac OS 用のものがあり、愛好者は多いようです。

もう少し詳しい歴史については Google で検索するなどして、調査すると面白いです。何故、当初 Mosaic が普及したのか、何故開発者のマーク・アンドリーセンはイリノイ大学から新しい会社に移ったのかなど、自分で調べてみましょう。

6.3. ブラウザの基本機能

ここでは Safari の機能を中心に説明をします。

① ボタンの説明

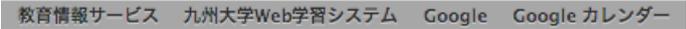
-  (戻る) : 一つ前の画面に戻ります。
-  (進む) : 前の画面に戻った後で、次の画面に進みます。
-  (更新) : 普通に「戻る」ボタンを押して、以前に表示した画面に戻ると、一時ファイル(キャッシュ)として保存されている内容を表示します。これでは元のページのデータが更新されていた場合に内容の変更が反映されません。そこで、変更があったかもしれない、変更されているはずだ、最新の情報を表示したいといった場合に、このボタンをクリックすると、表示しているページを取ってきた場所から最新のページの内容を改めて転送してきます。なお、一定時間毎に更新を自動的に行う仕組みもあり、例えば道路交通情報センターの福岡年高速道路の状況 (<http://www.jartic.or.jp/>) などはその例です。このボタンをクリックしても内容が更新されていなくて、それでも間違いなく内容は更

新されているはずだと思う時は、図 6-2 に示すように「Safari」→「キャッシュをクリア」を実行すると、すべてのキャッシュが消去されるので、間違いなく内容が最新のものになります。

もともとこのようにキャッシュとして前の画面の情報を記録する仕組みは、インターネットの回線が細かった時代に、常に画面の情報を転送すると時間がかかること、相手のサーバに負担をかけることなどに配慮して、組み込まれたと思いますが、理由と仕組みをきちんと理解しておかないと正しい使い方ができなくなります。



図 6-2 キャッシュのクリア

-  ページを表示しようとして、待っていてもなかなか表示されない場合に表示を中止する時にクリックします。ページが表示されるのに時間がかかる理由は、相手のサーバが混んでいる、途中の回線が混んでいる場合などがあります。ドメイン名を間違えて指定した場合には大抵はすぐに名前が違うとか、そのようなドメインはないというエラーメッセージが出ます。
-  (ブックマークバー)：ここに登録されているドメイン名をクリックすると、すぐにそのブックマークに移動して、内容を表示します。ここで示しているのは著者のブラウザの例です。
-  これをクリックするとお気に入りの一覧を画面の左側に分割して表示されます。一覧から選択すると、画面の一番上の「ブックマークバー」から選択した場合と同じ効果があります。
-  (検索)：「編集」→「検索」→「検索」を選択すると、Safari の右上にこの窓が表示され、ここで文字列を入力して **return** キーを押すと、そのページ内の文字列を検索して、同じものがあればそこにカーソルが移動して、内容を表示します。

② ブックマーク

頻繁にアクセスするページについてはURL入力の手間を省きたいものです。そのためには、そのページを開いた状態で、「ブックマーク」→「ブックマークに追加」を実行します。フォルダーも作成できるので、全部同じ水準に保存するのではなく、内容によって、整理して登録すると良いです。ブックマークバーに登録したページを開くには、Safariの画面上端のブックマークバーに登録されているURLをクリックすると、直接移動することができます。

をクリックすると、登録しているブックマークの一覧が表示されるので、ブックマークとして登録されているURLをダブルクリックすると、そのURLに移動することができます。一覧が表示されている状態で、個別のブックマークをドラッグすると場所を調整することができますので、順番を整理できます。またControlキーを押しながらクリックするとメニューが表示されて、削除することもできます。

お気に入りを他のパソコン（例えば自宅のパソコン）などに移すことができます。そのためには「ファイル」から「ブックマークを書き出す」を選択します。HTML形式でブックマークが出力されるので、このファイルを別のパソコンに移してクリックすると、ブックマークの一覧がブラウザで表示され、選択することができます。

③ 印刷

必要な情報が見つかったら印刷して持ち歩く必要があるかもしれません。表示されているページを印刷するには「ファイル」→「プリント」で印刷できます。「プリント」の窓で、プリンタの選択、印刷するページ範囲の選択、印刷部数などを選択できます。ページによっては、例えばA4の縦に収まらない形式のものがあり、そうしたページを印刷すると右が切れたままで役に立たないことがあります。そこで、印刷する前には印刷の窓の左側に表示されているプレビューを見て、問題がないかどうかを確認する癖をつけましょう。もしはみ出たり、小さすぎたりする場合には、拡大縮小で調整することができます。あるいはA4の縦にきちんと収まらない時は「方向」を「縦」から「横」にします。これで大抵は大丈夫です。

紙に印刷するのではなく、からPDFファイルとして保存することもできます。PDFファイルに保存する場合には「セキュリティオプション」で、文書ファイルを開く際、文字列をコピーする際、印刷する際にパスワードを要求するように設定することができます。できるだけ不要なページは印刷しないようにしましょう。

④ ページのファイル保存

表示しているページは実はいつまでもあるとは限りません。もう二度と見る可能性がなければ特に気にする必要はありませんが、重要な情報で、論文や報告書を書いているときの資料などとして、後日、改めて見る可能性がある時は手元にファイルとして保存しておかないと、二度と見るができなくなる場合があります。表示しているページを保存するためには「ファ

イル」→「別名で保存」をクリックします。この時に保存形式としては「Web アーカイブ」と「ページのソース」の2種類があります。前者ではイメージも含めて完全に保存されますが、後者はHTML形式で保存され、テキストだけで、イメージは保存されません。

⑤ 文字コード

文字はコンピュータの内部では0と1の並びで表されています。英字は8ビットが一単位(1バイト)で表示され、日本語文字は16ビット(2バイト)を一単位で表示します。この文字の体系を文字コード(コードとは符号の意)と呼びます。日本語文字を表す文字コードとしては、次のものがあります。

- EUC-JIS : UNIX や Linux で普通に使われる。
- SHIFT-JIS : パソコンでよく使われる。
- JIS : 通信などで標準的に使われる。
- UTF-8 : UTF は Unicode Text Format の略で、最近の Windows など、多言語化の一環として使用されています。

WWWのページを記述するときには、どの文字コードを使うかをページの先頭で明示することになっていますが、それが正しく記述されていないことがあります。その場合、使用するブラウザによって想定されている文字コードと中身の実際の文字コードが一致しないために文字化け(読めない)という現象が起ります。このときは、「表示」→「テキストエンコーディング」→(適当と思われるコードの選択)の順にクリックすれば、正しい表示ができます(ことが多い)。もしこれでうまくいかない場合には、「Safari」→「キャッシュを空にする」を選択して、キャッシュ(インターネットの一時ファイル)を消去してから再度、試みて下さい。

ファイルの漢字コードをどうするかで、一番問題になるのが、改行コードの問題です。改行をあらわす文字(コード)は次に示すように微妙に違います。

- UNIX では、LF (0x0a)
- Mac OS では、CR (0x0d)
- Windows では、CR+LF (0x0d, 0x0a の2バイト)

ここでCRとはCarriage Returnで、カーソルを行の左端に移動する操作を、LFはLine Feedでカーソルを一行下の行に移動する操作を意味しています。したがって厳密に言うとCR+LFをセットで使うと次の行の左端にカーソルが移動することになります。

⑥ 画像の保存

表示しているページは文字と画像で構成されていることがあります。そのページに含まれている画像だけを保存したい場合には、ページの保存したい画像上でControlキーを押しながらマウスでクリックすると、図6-3に示すようにイメージで保存するためのメニューが表示されます。

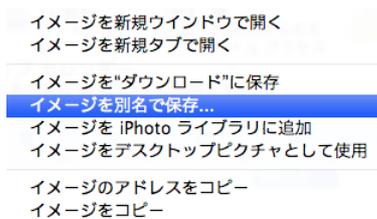


図 6-3 画像の保存メニュー

なお、保存した画像には著作権、肖像権、公衆送信権など、様々な権利がありますので、使い方には十分に注意してください。

6.4. 設定の変更など

① 起動時の動作

「Safari」→「環境設定」からブラウザの様々な動作の設定を行うことができます。図 6-4 に、起動した時に表示する Web ページ、ダウンロードファイルの保存先などを設定する「一般」の画面例を示します。「一般」の「ホームページ」では、起動時に表示したいページの設定ができます。みなさんの場合には <http://www.s.kyushu-u.ac.jp> が設定されていると思いますが、これを変更することで、Safari を起動した時に表示される画面を別のものに変更することができます。「現在開いているページを指定」にすると、現在表示しているページが次回からの起動時に表示されるようになります。例えば Gmail を頻繁に使用すると思えば、Gmail のログイン画面を表示して、「現在開いているページを指定」にすると、Safari を起動した時にいきなり Gmail を読むためのログイン画面になります。

「セキュリティ」を選択すると、図 6-5 に示すように、詐欺サイトの排除、Web コンテンツの表示に関連して Java、JavaScript、プラグインの使用の是非、Cookies の取り扱いなどを指定することができます。

② 一時ファイル（キャッシュ）の削除

「インターネット一時ファイル」（キャッシュ）はブラウザで表示するために転送してきたページの内容を手元のパソコンに蓄積したものです。こうすることによって、同じページを何度も表示しようとした時に元々のサイトまでページの内容を改めて取りに行く必要がなくなり、すばやく表示できるようになります。昔、インターネットの帯域が十分になかったときには重宝した機能でした。しかしながら、キャッシュがあまりにたくさん蓄積されると、ディスクの使用容量が増える、あるいはキャッシュを探すのに時間がかかるようになるなど、いくつかの弊害もあります。そこで、時々、「Safari」→「キャッシュを空にする」を選択してキャッシュを削除すると、ディスクを空けることができ、また軽やかに動作するようになります。

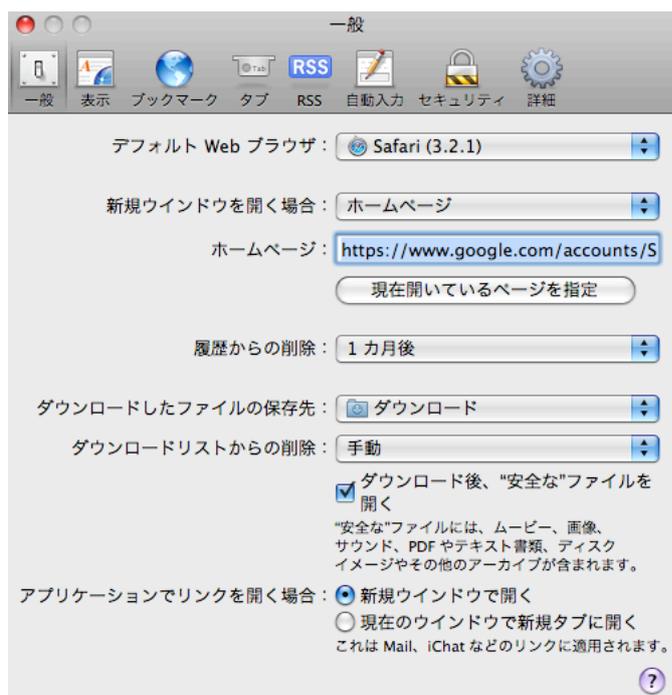


図 6-4 Safari の環境設定の「一般」

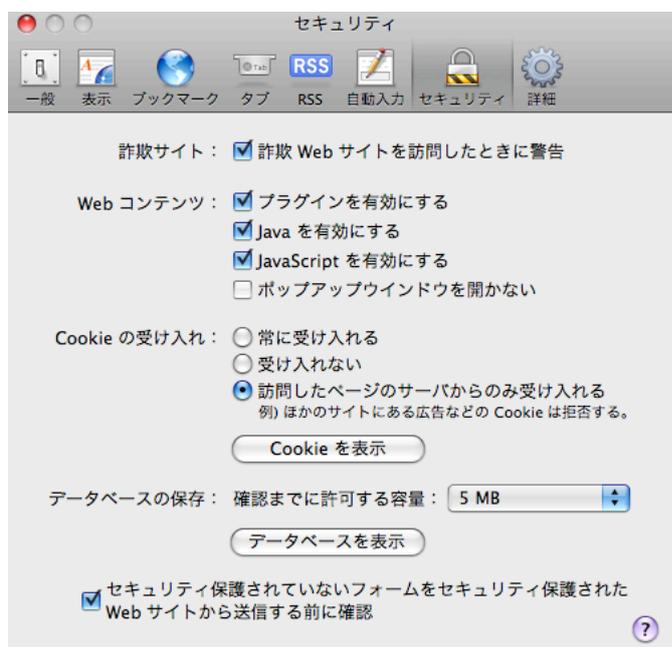


図 6-5 Safari の環境設定の「セキュリティ」

③ クッキーの削除

オンラインショッピングなどを行うと連続した通信であることを確認するためにクッキー (Cookie) と呼ばれる機能を使用します。また、クッキーを利用するとログイン名やパスワード

ドを自動的に補足してくれるなど、便利なこともあります。重要な個人情報が記録されたままになり、外部から不法に読み出される可能性があります。したがって、重要な個人情報を使った通信を行った後では、この「Cookie」を削除しておいた方が安全です。「環境設定」→「セキュリティ」で、「Cookies を表示」をクリックし、Cookies の一覧を表示し、どれかを選択して「取り除く」をクリックすると選択した Cookie を削除できます。あるいは画面下の「すべてを取り除く」をクリックすると、すべての Cookies を削除することができます。

④ 履歴の削除

Safari はこれまで何処をアクセスしたかという履歴を記憶しています。この履歴を消すタイミングは「環境設定」→「一般」の「履歴からの削除」で設定されています。自動的に削除するようにしておくか、手動で自分で意識的に削除するか、どちらにするかを決めます。自分の意志で履歴を削除するためには、「履歴」から「履歴の消去」を実行すると、すべての履歴を消去し、何処をアクセスしたかを他人に知られることがなくなります。ただし、この履歴の機能が働いていると、「アドレス」部分に URL を入力する時に、候補を表示してくれるので、個人で占有して使用しているパソコンなどでは、その候補の中から必要な URL を選択することで、楽をすることができます。

⑤ パスワードなどの記録

「環境設定」→「自動入力」で、「ユーザ名とパスワード」にチェックをつけておくと、ユーザ名とパスワードを記憶してくれ、次回から入力しなくても済むようになります。ただし、この設定を有効にしておくと、ユーザ名やパスワードをプログラムが記憶してしまい、他の人が勝手にログインできるようになったりするので、注意が必要です。インターネットカフェなどで、この機能でパスワードなどを記憶させる状態では絶対に使用してはいけません。

6.5. 知っているの良い機能

① 文字列のコピーと貼り付け

表示されているページの中から一部または全部の文字列をクリップボード(一時的な保管場所)にコピーして、その文字列を別の場所に張り付けることができます。例えば表示しているページのコピーしたい部分をマウスでドラッグすると、その部分が選択されます。このとき、文字列の表示が反転するので、どこが選択されているかを確認することができます。選択し終わったらクリックを解除し、「編集」→「コピー」を選択するか、Control キーを押しながらマウスをクリックして「コピー」を選択するか、Command+C をキー入力すると、選択した文字列をクリップボードにコピーすることができます。クリップボードに取り込まれている文字列は「編集」から「ペースト」を実行するか、Control キーを押しながらマウスをクリックして「ペースト」を選択するか、Command+V キーを入力することでカーソルがある場所に

張り付けることができます。なお、ページ全体を選択したいときは、「編集」から「すべて選択」でも良いです。

② 文字列の検索

表示されているページの中から特定の文字列（用語など）を探したいときは、「編集」→「検索」を実行すると、検索用のウィンドウ  が開きます。そこで検索したい文字列を入力してから return キーを押すと、文字列を検索して、見つかったら文字列の表示が反転するので、大量の文章の中で特定の文字列を容易に探すことができます。

③ ソースコードの表示

WWWのページはHTML (Hyper Text Mark-Up Language)という言語で記述されています。あるページが具体的にどのように記述されているかを知りたいとき、「表示」→「ソースを表示」で、その Web ページのソーステキストを表示することができます。これで、どのような書き方がなされているかを知ることができ、HTML の記述の仕方を勉強することもできます。ただし、アプリケーションプログラムが生成した HTML のページはやたらと余計な修飾が行われていて、とても読みにくいことがあります。例えば Microsoft PowerPoint で作成したファイルを Web ページとして保存した場合にソースを表示してみてください。

④ 表示文字の大きさ

表示しているページの文字の大きさを変更することができます。「表示」→「文字を拡大」か「文字を縮小」で、読みやすい大きさになるように設定します。文字が小さすぎて読めない場合には役に立ちます。ただしページの書き方としてフォントの大きさを絶対値で指定してあるページでは効果がない場合があるようです。ページを印刷する時にも文字の大きさは影響します。「表示」→「文字を標準サイズにする」で元に戻すことができます。

6.6. 検索エンジンの使い方

① WWW 情報の特徴

インターネットは情報の宝庫です。ありとあらゆる情報がインターネット上に存在します。おかげで最新の情報（製品、ニュース、経済活動など）をオンラインで簡単に検索できます。あるいは古い情報も検索することができます。一方、インターネットはゴミの山でもあります。WWW では、発信コストが低い分、全体としては情報の質が低下します。新聞や TV で情報を発信するコストと比べて見てください。研究という視点から見ると、ありとあらゆる研究に無関係な情報がインターネット上に存在します。家族やペットの紹介は本人には大事でも、研究には普通は全く役に立ちません。その逆もあり得ます。

インターネットから得られる情報は次のような特徴を持っています。

- 国境がない。（国内、国外を問わず情報を発信、入手できる）

- 情報の入手に時間がかからない。
- 発信も受信もコストが安い。
- オンラインで検索できる。したがって、再利用も容易である。
- 最新のデータを自分で保持する必要がない（各種統計データなど）。ただしものによっては、その情報がいつなくなるか分からない。

② 検索エンジン

こうした情報を上手に使いこなすためには、情報の検索の仕方に「コツ」が必要です。検索したいキーワードが明確な場合には検索エンジンを活用して速やかに検索することができます（電話帳型検索）。一方、ぼんやりした検索（最近の XX の傾向など）では、カテゴリ別に分類してある WWW サイトを活用すると良いでしょう（ガイドブック型検索）。いずれにしても「検索エンジン」と呼ばれる仕組みを利用して必要な情報を探します。これがインターネットの世界を大きく変えました。

インターネットには様々な情報が存在しますが、それらの情報が何処にあるかが分からないと役に立ちません。必要な情報が何処にあるかを調べるための機能が「検索エンジン」と呼ばれている機能で、利用者からの要求にしたがって情報を調べて存在場所の URL を教えてくれます。この検索エンジンはその仕組みから二種類のものがあり、「ディレクトリ型」と「ロボット型」と呼ばれています。最近では双方の長所を取り入れた複合型も増えています。

「ディレクトリ型」というのは世界中の WWW サイトの情報を入手した上で、人がある約束ごとにしたがって整理し、利用者から探しやすく配置しているものです。有名なものに Yahoo があります。一方、「ロボット型」と呼ばれるものは世界中の WWW サイトから情報を収集し、それらの中からキーワードを抽出して、データベースに登録しておき、利用者からの要請に応じてデータベースを検索し、検索に指定されたキーワードを含むページを表示するものです。有名なものとして Google があります。こちらは実物をアクセスしてもらえば (<http://www.google.co.jp/>)わかるようにキーワードを入力するだけで、Yahoo のようにメニューがありません。データベースを検索した結果をどのような順番で表示すると利用者の希望に添えるかが重要で、様々な方式が研究され、実用化されています。例えば、たくさんのリンクが張られているページは値打ちがあるとかいった評価尺度で順番を決めるわけです。

特定のキーワードを含むページを検索するには検索エンジンを使用すると便利です。ここでは Google を例にキーワードを使った検索方法を説明します。

- キーワードを複数指定するときには語と語の間を半角の空白（スペース）で区切ります。Google ではすべての単語を含むページを表示します。
- 指定したキーワードで表示された結果で、さらに絞り込をする場合にはキーワードを追加して、検索します。

- 特定のキーワードを含まないページを検索したい時はキーワードの先頭に- (マイナス)をつけて検索します。
- キーワードを複数指定すると、すべてのキーワードを含むページが表示されますが、順番は無関係になります。したがって特定のフレーズを検索したい時は "Love is over"のようにダブルクォートではさんで指定します。

詳細は Google のトップページの下の方にある「Google について」をクリックし、「検索方法とヘルプ」を参照してください。基本検索、"and" 検索、ストップ語については理解してください。検索した結果のページサイズが大きくて、指定したキーワードが何処にあるか探しにくい時は、ページ内の文字列の検索を利用しても良いですが、検索結果が表示されている一覧で、「キャッシュ」という表示がありますから、これをクリックすると指定したキーワードに色がついて表示されるので、分かり易くなります。

③ 古い情報

昔のページで元々の場所にはなくなってしまった場合でも、主要なページについては次のような方法で見ることができる場合があります。

- 比較的最近のページの場合には Google などで検索して、「キャッシュ」をクリックします。こうするとオリジナルが既に消去あるいは編集されていても、見ることができる場合があります。
- 古いページを専門に保存しているサイトがあります。次の URL で探して見ると、意外に昔のページがあるかもしれません。

<http://www.archive.org/>

ここで www.kyushu-id.ac.jp と入力して、Take Me Back をクリックすると、昔々の九州芸術工科大学のホームページを見ることができます。例えば次のようなものです。

<http://web.archive.org/web/19961221022727/http://kyushu-id.ac.jp/>

国会図書館も 2100 ぐらいのサイトの過去のページを収集しており、九州芸術工科大学や九州大学の昔のページを見ることができます。

6.7. 使用上の注意

WWW で提供されている情報が信頼できるものである保証は全くありません。場合によっては内容が信頼できない場合、あるいは故意に詐欺を働く目的で準備されているものもないとは言えません。表示されている情報を信じるか、信じないかは、最終的には個人の判断によります。表示されたページがどんなにすばらしく見えても、若干の手間と費用をかければどんなペ

ージでも作れることを考えれば、ページと運営母体の組織の信頼性とは無関係と置いていたほうが安全です（特にオンラインショッピングなど）。さらには、セキュリティ上安全でないサイトの場合もあります。怪しそうなサイトには近寄らないようにしましょう。

授業の課題などを準備する時に、最近では皆さん、インターネットを使って調べてから課題を作成するケースが多いというか、そればかりする人が多いような気がします。しかしながら Google などの検索エンジンで検索した結果をそのままコピーして張り付けることは著作権法に違反する行為です。著作権法に違反すると、最大、500 万円以下の罰金、and/or 5 年以下の懲役、さらに皆さんが著作権法に違反すると所属している九州大学に 1 億 5 千万円までの科料が課せられます。

他人の著作を引用する場合には作法があります。ここからここまでを引用したということが誤解の余地なく明確にわかるようにすること、それが誰のどの著作のどの部分からの引用であるかをわかるようにきちんと明示することです。また引用と言うからには分量も大事で、著作のほとんど全部を引用というようなことはありえません。出典の記述の仕方は分野によって微妙に異なりますが、論文の場合には、著者名、論文のタイトル、出典（論文誌名）、巻、号、ページ数、発行年などを記述します。ほかの資料を引用する場合にはこれに準じて、誰でも関心がある人が引用元を直接参照することができるだけの情報を表示することが大事です。

URL を引用先としてあげる場合にそれが「いつ書かれたもの」で、「いつまで見えるか」について、注意をすることが大事です。また誰が書いたかにも注意を払ってください。WWW では、新聞やTVなどと異なり、極めて低コストで情報を発信することができるため、内容に対する吟味が不十分なものが多いと思ってください。新聞記事でも本当に大事な記事には書いた記者の名前が入っていることを知っていますか。

先生は皆さんのレポートを日々、見続けているので、レポートをちょっと見ると画面から貼り付けたものであるかどうかをたちどころに判断することができます。そもそも文体、言い回し、使用する単語の類が、皆さんが日頃使っている、あるいは使えそうな範疇とは、かけ離れています。内容を理解し、自分の言葉できちんと書き下してください。過去に書かれたページで「〇〇年〇〇月に予定されている」という（その時は未来だった）日時が現在では既に過去だったりする例などは笑い話にもなりません。皆さんのレポートの一部を Google で検索すると該当ページがそっくりそのまま出てくるなどというのは論外です。画面の内容をそのまま貼り付けた場合には、最悪、カンニングと同じ扱いをすることがあります。

6.8. WEB ページについて

現在の Web ページは HTML (Hyper Text Markup Language)によって書かれています。HTMLは、Web 上でドキュメントや写真、音声、サービスなどを提供するために開発された言語

です。ワープロでは、文字の大きさや色などの属性を持っていて、それらを指定することで、様々な表現ができますが、HTMLは、このような属性指定をする言語です。現在広く使用されているバージョンは1999年12月に発表された4.01 (<http://www.w3.org/TR/html401/>) でしたが、2009年4月にHTML5が発表されました。

HTMLでは、タグと呼ばれる記号を利用して、ドキュメントに様々な意味づけ（フォントの変更、リンク、画像の表示、プログラムの起動など）を行います。

例)

- 簡単なタグの例

タグ: 開始タグ<TAG名>と終了タグ</TAG名>
<h1> 私のホームページ </h1> (見出し1)

6.9. HTML について

ホームページを作成するソフトウェアも出回っていますが、HTMLのタグを正しく理解できる知識がないと、きちんとしたホームページを作成するのは容易ではありません。一度はHTMLをきちんと勉強しましょう。

① HTML のタグ

全体の構造 HTMLの全体は図 6-6に示す構造をしています。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
  <head> <title> … </title> </head>
  <body> … </body>
</html>
```

図 6-6 HTML の基本的な構造

この構造で実際に中身が記述されると例えば図 6-7に示すような内容になります。

- ② 見出しのHeaderに書くおまじないは次の通りです。文字コードとしてUTF-8を指定する例です。もしもシフトJISを使用する場合にはUTF-8をSJISにします。

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

- ③ バックグラウンド色

画面に表示されるページの背景の色を指定する時は次のように指定します。

```
<body bgcolor="#FFFFFF">
```

16進表記で左から8ビット毎にRGBを示しています。したがって、全部のビットが立っている状態では白になります。16進表記でなく、色の名前をredやblueのように直接指定しても良いのですが、微妙な色合いの単語を知らない場合が多いので、困ることがあります。

図 6-7 に実際の Web ページの例を示します。

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
  <meta content="text/html; charset=Shift_JIS" http-equiv="content-type">
  <title> My Home Page </title>
</head>
<body>
<h1>私のホームページへようこそ</h1>
  <h2>お気に入りリンク集</h2>
  <ul>
    <li> <a href="http://www.google.co.jp"> Google </a> </li>
    <li> <a href="http://www.yahoo.co.jp"> Yahoo </a> </li>
    <li> <a href="http://webct.kyushu-u.ac.jp/">WebCT</a> </li>
  </ul>
</body>
</html>
```

図 6-7 簡単な Web ページの例

④ 簡条書き

⑤ 簡条書きをするには次のような3種類の書き方があります。最初の例は図 6-8に示すように単純な簡条書きです。

基本形 :

```
<ul>
  <li> 1 番目の項目</li>
  <li> 2 番目の項目</li>
</ul>
  <ul>
    <li> 1 番目の項目</li>
    <li> 2 番目の項目</li>
  </ul>
  <li> 3 番目の項目</li>
</ul>
```

- 1 番目の項目
- 2 番目の項目
 - 1 番目の項目
 - 2 番目の項目
- 3 番目の項目

(b) 表示の例

⑥ ソースの例

図 6-8 単純な簡条書きの例

次の例は図 6-9に示すように通し番号がつく箇条書きの例です。

基本形 : ` `

```
<ol>
  <li> 赤いリンゴ</li>
  <li> 青いリンゴ</li>

  <ol>
    <li> 青い海</li>
    <li> 白い空</li>
  </ol>

  <li> 黄色いバナナ</li>
</ol>
```

(a) ソースの例

```
1. 赤いリンゴ
2. 青いリンゴ
   1. 青い海
   2. 白い空
3. 黄色いバナナ
```

(b) 表示の例

図 6-9 通し番号付きの箇条書きの例

最後の例は図 6-10に示すようにカタログの仕様一覧を書く場合に使うような箇条書きです。

基本形 :
`<d1>`
`<dt> xx </dt><dd> yy </dd>`
`</d1>`

```
<d1>
  <dt> 1 番目の項目</dt><dd> 1 番目の説明</dd>
  <dt> 2 番目の項目</dt><dd> 2 番目の説明</dd>
  <dt> 3 番目の項目</dt><dd> 3 番目の説明</dd>
</d1>
```

(a) ソースの例

```
1 番目の項目
  1 番目の説明
2 番目の項目
  2 番目の説明
3 番目の項目
  3 番目の説明
```

(b) 表示の例

図 6-10 項目別箇条書きの例

⑤ 表

表を書くにはtableタグを使います。表の幅、枠線の表示・非表示、セルの幅の調整、右詰め、中央揃えなど色々なパラメータがあります。具体的な例を図 6-11に示します。

基本形 :

```
<table border width=75%>
  <tr><th> hh </th> <td> xx </td> <td> yy </td> </tr>
</table>
```

```

</table>
<table border=1 width=60%>

<tr><th> 1 番目の見出し</th><td align=left> 1 番目の内容</td></tr>
<tr><th> 2 番目の見出し</th><td align=center> 2 番目の内容</td></tr>
<tr><th> 3 番目の見出し</th><td align=right> 3 番目の内容</td></tr>

</table>

```

(a) ソースの例

1 番目の見出し	1 番目の内容
2 番目の見出し	2 番目の内容
3 番目の見出し	3 番目の内容

(b) 表示の例

図 6-11 表の例

⑥ 文字飾り

ブラウザで表示する時に、表 1や表 2に示すように、文字の種類、飾り、色、大きさなどを指定することができます。

表 1 文字飾りの例

ソース	表示
<i>Italic</i>	<i>Italic</i>
Bold	Bold
<tt>Typewriter</tt>	Typewriter
<S>Strike String</S>	Strike String
青色	赤色
X²	X ²
H₂O	H ₂ O
<marquee>SCROLL</marquee>	SCROLL

表 2 文字の大きさの指定

ソース	表示
<pre> 大きさ= 1
 大きさ= 2
 大きさ= 3
 大きさ= 4
 大きさ= 5
 大きさ= 6
 大きさ= 7
</pre>	<p>大きさ=1 大きさ=2 大きさ=3 大きさ=4 大きさ=5 大きさ=6 大きさ=7</p>

⑦ 表題の指定

文章中にメリハリをつけるために、大きさや位置を指定して、見出しを付けることができます。表 1に見本を示します。

基本形 : `<h1> 見出し</h1>`

表 3 見出しの大きさ

ソース	表示
<pre><h1> 大きさ1の見出し</h1>
 <h2> 大きさ2の見出し</h2>
 <h3> 大きさ3の見出し</h3>
 <h4> 大きさ4の見出し</h4>
 <h5> 大きさ5の見出し</h5>
 <h6> 大きさ6の見出し</h6>
</pre>	<p>大きさ1の見出し 大きさ2の見出し 大きさ3の見出し 大きさ4の見出し 大きさ5の見出し 大きさ6の見出し</p>

⑧ 改行、段落

HTMLで記述すると、空白、タブ、改行などは無視されます。したがって改行したい場合には表 4に示すように、明示的に改行を指示する必要があります。その時に右詰めなどの横の位置を指定することができます。

基本形 : `<p>テキスト</p>` 改行して、一行空ける。

改行する。

表 4 段落と改行の指定

ソース	表示
<pre><p align=left> テキストを左詰めにする。 </p> <p align=center> テキストを中央揃えにする。 </p> <p align=right> テキストを右詰めにする。 </p></pre>	テキストを左詰めにする。 テキストを中央揃えにする。 テキストを右詰めにする。

単純に中央揃えにするには `<center>テキスト</center>` とする方法もあります。また行の途中でどうしても少し隙間を空けたい時は、"` `" (アンド、nbsp、セミコロン) を使うと隙間を空けることができます。

⑨ 他へのリンク

WWWの特徴の一つであるクリックすると別のURLへ移動するという機能は次に示すようなタグを使用します。hrefの次に書かれる内容が同じコンピュータのファイル、別のコンピュータのファイル、イメージなど、書き方によって利用者から見た時の動作が変わります。またtarget=を記述すると、別の窓を開いて表示するようにできます。

基本形 : ` 他のファイルへのリンク `
`HTML用の別の窓<p>`
` 他のコンピュータへのリンク `
` 別窓を開いて表示`
`` イメージ (写真) の張り付け
``

⑩ 下線を引く

下線を引くには<hr>タグを使います。太さ、横位置、幅などを指定することができます。

基本形 : `<hr>` `<hr size=10pt>` `<hr width=50% align=left>`

⑪ そのまま表示

HTMLで記述したテキストは、改行、空白、タブなどは基本的に無視 (詰めて) 表示されます。したがってプログラムリストなどをそのままの形式で表示させたい時には、ここで示す<pre>タグを使います。そうしないとテキストがグチャと詰まって表示されます。

基本形 : `<pre>` プログラムなど `</pre>`

次に示すようなプログラムを表示する時に<pre>を使う場合と使わない場合の違いを表 5に示します。

```
public class app
{
    public static void main(String[ ] args)
    {
        System.out.println("Hello from Java!");
    }
}
```

表 5 <pre>タグの効果

<pre>の有無	表示
なし	<code>public class app { public static void main(String[] args) { System.out.println("Hello from Java!"); } }</code>
あり	<code>public class app { public static void main(String[] args) { System.out.println("Hello from Java!"); } }</code>

6.10. ホームページの作成

教育情報システムにログインし、Finderで自分のホームディレクトリを見ると `public_html` というフォルダ（ディレクトリ）があることが分かります。この下に移動して、`index.html` という名前のファイルを作成すると、例えば、`http://www.s.kyushu-u.ac.jp/~利用者ID/` というようなURLで、自分のホームページとして見るすることができます。

ここでは図 6-12 自分のホームページの例に示すような内容の自分のホームページを作成して見ましょう。これをブラウザで表示すると、図 6-13に示すようなページとして表示されます。この例で、利用者IDは自分のログイン名に適宜置換えて下さい。

```
<html>
<head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<title>HTMLの見本</title>
</head>

<BODY BGCOLOR="#ffffc0">

<font size=4>

<h2>藤村直美</h2>

私の趣味は久住登山です。
<p>

</body>

</html>
```

図 6-12 自分のホームページの例



図 6-13 ホームページの例

6.11. ホームページ作成上の注意

ホームページを作成する上で注意すべきことを次に示します。

- 自分が著作権を所有しないものを載せてはいけません。写真、映像、音楽など、他の人が権利を所有しているものを公開すると法律に違反し、損害賠償を請求されることもあります。
- 他人の情報を無断で公開してはいけません。また他人のプライバシーを侵害したり、誹謗中傷する内容も掲載してはいけません。
- 公序良俗に反する内容を掲載してはいけません。
- 個人の名前はともかく、写真、電話番号、住所など、個人的な情報を公開すると、それらの情報は世界中の人が見ることができるので、何が起るか十分に考えて下さい。他の人に URL を教えていないから大丈夫とは思わないで下さい。ロボット型の検索エンジンのように世界中で自動的に情報を収集する仕組みが動いているので、WWW で公開した情報はたちまち誰かの目にとまることになります。一旦、インターネット上に流出した情報を完全に抹消することは現実的には不可能です。かならず何処かにコピーやキャッシュの形で残るので、未来永劫消えないと思ってください。
- 九州大学の教育情報システムは国民の税金で運用されているシステムです。教育・学習目的以外に使用してはいけません。