
情報教育コース フリーソフトの活用

情報科学部 知能情報学科 仲 隆

今日の内容

今日のコンセプト

- フリーソフトしか使わない
- 外部のサーバを使わない
- ダウンロード&インストールから

■ Perl入門(2コマ)

- プログラミング言語(スクリプト記述言語)
- テキストの一括処理に便利
- 正規表現が使える

■ Webアンケート入門(2コマ)

- PerlとCGIの応用
- Webページを利用したテストやアンケート
- オンラインテストの紹介

Perl入門

Perl関連フリーウェアのインストール

Perlの簡単な使い方

Perlによる実用的で簡単なプログラム

Perl関連ソフトウェアのインストール

■ Perlの特徴

- プログラミング言語である。
- 基本的にフリーで入手できる。
- 実際に役に立つプログラムがわずかな命令で書ける。
- OSの機能を使いやすく改良できる。
- CGIスクリプトを組めばインターネットのWebページで対話式プログラムも作ることができる。

フリーソフトの利用方法(1)

1. プログラムの取得

- インターネット(ウェブブラウザ, FTPクライアント)
- 雑誌の付録のCD-ROMなど

2. 事前準備

ウィルスチェック

インストール方法を読む.

バージョンアップの場合は, まず, 古いバージョンのアンインストール

解凍(圧縮されていない場合は省略)

- 圧縮ファイルをダブルクリック

フリーソフトの利用方法(2)

3. インストール

- インストーラが付属している場合(+Lhacaなど)
インストール用ファイルをダブルクリック
通常, C:¥Program Filesの下にインストールされる.
スタートメニュー(クイック起動, デスクトップ)に登録される.
- インストーラが付属していない場合(asperlなど)
インストール用フォルダを作成する. C:¥Programs
解凍して作成されたフォルダをその下に移動する.
デスクトップにエイリアスを作成する.

4. 動作確認

演習：+Lhaca (圧縮・解凍ツール) のインストール

- ダウンロード
 - 保存用フォルダの作成
 - マイドキュメント/Archives
 - フリーソフト紹介のウェブページへアクセス
 - アーカイバ(圧縮・解凍) : +Lhaca / local をクリック
 - Lhaca075.EXE をクリック
 - マイドキュメント/Archivesに保存 ウェブブラウザ終了
- 解凍 & インストール
 - Lhaca075.EXE をダブルクリックして実行
 - インストール先を確認してOK
 - readme.txtが開くので適用に読んで閉じる
- 初期設定
 - デスクトップの+Lhacaをダブルクリックして起動
 - 関連付け(LZHとZIP) をクリックして, OK

Perl関係フリーウェアのインストール

■ 対象

- ActivePerl (Perl処理系)
- asperl (Perl用テキストエディタ)
- Drop on Script (ドラッグ&ドロップで実行可能にする)
- 演習で使うサンプルファイル

■ 手順

- ダウンロード
- 解凍&インストール
- 動作確認&初期設定

演習：ダウンロード(Perl関係)

- フリーソフト紹介のウェブページへアクセス
 - www.is.kyusan-u.ac.jp/~naka/ IT講習会：お気に入りに登録
- ActivePerl(Perl処理系)
 - 言語処理系：[ActivePerl](#) / localをクリック
 - [LANGUAGES/ActivePerl NEW](#) をクリック
 - [DOWNLOADS/ActivePerl Free Distribution](#)をクリック
 - [Next](#)をクリック
 - Windows / [MSI](#)をクリック
 - [マイドキュメント/Archives](#)に保存
- asperl(Perl用テキストエディタ)
 - テキストエディタ：[Perlを始めよう！\(asperl\)](#) / localをクリック
 - Version Up Informationの2006/01-25「[Perlを始めよう！](#)」をクリック
 - Download: [aspl2065.lzh](#)(274KB)をクリック
 - [マイドキュメント/Archives](#)に保存

演習：ダウンロード(Perl関係, 続き)

- Drop on Script(ドラッグ&ドロップで実行可能にする)
 - ドラッグ&ドロップ対応: [Drop on Script / local](#) をクリック
 - [Drop on Script](#) をクリック
 - [Drop on Script Ver. 0.3.1 FROT031.EXE \(142KB\)](#) をクリック
 - マイドキュメント/Archivesに保存
- 演習で使うサンプルファイル
 - サンプルファイル: [local](#) をクリック
 - マイドキュメント/Archivesに保存
- ウェブブラウザ終了

演習：解凍，インストール，動作確認， & 初期設定

- ActivePerl (Perl処理系)
 - [ActivePerl-5.8.8.817-MSWin32-x86-257965.msi](#) をダブルクリックして実行
 - Next "I accept the ..."をチェックしてNext Next Next Install Finish
 - Release Notesを適当に読んで閉じる。
- asperl (Perl用テキストエディタ)
 - [aspl2065.lzh](#)をダブルクリックして解凍。
 - インストール用フォルダ (C:¥Programs) を作成
 - デスクトップのaspl2065をインストール用フォルダに移動
 - デスクトップにasperl.exeのショートカットを作成
 - ショートカットasperlをダブルクリックして起動
 - コマンド/環境設定 使用するPerlの選択 C:¥Perl¥bin OK
終了

演習：解凍，インストール，動作確認， & 初期設定（続き）

- Drop on Script（ドラッグ＆ドロップで実行可能にする）
 - DROT031.EXEをダブルクリックして実行
 - 次へ 次へ 次へ インストール 完了
 - Drop on Script 環境設定
 - 拡張子 pl にチェック
 - 「適用」 「OK」 「終了」
- 演習で使うサンプルファイル
 - samples.zipをダブルクリックして解凍
 - 全てのファイルをマイドキュメントに移動

基本的使い方

- エディタ(メモ帳など)でプログラムファイルを作成(hello.pl)
 - `print "hello world.¥n";`
 - `print "こんにちは.¥n";`
 - `$now = localtime(time);`
 - `print $now;`
- コマンドプロンプトで実行
 - `perl hello.pl`

演習：perlの基本的使い方

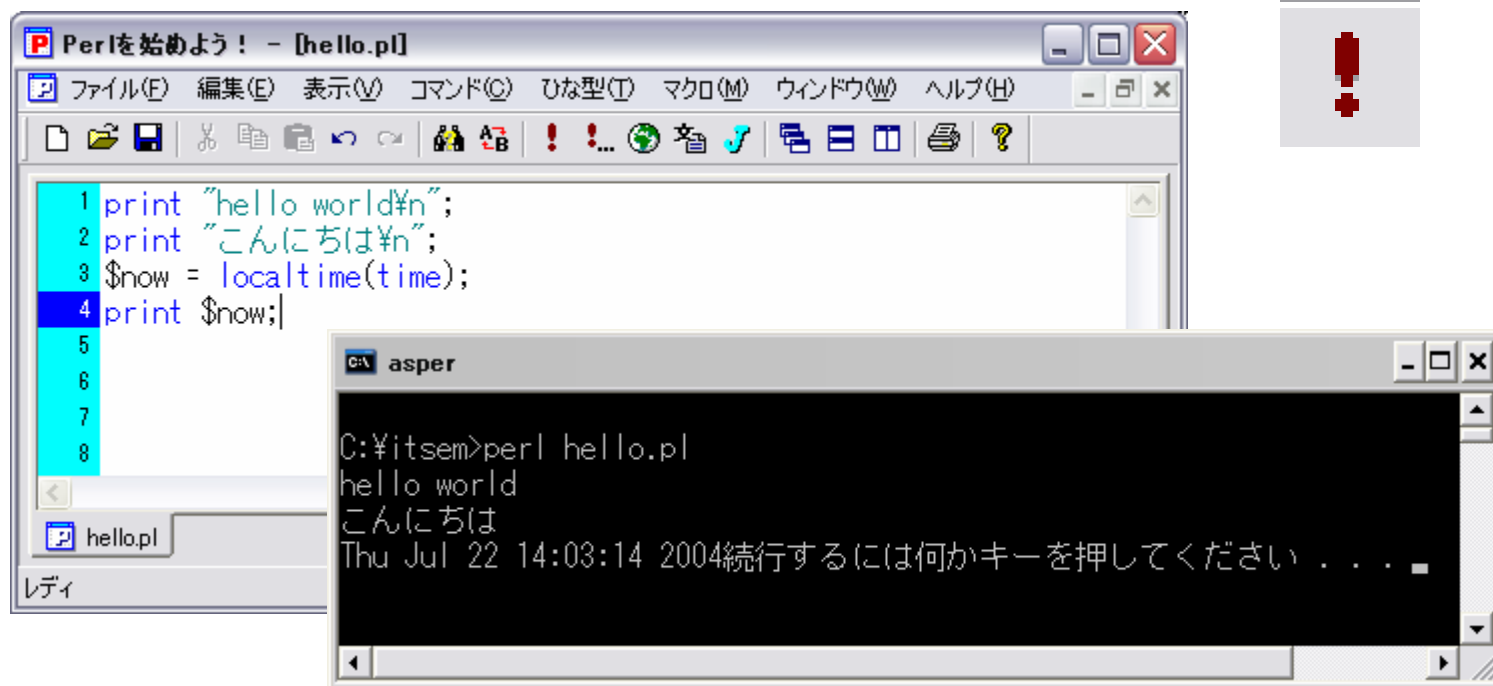
1. メモ帳を起動
2. 右のプログラムを入力
3. ファイル名“hello.pl”でマイドキュメントに保存
4. コマンドプロンプトを起動
5. 右のように、マイドキュメントへ移動
6. 右のように、プログラムを実行

```
print "hello world.¥n";  
print "こんにちは . ¥n";  
$now = localtime(time);  
print $now;
```

```
C:¥...>cd my documents  
C:¥...>perl hello.pl  
hello world.  
こんにちは .  
Thu Jul 22 12:42:57 ...
```

演習 : asperlの利用

1. hello.pl を asperl にドラッグ&ドロップ
2. [コマンド]の[実行],あるいはツールバーのびっくりマーク



Perlの機能

- 四則演算, 論理演算
- 文字列処理
- 配列
- ファイル入出力
- 制御構造 (if, while, for, ...)
- 正規表現 (パターンマッチ) によるテキスト処理
- データベース

演習：正規表現によるテキスト処理

1. books.txtをダブルクリックして中身を確認。
2. 確認したらメモ帳を終了しておく。

1)細胞生物学実験室便覧

Cell Biology, Four-Volume Set: A Laboratory Handbook, 3rd Edition

編著者: Celis, J. and Carter, N. Simons, K. Small, J.V. Hunter, T. Shotton, D.

出版社: Academic Press/Elsevier USA

発行年月: September-2005 ハードカバー 2500 pp. ISBN: 0121647307

国内定価 ¥38,860 特別価格(公費) ¥32,764 (私費) ¥29,785

2)ゼブラフィッシュ

The Zebrafish: Part 1, Cellular and Developmental Biology, 2nd Ed.

Methods in Cell Biology, Vol. 76

...

演習: ファイル内容表示 (view.pl)

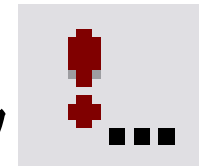
1. メモ帳を起動
 2. 右のプログラムを入力
 3. ファイル名“view.pl”でマインドキュメントに保存
 4. コマンドプロンプトを起動
 5. 右のように, マインドキュメントへ移動
 6. 右のように, プログラムを実行
- "\$_" は省略可

```
while(<>){  
    print $_;  
}
```

```
C:¥...>cd my documents  
C:¥...>perl view.pl books.txt  
1)細胞生物学実験室便覧  
Cell Biology, Four-Volume Set:  
A Laboratory . . .
```

演習 : asperlの利用 (view.pl)

1. "view.pl"をasperlにドラッグ&ドロップ(書いたつもり)
2. [コマンド]の[実行],あるいはツールバーのびっくりマーク



The screenshot shows a Perl IDE window titled "Perlを始めよう! - [view.pl]". The code editor contains the following Perl script:

```
1 while(<>){
2     print $_;
3 }
4
5
6
```

A dialog box titled "引数を指定して実行" (Execute with arguments) is open, showing the argument "books.txt" entered in the "引数を指定して下さ" (Specify arguments) field. The dialog has "OK" and "キャンセル" (Cancel) buttons.

Below the IDE, a command prompt window titled "C:\Programs2\asperl\asperl.exe" shows the output of the script:

```
7)サイトメトリー
Cytometry: New Developments, 4th Ed.
Methods in Cell Biology, Vol. 75
編著者: Darzynkiewicz, Z., Roederer, M., Tanke, H.J.
出版社: Academic Press/Elsevier USA
発行年月: November-2004 ハードカバー 920 pp. ISBN: 0125641702
国内定価 ¥25,340 ⇒ 特別価格(公費) ¥22,615 (私費) ¥20,559続行するには何かキー
を押してください . . . . .
```

演習：行番号の付加 (addno1.pl)

1. view.plを修正 (赤部分)
2. ! ...マークで実行
3. 行番号が付加されることを確認 .
4. ファイル名“addno1.pl”でマイドキュメントに保存
5. asperlを終了 .

```
$i = 0;  
while(<>){  
    print ++$i;  
    print ": ";  
    print $_;  
}
```

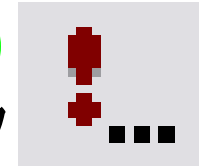
演習：行番号の付加 (addno1.pl)

1. コマンドプロンプトを起動.
2. 下のよう、マイドキュメントへ移動.
3. 下のよう、プログラムを実行.
4. nolist.txtをダブルクリックして中身を確認.
5. nolist.txtを削除

```
C:¥...>cd my documents  
C:¥...>perl addno1.pl books.txt > nolist.txt  
C:¥...>
```

演習 : asperlの利用 (addno1.pl)

1. “addno1.pl”をasperlにドラッグ&ドロップ(書いたつもり)
2. [コマンド]の[実行],あるいはツールバーのびっくりマーク



The screenshot shows a Perl IDE window titled "Perlを始めよう! - [add_lineno1.pl]". The code in the editor is:

```
1 $i = 0;
2 while(<>){
3     print ++$i;
4     print ": ";
5     print $_;
6 }
7
```

A dialog box titled "引数を指定して実行" (Execute with arguments) is open, with the text "引数を指定して下さい。" (Please specify arguments.) and a dropdown menu containing "books.txt > nolist.txt".

A terminal window titled "C:\Programs2\asperl\asperl.exe" shows the command:

```
C:\Home\Lectures\IT講習会\2006\samples\perl>perl add_lineno1.pl books.txt 1>nolist.txt
続行するには何かキーを押してください . . .
```

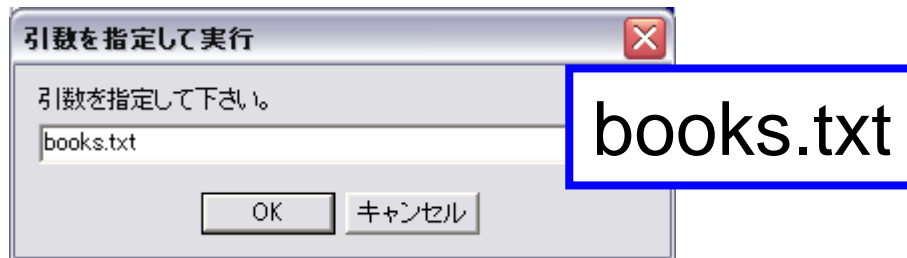
A blue box highlights the text "books.txt > nolist.txt" in the dialog box.

演習：行番号の付加 (addno1.pl)

1. nolist.txtが生成されていることを確認。
2. nolist.txtをダブルクリックして中身を確認。
3. nolist.txtを削除

演習：行番号の付加 (addno2.pl)

1. addno1.plを修正(赤部分)
2. ! ...マークで実行



```
$i = 0;  
while(<>){  
    print $i."": ".$_";  
}
```

3. 行番号が付加されることを確認.
4. ファイル名“addno2.pl”でマイドキュメントに保存

ピリオド(.)は文字列の連結の演算子

演習：行番号の付加 (addno3.pl)

1. addno2.plを修正(赤部分)
2. ! ...マークで実行
3. 行番号が付加されることを確認.
4. ファイル名“addno3.pl”でマイドキュメントに保存

```
while(<>){  
    print $_": "._$_.  
}
```

ダラーピリオド(\$.)には行番号が入っている

演習：行番号の付加 (addno4.pl)

1. addno3.plを修正(赤部分)
2. !...マークで実行
3. 行番号が付加されることを確認.
4. ファイル名“addno4.pl”でマイドキュメントに保存

```
while(<>){  
    print "$.: $_";  
}
```

変数は文字列(“...”)に入れてもよい

演習：パターンマッチ (pattern1.pl)

1. addno4.plを修正 (赤部分)
2. ! ...マークで実行
3. Cellを含む行が抽出されることを確認 .
4. ファイル名“pattern1.pl”でマイドキュメントに保存

```
while(<>){  
    if( /Cell/ ){  
        print;  
    }  
}
```

`/.../` はパターン `...` を含む行を表す

演習：パターンマッチ (pattern1.pl) 続き

1. pattern1.plを修正 (赤部分)
2. ! ...マークで実行
3. **行頭から始まるCell**を含む行が抽出されることを確認.

```
while(<>){  
    if( /^Cell/ ){  
        print;  
    }  
}
```

ハット(^) は行頭を表す

演習：パターンマッチ (pattern1.pl) 続き

1. 下のパタンのそれぞれについて同様に試してみる。

<code>/Cell/</code>	: Cellを含む
<code>/^Cell/</code>	: 行頭からのCellを含む
<code>/3/</code>	: 3を含む
<code>/[0123456789]/</code>	: 数字を含む
<code>/[0-9]/</code>	: 上と同じ
<code>/\d/</code>	: 上と同じ

演習：パターンマッチ (pattern1.pl) 続き

1. 右のパタンのそれぞれについて同様に試してみる。

<code>/[^0-9]/</code>	: 数字以外の文字を含む
<code>/^[^0-9]+\$</code>	: 数字を含まない. \$ は行末.
<code>/^\d+\$</code>	: 上と同じ
<code>/^\s+:\s/</code>	: 行頭からスペース以外が1つ以上あり, コロン (:), そして, スペース文字
<code>/^\d{4}:\d{2}:\d{2}.*ISBN:\d+\$</code>	: 発行年月日の行の形

Perlの正規表現 (/.../)の...に書けるもの

A	: Aという文字
ABC	: ABCという文字列
A+	: 1個以上のA A(A, AA, AAA, ...)
A*	: 0個以上のA A(, A, AA, AAA, ...)
.	: 1つの任意文字 (A, B, C, ...) (¥nを除く)
?	: 0または1つの任意文字 (, A, B, C, ...)
^ABC	: (行頭)ABC
ABC\$: ABC(行末)
[ABC]	: A,B,Cのいずれか1文字
[A-Z]	: A~Zのいずれか1文字

[^ABC]	: A,B,C以外の文字
[^A-Z]	: A~Z以外の文字
A B C	: AまたはBまたはC
¥w	: 英数文字かアンダーバー
¥W	: ¥w以外の文字
¥d	: 数字.[0-9]と同等
¥D	: ¥d以外の文字
¥s	: 空白文字
¥S	: ¥s以外の文字
¥b	: 単語の区切り
¥B	: ¥b以外の文字
¥nなど	: エスケープシーケンス

演習：パターンマッチ (pattern2.pl)

1. pattern2.plをasperlにドラッグ & ドロップして(書いたつもり)、中身を確認。
2. ! ...マークで実行して結果を確認。

```
while(<>){  
  if( /^発行年月:(¥S+)-(¥d+).*ISBN:(¥d+)$/ ){  
    print "$3,$2,$1¥n";  
  }  
}
```

(パターン)は\$数字で参照できる

演習：パターンマッチ (pattern3.pl)

1. pattern3.plをasperlにドラッグ & ドロップして(書いたつもり), 中身を確認.
2. ! ...マークで実行して結果を確認.

```
while(<>) { 変数を利用した行をまたがる情報の収集  
    if( /^¥d¥)(.+)$/ ){ $title = $1; }  
    elsif( /^発行年月: ¥S+-(¥d+).*ISBN: (¥d+)$/ ) {  
        $year = $1; $isbn = $2; }  
    elsif( /^国内定価 ¥¥([¥d¥,]+)/ ) {  
        print "$year,$title,$isbn,$1¥n"; } } }
```

演習：ドロップレットの作成 (droplet1.pl)

1. droplet1.plをasperlにドラッグ&ドロップして(書いたつもり)、中身を確認。
2. books.txt を droplet1.pl にドラッグ&ドロップ。
3. 生成された out.csv をダブルクリックして確認。

```
open(OUT,">out.csv");  
select(OUT);
```

```
while(<>){  
    if( /^¥d¥)(.+)$/ ){ ...
```

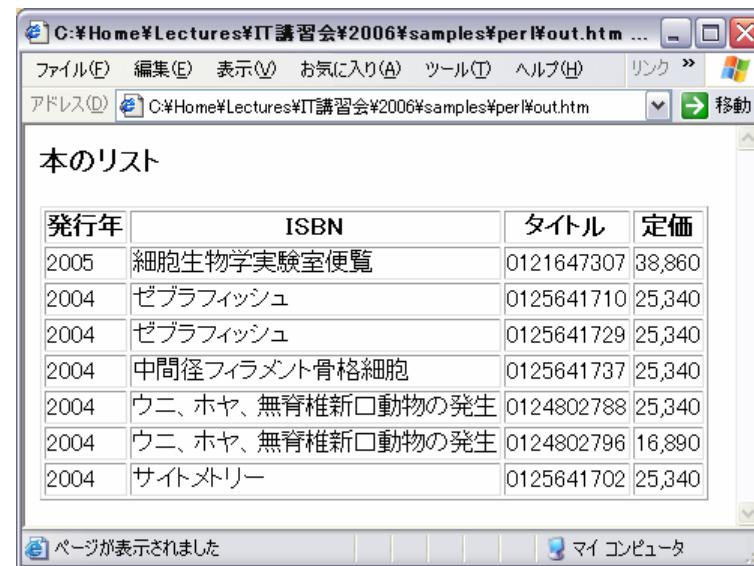
出力先をファイル out.csv に変更
(">>out.csv"とすると追加)

演習 : HTMLファイル生成 (droplet2.pl)

1. droplet2.plをasperlにドラッグ&ドロップして(書いたつもり), 中身を確認.
2. books.txt を droplet2.pl にドラッグ&ドロップ.
3. 生成された **out.htm** をダブルクリックして確認.

```
open(OUT,">out.htm");
select(OUT);

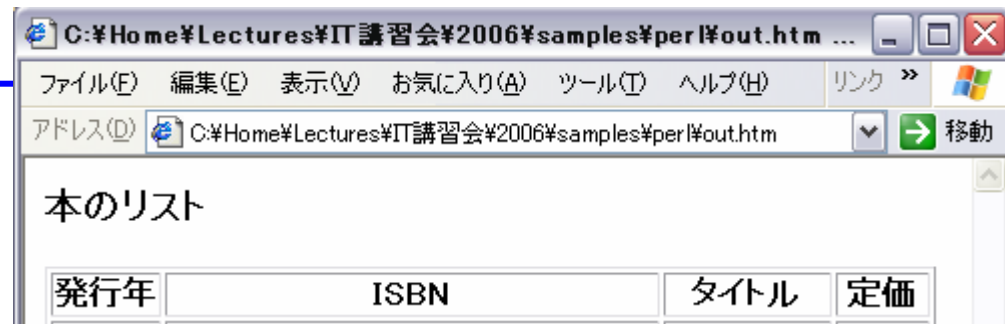
print "<html>¥n";
print "<head>¥n"; ...
```



HTMLファイル生成 (droplet2.pl)

```
open(OUT, ">out.htm");  
select(OUT);
```

```
print "<html>¥n";  
print "<head>¥n";  
print "</head>¥n";  
print "<body>¥n";  
print "<h3>本のリスト</h3>¥n";  
print "<table border=1>¥n";  
print "<tr><th>発行年</th><th>ISBN</th><th>タイトル</th><th>定価</th></tr>¥n";
```



HTMLファイル生成 (droplet2.pl)

```
while(<>){
  if( /^¥d¥)(.+)$/ ){ $title = $1; }
  elsif( /^発行年月: ¥S+-(¥d+).*ISBN: (¥d+)$/ ){
    $year = $1; $isbn = $2;    }
  elsif( /^国内定価 ¥¥([¥d¥,]+)/ ){
    print "<tr><td>$year</td><td>$title</td>
          <td>$isbn</td><td>$1</td></tr>¥n"; } }
print "</table>¥n";
print "</body>¥n";
print "</html>¥n";
```

2005	細胞生物学実験室便覧	0121647307	38,860
2004	ゼブラフィッシュ	0125641710	25,340
2004	ゼブラフィッシュ	0125641729	25,340
2004	中間径フィラメント骨格細胞	0125641737	25,340
2004	ウニ、ホヤ、無脊椎新口動物の発生	0124802788	25,340
2004	ウニ、ホヤ、無脊椎新口動物の発生	0124802796	16,890
2004	サイトメトリー	0125641702	25,340

ページが表示されました

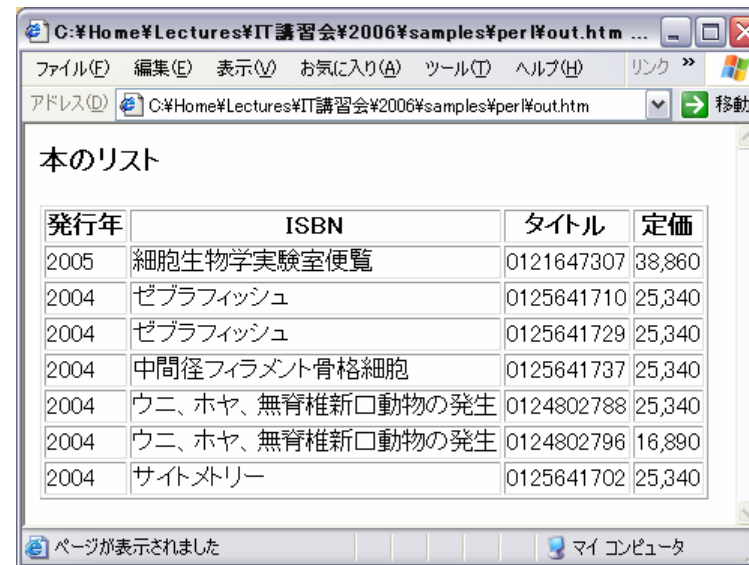
マイ コンピュータ

演習 : HTMLファイル生成 (droplet3.pl)

1. out.htm を削除
2. droplet3.plをasperlにドラッグ&ドロップして(書いたつもり), 中身を確認.
3. books.txt を droplet3.pl にドラッグ&ドロップ.
4. 生成された out.htm をダブルクリックして確認.

```
open(OUT,">out.htm");
select(OUT);

print <<"EOM";
<html> ...
```



ヒアドキュメント (droplet3.pl)

```
print "<html>¥n";
print "<head>¥n";
print "</head>¥n";
print "<body>¥n";
print "<h3>本のリスト
</h3>¥n";
print "<table border=1>¥n";
print "<tr><th>発行年
</th><th>ISBN</th><th>タイ
トル</th><th>定価
</th></tr>¥n";
```

```
print <<"EOM";
<html>
<head>
</head>
<body>
<h3>本のリスト</h3>
<table border=1>
<tr><th>発行年
</th><th>ISBN</th><th>タイ
トル</th><th>定価</th></tr>
EOM
```

電子メールの処理

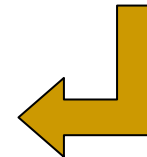
X-UIDL: 1090456471.20412.sail
X-Mozilla-Status: 0001
X-Mozilla-Status2: 00000000
Return-Path: <naka@is.kyusan-u.ac.jp>
Delivered-To: naka@ip.kyusan-u.ac.jp
Received: (qmail 20401 invoked from network); 22 Jul 2004 00:34:31 -0000
Received: from unknown (HELO nsp-in.kyusan-u.ac.jp) (133.17.5.20)
by sail.ip.kyusan-u.ac.jp with SMTP; 22 Jul 2004 00:34:31 -0000
Received: from mvg.kyusan-u.ac.jp by nsp-in.kyusan-u.ac.jp via smtpd (for sail.ip.kyusan-u.ac.jp [133.17.100.3]) with SMTP; 22 Jul 2004 00:34:31 UT
Received: from smtp.is.kyusan-u.ac.jp(133.17.169.25) by mvg.kyusan-u.ac.jp via csmmap id 24130; Thu, 22 Jul 2004 09:45:39 +0900 (JST)
Received: from smtp.is.kyusan-u.ac.jp by nsp-dmz.kyusan-u.ac.jp via smtpd (for mvg.kyusan-u.ac.jp [133.17.5.20]) with SMTP; 22 Jul 2004 00:34:31 UT

Received: from [133.17.165.236] (unknown [133.17.165.236]) by smtp.is.kyusan-u.ac.jp (Postfix) with ESMTP id E4982376A39 for <naka@ip.kyusan-u.ac.jp>; Thu, 22 Jul 2004 09:34:30 +0900 (JST)
Message-ID: <40FF0B95.9080402@is.kyusan-u.ac.jp>
Date: Thu, 22 Jul 2004 09:34:29 +0900
From: Takashi NAKA <naka@is.kyusan-u.ac.jp>
User-Agent: Mozilla Thunderbird 0.7.2 (Windows/20040707)
X-Accept-Language: ja, en-us, en
MIME-Version: 1.0
To: naka@ip.kyusan-u.ac.jp
Subject: The first mail
Content-Type: text/plain; charset=ISO-2022-JP
Content-Transfer-Encoding: 7bit

こんにちは
はじめてのメールです。

× 3

Mails.txt



電子メールの中身(抜粋)

Date: Thu, 22 Jul 2004 09:34:29 +0900

From: naka@is.kyusan-u.ac.jp

To: naka@ip.kyusan-u.ac.jp

Subject: The first mail

こんにちは
はじめてのメールです。

× 3

パターンマッチによる行の抽出

```
while(<>){  
  if( /^Subject:/ ){  
    print;  
  }  
  elsif( /^From:/ ){  
    print;  
  }  
  elsif( /^Date:/ ){  
    print;  
  }  
}
```

```
...  
Date: Thu, 22 Jul 2004 09:34:29 +0900  
From: naka@is.kyusan-u.ac.jp  
To: naka@ip.kyusan-u.ac.jp  
Subject: The first mail
```


```
...  
こんにちは  
はじめてのメールです .
```

```
...  
Date: Thu, 22 Jul 2004 09:35:05 +0900  
From: naka@is.kyusan-u.ac.jp  
To: naka@ip.kyusan-u.ac.jp  
Subject: The seocnd mail
```

```
...  
こんにちは  
2通目のメールです .
```

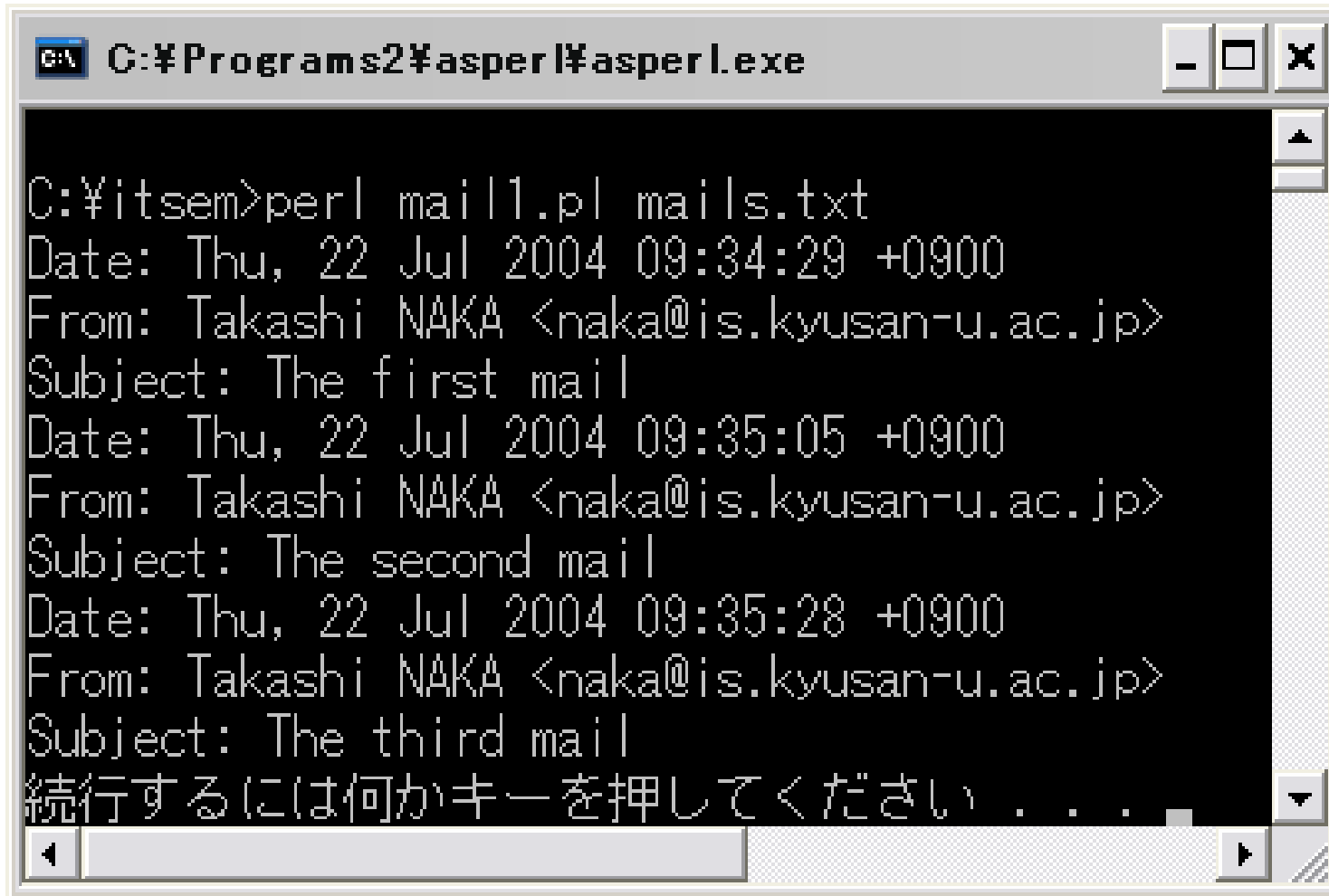
```
...  
...
```

演習：パターンマッチによる行の抽出

1. asperlを起動
2. 右のプログラムを入力
3. ファイル名“mail1.pl”でマインドキュメントに保存
4. [コマンド]の[引数を指定して実行]または  ...
5. データファイル名“Mails.txt”を指定して実行

```
while(<>){
  if( /^Subject:/ ){
    print;
  }
  elsif( /^From:/ ){
    print;
  }
  elsif( /^Date:/ ){
    print;
  }
}
```

演習：抽出された行



```
C:\Programs2\asper\asper.l.exe

C:\itsem>perl mail1.pl mails.txt
Date: Thu, 22 Jul 2004 09:34:29 +0900
From: Takashi NAKA <naka@is.kyusan-u.ac.jp>
Subject: The first mail
Date: Thu, 22 Jul 2004 09:35:05 +0900
From: Takashi NAKA <naka@is.kyusan-u.ac.jp>
Subject: The second mail
Date: Thu, 22 Jul 2004 09:35:28 +0900
From: Takashi NAKA <naka@is.kyusan-u.ac.jp>
Subject: The third mail
続行するには何かキーを押してください . . .
```

パターンマッチによる語の抽出

```
while(<>){
  if( /^Subject: (.*)$/ ){
    print "¥".$from."¥",¥".$date."¥",¥".$time."¥",¥".$1."¥";
    print "¥n";
  }
  elsif( /^From:.*<([^@]+)@/ ){
    $from = $1;
  }
  elsif( /^Date: ¥S+, (¥d+ ¥w+ ¥d+) (¥d+:¥d+:¥d+)/ ){
    $date = $1;
    $time = $2;
  }
}
```

Fromの抽出

```
elseif( /^From:. *<([^\@]+)\@/ ){  
    $from = $1;  
}
```

Date: Thu, 22 Jul 2004 09:34:29 +0900

From: Takashi NAKA <naka@is.kyusan-
u.ac.jp>

Subject: The first mail

Dateの抽出

```
elsif( /^Date: ¥S+, (¥d+ ¥w+ ¥d+)  
(¥d+:¥d+:¥d+)/ ){  
    $date = $1;  
    $time = $2;  
}
```

Date: Thu, 22 Jul 2004 09:34:29 +0900

Subject: The first mail

Subjectの抽出と出力


```
if( /^Subject: (.*)$/ ){  
    print "¥" ".$from." "¥", ¥" ".$date."  
        "¥", ¥" ".$time." "¥", ¥" ".$1." "¥";  
    print "¥n";  
}
```

Date: Thu, 22 Jul 2004 09:34:29 +0900

Subject: The first mail

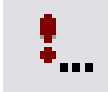
演習：パターンマッチによる語の抽出

1. asperlを起動
2. “Mails.txt”をドラッグ&ドロップ. 中身を確認.
3. “mail2.pl”をドラッグ&ドロップ(書いたつもり)
4. [コマンド]の[引数を指定して実行]または
5. データファイル名”Mails.txt”を指定して実行



```
C:\Programs2\asperl\asperl.exe
C:\itsem>perl mail2.pl mails.txt
"naka", "22 Jul 2004", "09:34:29", "The first mail"
"naka", "22 Jul 2004", "09:35:05", "The second mail"
"naka", "22 Jul 2004", "09:35:28", "The third mail"
続行するには何かキーを押してください . . .
```

演習：抽出結果のファイルへの保存

1. asperlを起動
2. “mail2.pl”をドラッグ&ドロップ
3. [コマンド]の[引数を指定して実行]または 
4. ”Mails.txt>out2.csv”を指定して実行
5. 生成されたout2.csvをダブルクリックしてExcelで確認

```
perl mail2.pl Mails.txt > out2.csv
```

Mails.txt を mail2.pl で処理し、
結果をout2.csv に出力せよ

演習：ドラッグ & ドロップでの実行

1. asperlを起動
2. asperl へ “mail2.pl”をドラッグ & ドロップ
3. 下記を先頭に追加
open(OUT,">mails3.csv");
select(OUT);
4. 保存
5. Mail.txtをmail2.plへドラッグ & ドロップ .
6. 生成されたout3.csvをダブルクリックしてExcelで確認 .

```
open(OUT,">out3.csv");  
select(OUT);  
while(<>){  
    if( /^Subject: (.*)$/ ){  
        print "¥"."$from."¥"    }  
}
```

オンラインアンケート

Webサーバ関連フリーウェアのインストール

オンラインアンケートの仕組み

オンラインテストの紹介

オンラインアンケートの仕組み

WWWサーバ

enquete01.htm

```
<html>
<head>
<META HTTP-EQUIV="..."
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか? <br>
...
```

ウェブブラウザ

The screenshot shows a web browser window with the title "Samba#naka#online test...". The address bar contains "Samba#naka#online". The main content area displays a survey titled "アンケート" (Survey) with the following questions and options:

- 1. 参加してよかったですか?
 はい いいえ
- 2. よかった内容は?
 VBA Perl その他
-- その他

At the bottom of the form, there are two buttons: "送信" (Submit) and "消去" (Clear). The browser's status bar at the bottom indicates "イントラネット" (Intranet).

save01.pl

data.txt

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```

Webサーバプログラムのセットアップ

- インストール対象
 - AN HTTPD:Webサーバ
 - cgi-bin.pl, (Jcode.pl) : CGI用perlライブラリプログラム
 - Active Perl: perl処理系 (インストール済)
 - (参考)wsnedmail: メールサーバ代替 (メール送信が必要な場合)
- インストール手順
 - ダウンロード
 - 解凍とインストール
 - ウェブサーバの設定
 - 動作確認

演習：ダウンロード

- ウェブサーバ: [AN HTTPD](#) / localをクリック
 - ダウンロードの [httpd142p.zip](#) (1,099,264 bytes) をクリック
 - マイドキュメント/Archivesに保存
- ライブラリ: [Cgi-lib.pl](#) / localをクリック
 - [Version 2.18](#) を右クリック 対象をファイルに保存
 - マイドキュメント/Archivesに保存 ウェブブラウザ終了
- (参考) Windows用簡易sendmail: [wsendmail](#) / localをクリック
 - [wsendmail.txt](#) を右クリック 対象をファイルに保存
 - マイドキュメント/Archivesに保存
- (参考) ライブラリ: [Jcode.pl](#) / localをクリック
 - [jcode.pl-2.13](#) を右クリック 対象をファイルに保存
 - マイドキュメント/Archivesに保存

演習：解凍とインストール

- フォルダの準備
 - インストール用フォルダ `www` を `C:¥` に作成
 - `www` に公開用フォルダ `public_html` を作成
- 解凍 & インストール
 - `httpd142n.zip` をダブルクリックして解凍
 - 解凍された全てのファイルとフォルダを `C:¥www` に移動
 - `cgi-lib.pl.txt` を `cgi-lib.pl` に名前変更
 - `C:¥www¥cgi-bin` へ移動
 - (参考) `jcode.pl-2.13` を `jcode.pl` に名前変更
 - `C:¥www¥cgi-bin` へ移動

(参考) 演習 : wsendmailのインストール

- `wsendmail.txt` をダブルクリックして内容を表示
- 以下のように内容を修正
 - `$smtp = "SMTPメールサーバ.com";`
`$smtp = "smtp.is.kyusan-u.ac.jp";`
 - `$from = 'メールアドレス@SMTPメールサーバ.com';`
`$from = 'xxxx@ip.kyusan-u.ac.jp';`
- 名前を `sendmail.cgi` に変更
- `C:¥www¥cgi-bin` へ移動

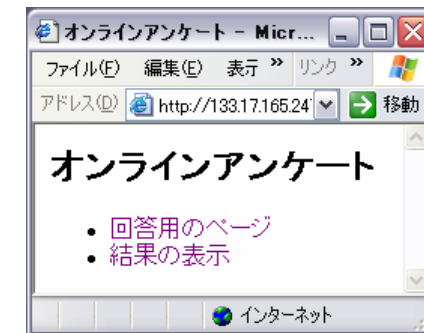
演習：ウェブサーバの設定と動作確認

■ ウェブサーバの設定

- `httpd.exe` をダブルクリックしてサーバを起動し以下を確認, 異なっていたら修正・設定.
 - オプション/一般/ドキュメントルート `C:¥www¥public_html`
 - オプション/エイリアス/仮想パス/cgi-bin `C:¥www¥cgi-bin`

■ 動作確認

- マイドキュメントの `index.htm` を `public_html` に移動
- ウェブブラウザ (IE) を起動
- `localhost/` にアクセス
- 他のマシンへアクセス
 - (<http://マシン名.ip.kyusan-u.ac.jp>)



オンラインアンケート01

WWWサーバ

enquete01.htm

```
<html>
<head>
<META HTTP-EQUIV="...
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか? <br>
...
```

save01.pl

data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```

ウェブブラウザ

アンケート

1. 参加してよかったですか?

はい いいえ

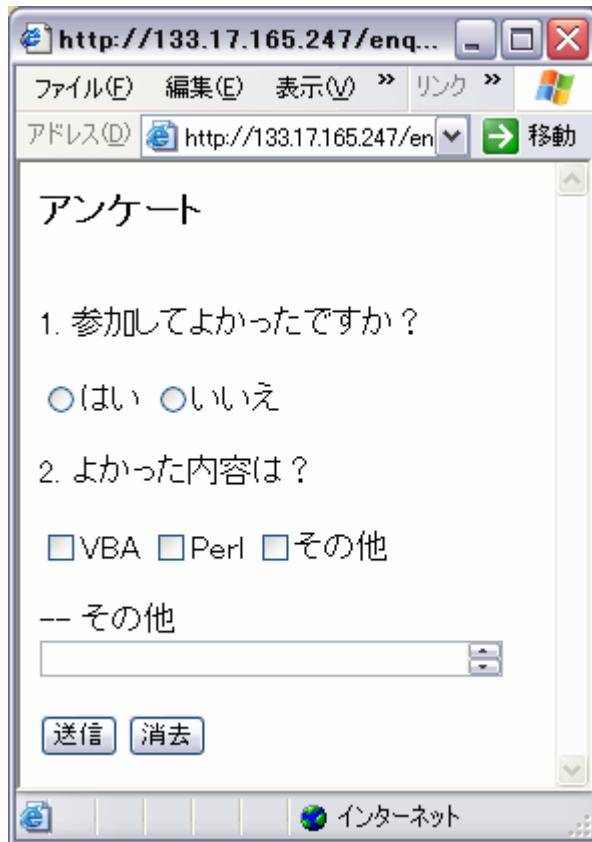
2. よかった内容は?

VBA Perl その他

-- その他

ありがとう。

アンケート表示用HTMLファイル(1)



The screenshot shows a web browser window with the address bar displaying 'http://133.17.165.247/enq...'. The page content is as follows:

アンケート

1. 参加してよかったですか？

はい いいえ

2. よかった内容は？

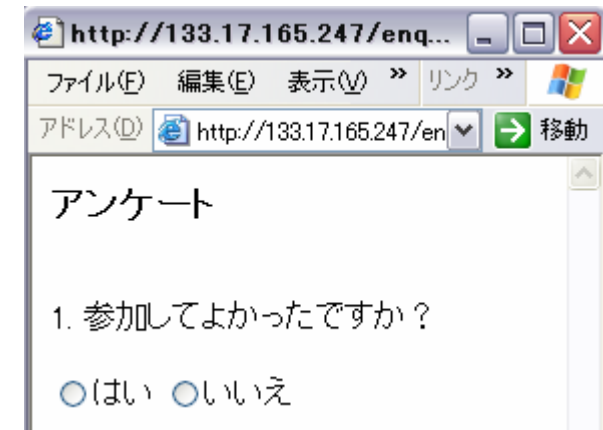
VBA Perl その他

-- その他

```
<html>
<head>
<META http-equiv="Content-Type" content="text/html; charset=Shift_JIS"></head>
<body>
<form method="POST" action=" ../cgi-bin/save01.pl">
<h3>アンケート</h3>
<br>
1. 参加してよかったですか？ <br>
<br>
<input type="RADIO" name="Q1" value="Y">はい
<input type="RADIO" name="Q1" value="N">いいえ
<br>
<br>
2. よかった内容は？ <br>
<br>
<input type="CHECKBOX" name="Q2" value="A">VBA
<input type="CHECKBOX" name="Q2" value="B">Perl
<input type="CHECKBOX" name="Q2" value="C">その他
<br>
<br>
-- その他<br>
<textarea name="Q3" rows="1" cols="30"></textarea><br>
<br>
<input type="SUBMIT" value="送信">
<input type="RESET" value="消去">
</form>
</body>
</html>
```

アンケート表示用HTMLファイル(2)

```
<html>
<head>
<META http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
</head>
<body>
<form method="POST" action="../cgi-bin/save01.pl">
<h3>アンケート</h3>
<br>
1. 参加してよかったですか? <br>
<br>
<input type="RADIO" name="Q1" value="Y">はい
<input type="RADIO" name="Q1" value="N">いいえ
<br>
<br>
```



アンケート表示用HTMLファイル(3)

2. よかった内容は？

<input type="CHECKBOX" name="Q2" value="V">VBA

<input type="CHECKBOX" name="Q2" value="P">Perl

<input type="CHECKBOX" name="Q2" value="O">その他

-- その他

<textarea name="Q3" rows="1" cols="30"></textarea>

<input type="SUBMIT" value="送信">

<input type="RESET" value="消去">

</form>

</body>

</html>

2. よかった内容は？

VBA Perl その他

-- その他

送信 消去

インターネット

データ保存 & お礼表示用Perlスクリプト (1)

```
require 'cgi-lib.pl';

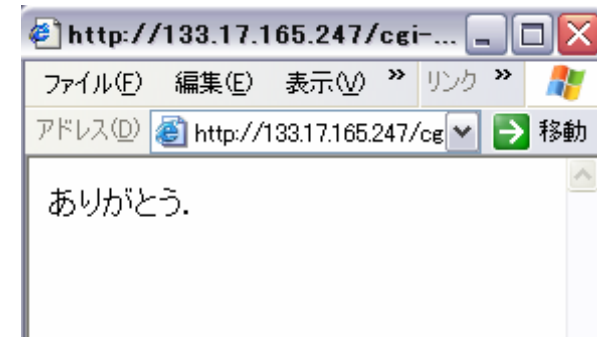
&ReadParse;

$answers = ",.${$Q1}",".${$Q2}",".${$Q3}."."¥n";
$answers =~ s/¥x00//g;
open(OUT, ">>../public_html/data.csv");
print OUT $answers;
close(OUT);

# HTML生成
print <<"EOM";
Content-Type: text/html¥n
<html>
<head>
<META http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
</head>
<body>
ありがとう.
</body>
</html>
EOM
```

data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
. . .
```



データ保存 & お礼表示用Perlスクリプト (2)

data.csv

```
require 'cgi-lib.pl';
```

```
&ReadParse;
```

```
$answers = ", ".$in{"Q1"}.", ".$in{"Q2"}.", ".$in{"Q3"}."¥n";
```

```
$answers =~ s/¥x00//g;
```

```
open(OUT, ">>../public_html/data.csv");
```

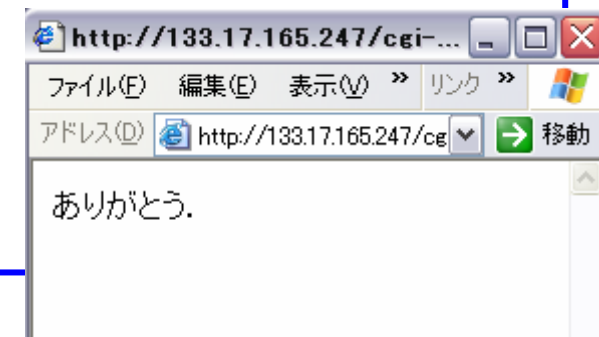
```
print OUT $answers;
```

```
close(OUT);
```

```
Y, VP,  
Y, O, オンラインソフトの利用  
N, ,  
Y, PO, Javascript  
...
```


データ保存 & お礼表示用Perlスクリプト (3)

```
# HTML生成
print <<"EOM";
Content-Type: text/html¥n
<html>
<head>
<META http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
</head>
<body>
ありがとう.
</body>
</html>
EOM
```



演習: アンケート01の作成

- アンケート表示用HTMLファイル
 - マイドキュメントのenquete01.htmをasperlにドラッグ & ドロップして中身を確認 (自分で書いたつもり) .
 - asperlを終了 .
 - enquete01.htmをC:¥www¥public_htmlへ移動 .
- データ保存 & お礼表示用Perlスクリプト
 - マイドキュメントのsave01.plをasperlにドラッグ & ドロップして中身を確認 (自分で書いたつもり) .
 - asperlを終了 .
 - save01.plをC:¥www¥cgi-binへ移動 .

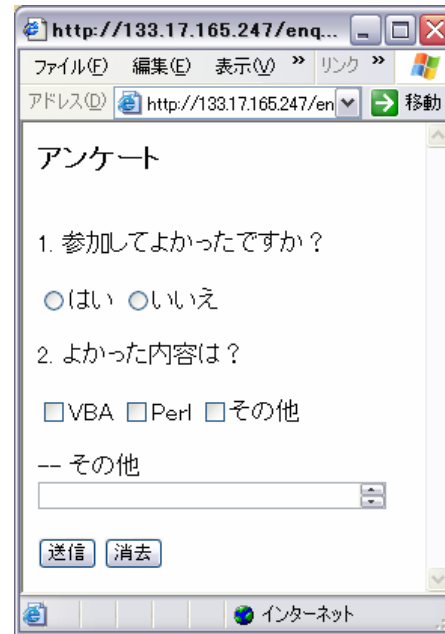
演習: オンラインアンケート01の動作確認

WWWサーバ

http://localhost/enquete01.htm

enquete01.htm

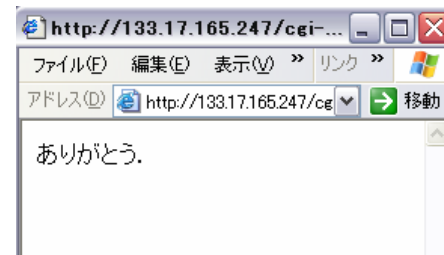
```
<html>
<head>
<META HTTP-EQUIV="...
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか? <br>
...
```



save01.pl

C:\www\public_html\data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N,
Y, PO, Javascript
...
```



オンラインアンケート02

WWWサーバ

ウェブブラウザ

enquete02.htm

```
<html>
<head>
<META HTTP-EQUIV="...
</head>
<body>
<form method="POST"
```

アンケートの内容変更を容易に行えるようにちょっとだけ改良

save02.pl

data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```

アンケート

1. 参加してよかったですか？

はい いいえ

2. よかった内容は？

VBA Perl その他

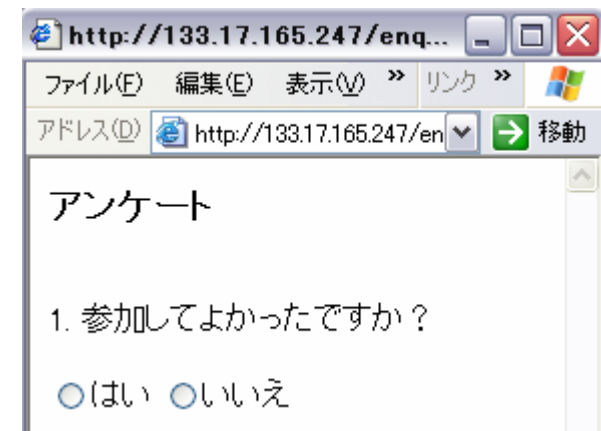
-- その他

送信 消去

ありがとう。

改) アンケート表示用HTMLファイル(1)

```
<html>
<head>
<META http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
</head>
<body>
<form method="POST" action="../cgi-bin/save02.pl">
<h3>アンケート</h3>
<br>
1. 参加してよかったですか? <br>
<br>
<input type="RADIO" name="Q1" value="Y">はい
<input type="RADIO" name="Q1" value="N">いいえ
<br>
<br>
```



改) アンケート表示用HTMLファイル(2)

2. よかった内容は？

<input type="CHECKBOX" name="Q2" value="V">VBA

<input type="CHECKBOX" name="Q2" value="P">Perl

<input type="CHECKBOX" name="Q2" value="O">その他

-- その他

<textarea name="Q3" rows="1" cols="30"></textarea>

<input type="SUBMIT" value="送信">

<input type="RESET" value="消去">

<input type="HIDDEN" name="QN" value=3>

</form>

</body>

</html>

2. よかった内容は？

VBA Perl その他

-- その他

送信 消去

インターネット

改) データ保存 & お礼表示用Perlスクリプト(1)

data.csv

```
require 'cgi-lib.pl';
```

```
&ReadParse;
```

```
$answers = ", ".$in{"Q1"}.", ".$in{"Q2"}.", ".$in{"Q3"}."¥n";
```

```
$answers =~ s/¥x00//g;
```

```
open(OUT, ">>../public_html/data.csv");
```

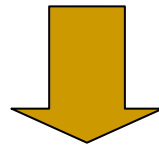
```
print OUT $answers;
```

```
close(OUT);
```

```
Y, VP,  
Y, O, オンラインソフトの利用  
N, ,  
Y, PO, Javascript  
...
```

改) データ保存 & お礼表示用Perlスクリプト(2)

```
$answers = ", ".$in{"Q1"}.", ".$in{"Q2"}.", ".$in{"Q3"}."¥n";
```



```
$qn = $in{"QN"};  
for( $n=0; $n<$qn; ++$n ){  
    $answers = $answers.", ".$in{"Q"}.($n+1)};  
}
```


演習：(改良)アンケート02の作成

- アンケート表示用HTMLファイル
 - マイドキュメントのenquete02.htmをasperlにドラッグ & ドロップして中身を確認(自分で書いたつもり)。
 - asperlを終了。
 - enquete02.htmをC:¥www¥public_htmlへ移動。
- データ保存 & お礼表示用Perlスクリプト
 - マイドキュメントのsave02.plをasperlにドラッグ & ドロップして中身を確認(自分で書いたつもり)。
 - asperlを終了。
 - save02.plをC:¥www¥cgi-binへ移動。

演習：オンラインアンケート02の動作確認

WWWサーバ

http://localhost/enquete02.htm

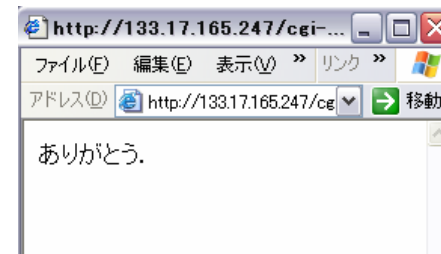
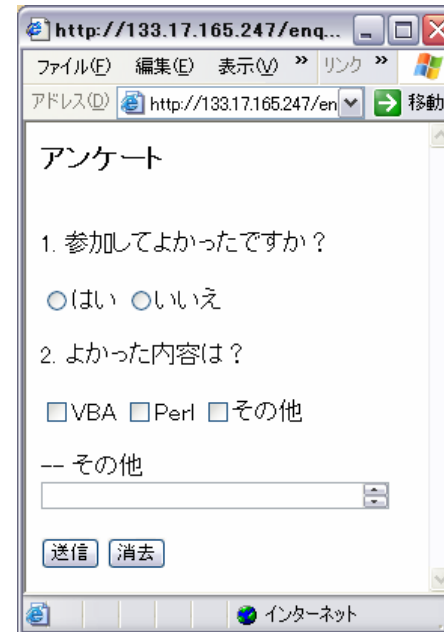
enquete02.htm

```
<html>
<head>
<META HTTP-EQUIV="...
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか？<br>
...
```

save02.pl

C:\www\public_html\data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```



オンラインアンケート02 + 表示01

WWWサーバ

enquete02.htm

```
<html>
<head>
<META HTTP-EQUIV="...
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか? <br>
...
```

ウェブブラウザ

http://133.17.165.247/enq...

アンケート

1. 参加してよかったですか?

はい いいえ

2. よかった内容は?

VBA Perl その他

-- その他

http://133.17.165.247/show...

アンケート結果

設問1	設問2	設問2のその他
Y	VPO	おおお
Y	PO	hhh
2	b	aaaggg
2	b	aaaggg
N	VP	aa

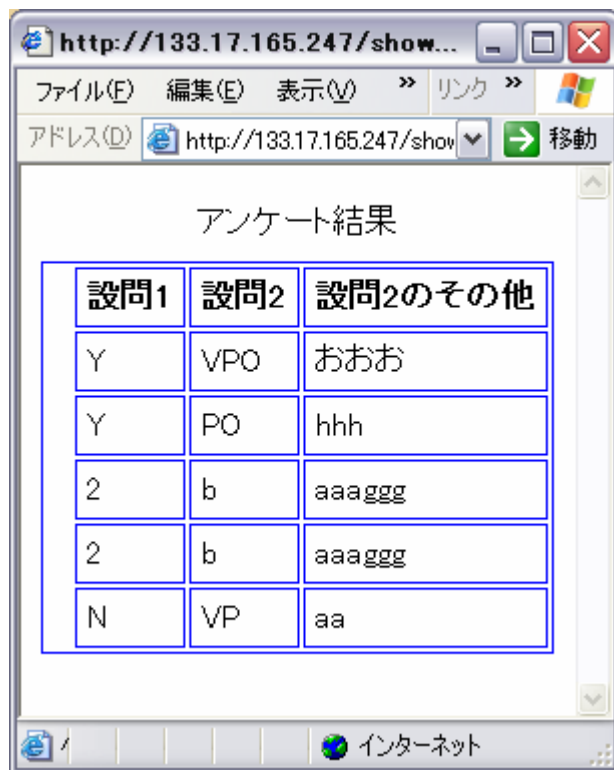
save02.pl

data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```

show01.cgi

データ表示用Perlスクリプト(1)



The screenshot shows a web browser window with the address bar containing 'http://133.17.165.247/show...'. The page title is 'アンケート結果' (Survey Results). The main content is a table with the following data:

設問1	設問2	設問2のその他
Y	VPO	ooo
Y	PO	hhh
2	b	aaaggg
2	b	aaaggg
N	VP	aa

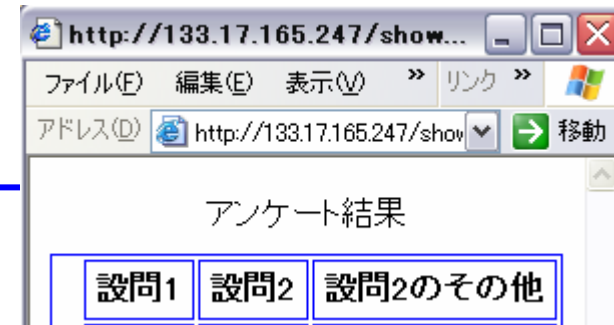
```
print "Content-type: text/html\n\n";
print <<"EOM";
<html>
<head>
<META http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
</head>
<body>
<center>
<table border="1" cellpadding="5" bordercolor="blue">
<caption>アンケート結果</caption>
<tr><th></th><th>設問1</th><th>設問2</th><th>設問2のその他</th></tr>
EOM

open(IN,"data.csv");
while(<IN>){
    chomp;
    print "<tr>\n";
    foreach $a ( split(/,/)) {
        print "<td>.$a.</td>\n";
    }
    print "</tr>\n";
}
close(IN);

print <<"EOM";
</table>
</center>
</body>
</html>
EOM
```

データ表示用Perlスクリプト(2)

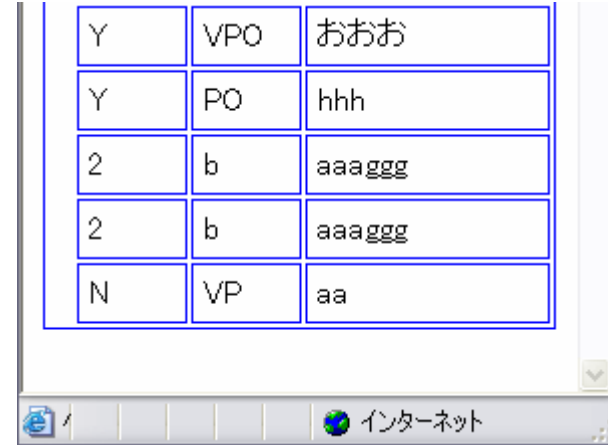
```
print "Content-type: text/html\r\n\r\n";  
print <<"EOM";  
<html>  
<head>  
<META http-equiv="Content-Type" content="text/html; charset=Shift_JIS">  
</head>  
<body>  
<center>  
<table border="1" cellpadding="5" bordercolor="blue">  
<caption>アンケート結果</caption>  
<tr><th></th><th>設問1</th><th>設問2</th><th>設問2のその他</th></tr>  
EOM
```



データ表示用Perlスクリプト(3)

```
open(IN,"data.csv");
while(<IN>){
    chomp;
    print "<tr>¥n";
    foreach $a ( split(/,/)) {
        print "<td>".$a."</td>¥n";
    }
    print "</tr>¥n";
}
close(IN);

print <<"EOM";
</table>
</center>
</body>
</html>
EOM
```



Y	VPO	ooo
Y	PO	hhh
2	b	aaaggg
2	b	aaaggg
N	VP	aa

演習：(改良)アンケート02 + 表示01の作成

- データ表示用Perlスクリプト
 - マイドキュメントのshow01.cgiをasperlにドラッグ & ドロップして中身を確認(自分で書いたつもり)。
 - asperlを終了。
 - show01.cgiをC:¥www¥public_htmlへ移動。

演習：データ表示Perlスクリプトの動作確認

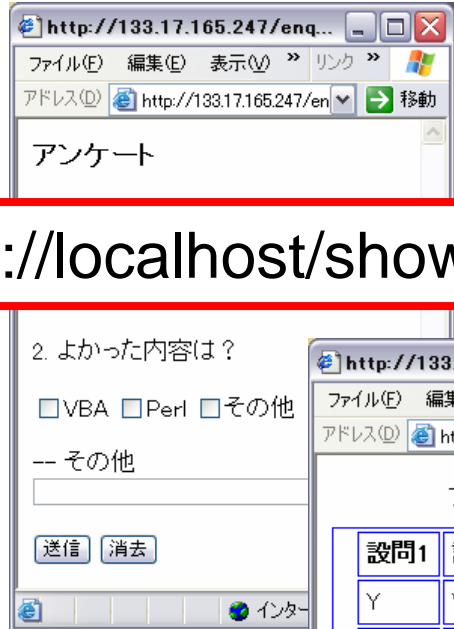
WWWサーバ

ウェブブラウザ

enquete02.htm

```
<html>
<head>
<META HTTP-EQUIV="...
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか？<br>
...
```

http://localhost/show01.cgi

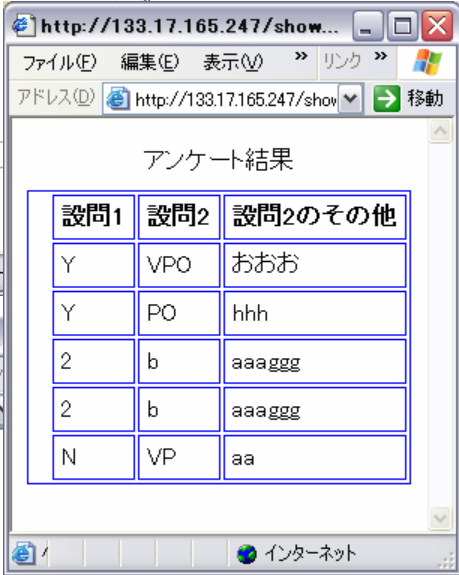


save02.pl

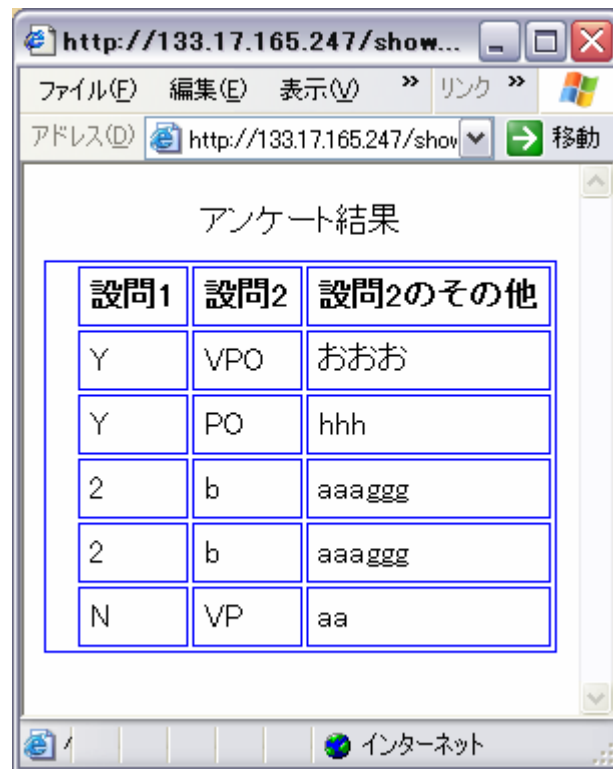
data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```

show01.cgi



データ表示用Perlスクリプトの改良(1)

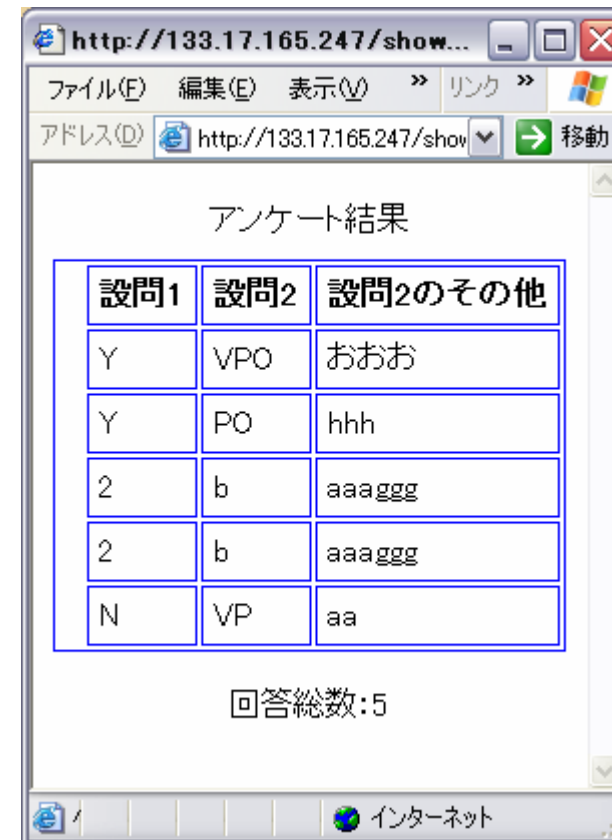
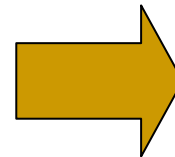


http://133.17.165.247/show...

アンケート結果

設問1	設問2	設問2のその他
Y	VPO	おおお
Y	PO	hhh
2	b	aaaggg
2	b	aaaggg
2	b	aaaggg
N	VP	aa

インターネット



http://133.17.165.247/show...

アンケート結果

設問1	設問2	設問2のその他
Y	VPO	おおお
Y	PO	hhh
2	b	aaaggg
2	b	aaaggg
N	VP	aa

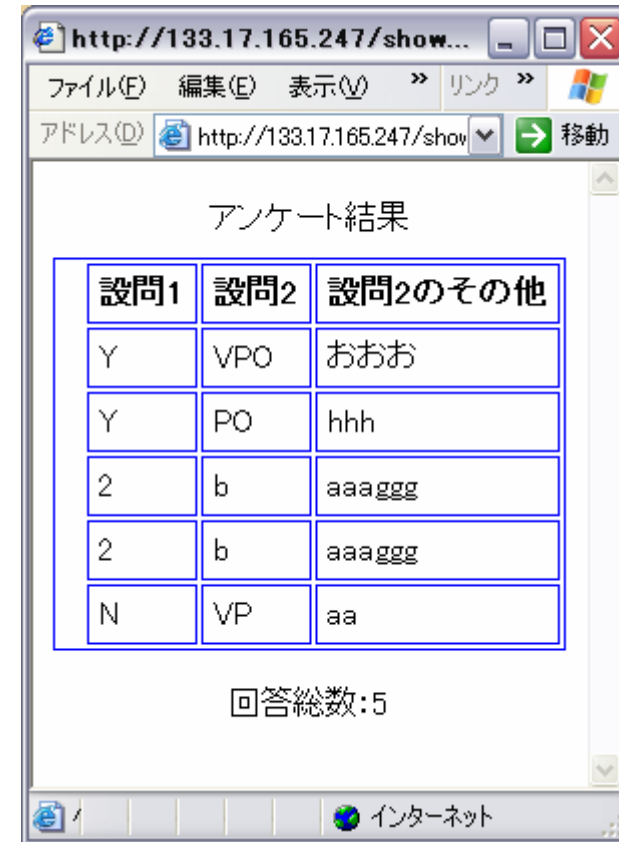
回答総数:5

インターネット

データ表示用Perlスクリプトの改良(2)

```
open(IN,"data.csv");
$an = 0;
while(<IN>){
    chomp;
    ++$an; print "<tr>¥n";
    foreach $a ( split(/,/)) {
        print "<td>".$a."</td>¥n";
    }
    print "</tr>¥n";
}
close(IN);

print <<"EOM";
</table>
<br>回答総数:$an<br>
</center>
```



演習：(改良)アンケート02 + 表示02の作成

- 改良版データ表示用Perlスクリプト
 - マイドキュメントのshow02.cgiをasperlにドラッグ & ドロップして中身を確認(自分で書いたつもり)。
 - asperlを終了。
 - show02.cgiをC:¥www¥public_htmlへ移動。

演習：改良版データ表示Perlスクリプトの動作確認

WWWサーバ

ウェブブラウザ

enquete02.htm

```
<html>
<head>
<META HTTP-EQUIV="...
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか？<br>
...
```

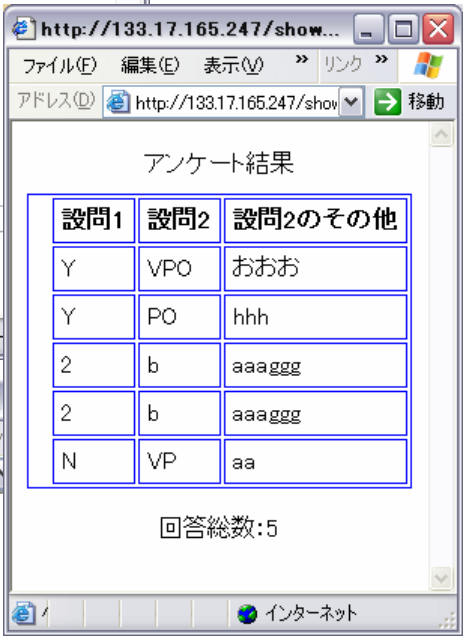
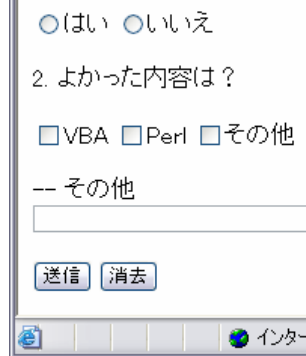
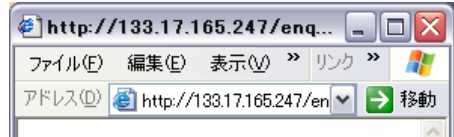
http://localhost/show02.cgi

save02.pl

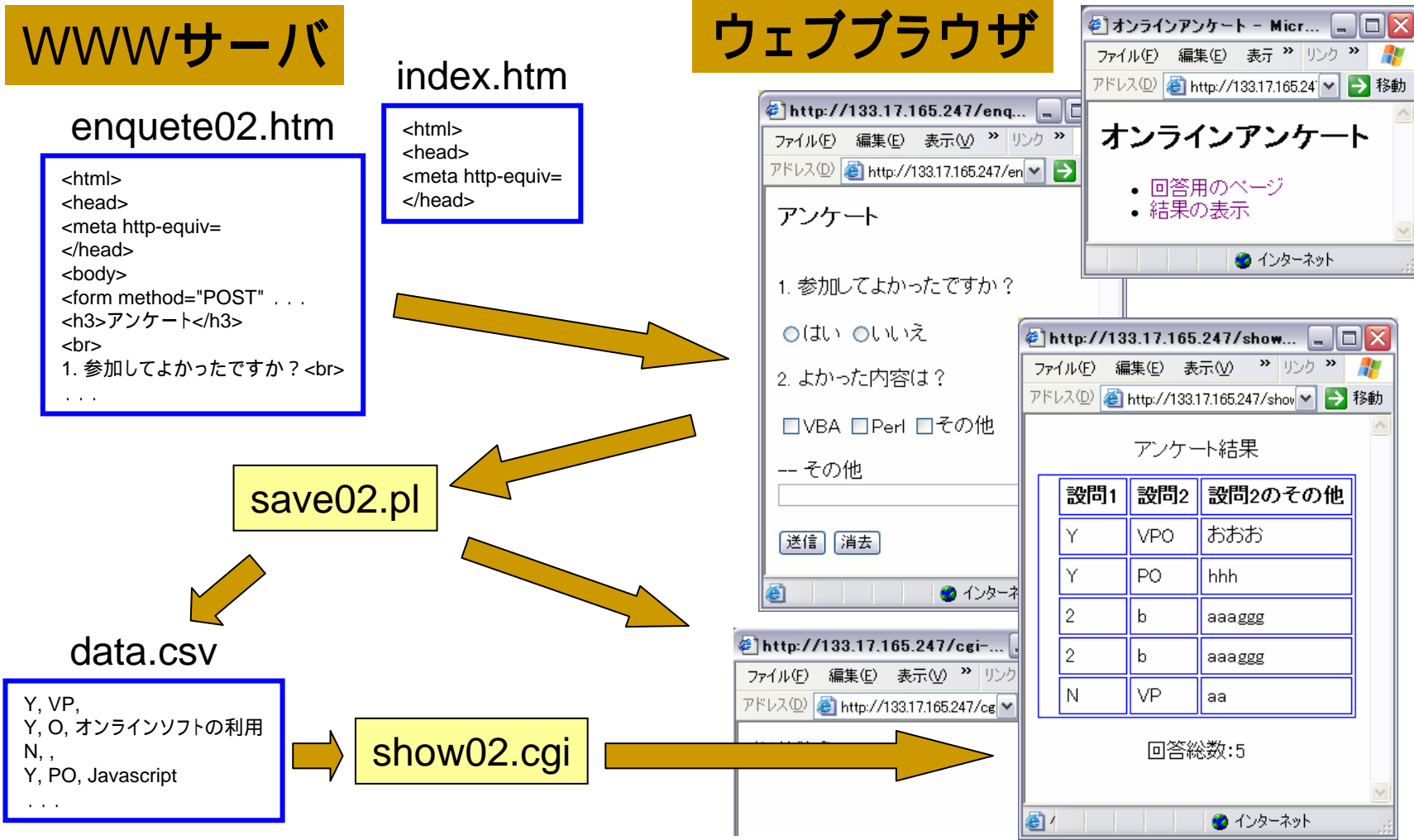
data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```

show02.cgi

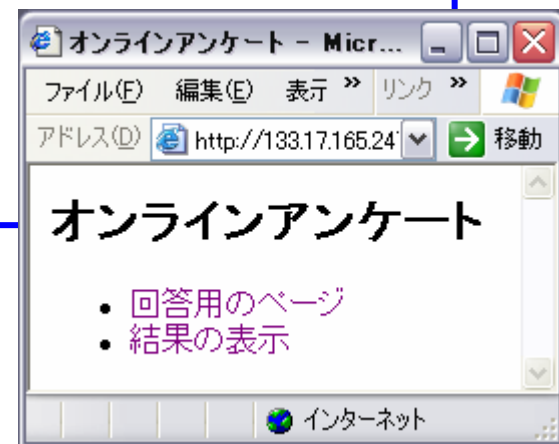


ホームページ (index.htm) の追加



ホームページ (index.htm)

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=shift_jis">
<title>オンラインアンケート</title>
</head>
<body>
<h2>オンラインアンケート</h2>
<ul>
  <li><a href="enquete02.htm">回答用のページ</a>
  <li><a href="show02.cgi">結果の表示</a>
</ul>
</body>
</html>
```



演習：ホームページ (index.htm) の作成

- ホームページ (index.htm) のhtmlファイル
 - マイドキュメントのindex.htmをasperlにドラッグ & ドロップして中身を確認 (自分で書いたつもり) .
 - asperlを終了 .
 - (サーバの動作確認の際に実施済) index.htmをC:¥www¥public_htmlへ移動 .

演習: ホームページ (index.htm) の動作確認

WWWサーバ

enquete02.htm

```
<html>
<head>
<meta http-equiv=
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか? <br>
...
```

index.htm

```
<html>
<head>
<meta http-equiv=
</head>
```

ウェブブラウザ http://localhost/

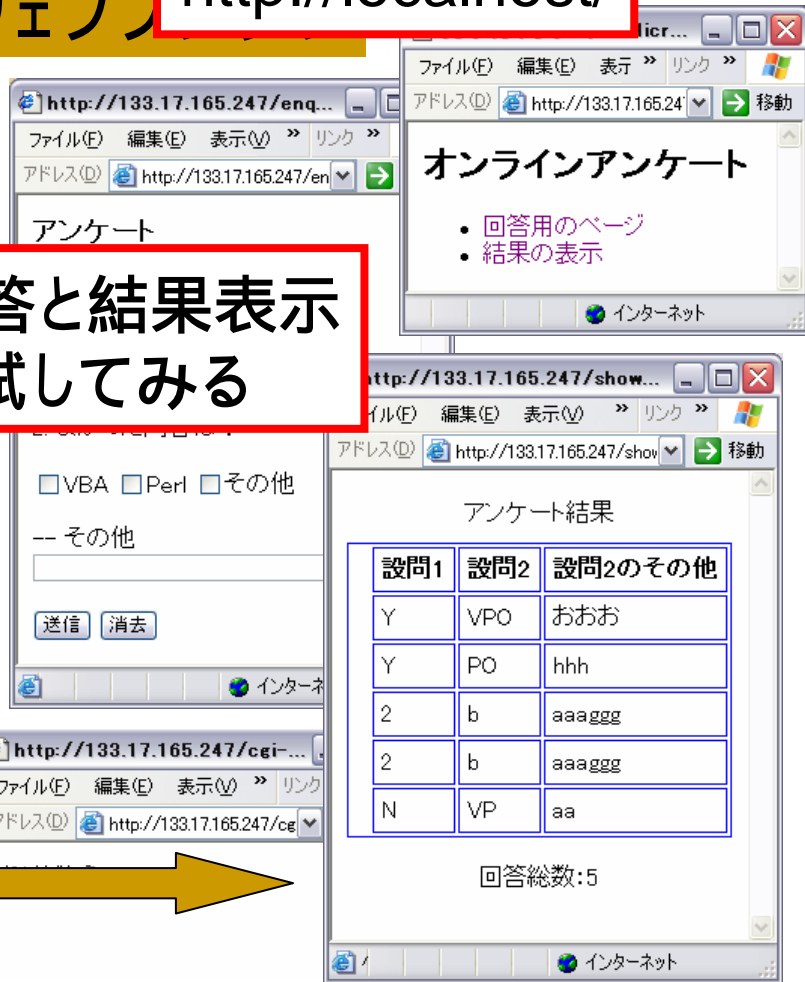
回答と結果表示
を試してみる

save02.pl

data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```

show02.cgi



演習：オリジナルアンケートの作成(1)

- public_htmlのenquete02.htmを修正。
 - 質問項目や回答項目の修正・削除・追加
 - 他の文言の修正
- 動作確認 (<http://localhost/>)
 - 回答してみる
 - データの確認
- 他の講習生のオリジナルアンケート
 - 回答してみる
 - データの確認

アンケートページの生成(1)

ウェブブラウザ

WWWサーバ

index.htm

```
<html>
<head>
<meta http-equiv=
</head>
```

```
<html>
<head>
<meta http-equiv=
</head>
<body>
<form method="POST" ...
<h3>アンケート</h3>
<br>
1. 参加してよかったですか? <br>
...
```

enquete03.txt

```
<h3>アンケート</h3>
1. 参加してよかったですか?
Y:はい, N:いいえ
```

make03.cgi

save02.pl

data.csv

```
Y, VP,
Y, O, オンラインソフトの利用
N, ,
Y, PO, Javascript
...
```

show02.cgi

http://133.17.165.247/enq...

アンケート

1. 参加してよかったですか?

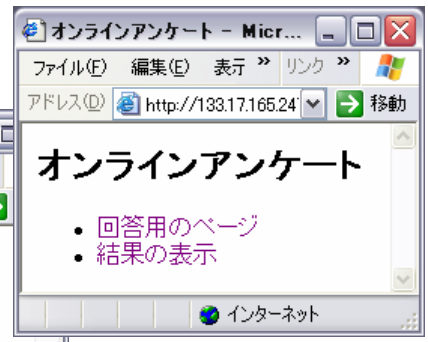
はい いいえ

2. よかった内容は?

VBA Perl その他

-- その他

送信 消去



http://133.17.165.247/show...

アンケート結果

設問1	設問2	設問2のその他
Y	VPO	おおお
Y	PO	hhh
2	b	aaaggg
2	b	aaaggg
N	VP	aa

回答総数:5

アンケートページの生成(2)

enquete03.txt

```
<h3>アンケート</h3>
1. 参加してよかったですか？

    Y:はい, N:いいえ

2. よかった内容は？

    V.VBA, P.Perl, O.その他

-- その他
    1x30
```

make03.cgi

enquete02.htm

```
<html>
<head>
<META http-equiv="Content-Type" content="text/html; charset=Shift_JIS">
</head>
<body>
<form method="POST" action=" ../cgi-bin/save02.pl">
<h3>アンケート</h3>
<br>
1. 参加してよかったですか？ <br>
<br>
<input type="RADIO" name="Q1" value="Y">はい
<input type="RADIO" name="Q1" value="N">いいえ
<br>
<br>
2. よかった内容は？ <br>
<br>
<input type="CHECKBOX" name="Q2" value="V">VBA
<input type="CHECKBOX" name="Q2" value="P">Perl
<input type="CHECKBOX" name="Q2" value="O">その他
<br>
<br>
-- その他<br>
<textarea name="Q3" rows="1" cols="30"></textarea><br>
<br>
<input type="SUBMIT" value="送信">
<input type="RESET" value="消去">
<input type="HIDDEN" name="QN" value=3>
</form>
</body>
</html>
```

アンケートページの生成用Perlスクリプト

```
print "Content-type: text/html\r\n\r\n";
print <<"EOM";
<html>
<head>
<META http-equiv="Content-Type" content="text/html;
charset=Shift_JIS">
</head>
<body>
<form method="POST" action="../cgi-bin/save02.pl">
EOM

open(IN,"enquete03.txt");
while(<IN>){
  chomp;
  # コメント行      : # で始まる
  if( /^#.*$/ ){
  }
  # 問題行          : 半角数字 + ピリオド(選択) ハイフン x 2(自由記述)
  elsif( /^%s*[0-9]+%.|^%s*%-%/ ){
    ++$qn;      print $_.<br>%n";
  }
  # 回答行(選択)   : コロン区切りの繰り返し
  elsif( /^[^:]+:[^:]+,[^:]+:[^:]+*/ ){

    foreach( split(/,/ ) ){
      /%s*([^:]+):%s*(%S+)%s*/;
      print "<input type=%"RADIO%" name=%"Q"."$qn."%"
value=%"."$1."%">". $2."%n";
    }
    print "<br>%n";
  }
}
```

```
# 回答行(複数選択) : ピリオド区切りの繰り返し
elsif( /^[^.]++%.|^%.[^.]++*/ ){
  foreach( split(/,/ ) ){
    /%s*([^.]++)%.%s*(%S+)%s*/;
    print "<input type=%"CHECKBOX%" name=%"Q"."$qn."%"
value=%"."$1."%">". $2."%n";
  }
  print "<br>%n";
}
# 回答行(自由記述) : 数字 x 数字
elsif( /^%s*([0-9]+)x([0-9]+)%s*$/ ){
  print "<textarea name=%"Q"."$qn."%" rows=%"."$1."%"
cols=%"."$2."%"></textarea>%n";
  print "<br>%n";
}
# その他はそのまま出力
else{
  print $_.<br>%n";
}
}
close(IN);
print <<"EOM";
<br>
<input type="SUBMIT" value="送信">
<input type="RESET" value="消去">
<input type="HIDDEN" name="QN" value=$qn>
</form>
</body>
</html>
EOM
```

アンケートページの生成用Perlスクリプト

```
open(IN,"enquete03.txt");
while(<IN>){
  chomp;
  # コメント行      : # で始まる
  if( /^#.*$/ ){ }
  # 問題行          : 半角数字 + ピリオド(選択)ハイフン × 2(自由記述)
  elsif( /^¥s*[0-9]+¥. | ^¥s*¥-¥-/ ){
    ++$qn; print $_."<br>¥n"; }
  # 回答行(選択) : コロン区切りの繰返し
  elsif( /^[^:]+:[^:]+(,[^:]+:[^:]+)* / ){
    foreach( split(/,/ ) ){
      /¥s*([^:]+):¥s*(¥S+)¥s*/;
      print "<input type=¥"RADIO¥" name=¥"Q".
            $qn."¥" value=¥"".$1."¥">".$2."¥n"; }
    print "<br>¥n"; }
}
```

アンケートページの生成用Perlスクリプト

```
# 回答行(複数選択):ピリオド区切りの繰り返し
elsif( /^^[^\.]+\.^[^\.]+(,[^\.]+\.^[^\.]+)*$/ ){
    foreach( split(/,/ ) ){
        /%s*([^\.]+)%.%s*(%S+)%.%s*/;
        print "<input type=%\"CHECKBOX\" name=%\"Q\".
                $qn.\"%\" value=%\"\".$1.\"%\">\".$2.\"%n\";  }

print "<br>%n"; }
# 回答行(自由記述):数字 × 数字
elsif( /^%s*([0-9]+)x([0-9]+)%s*$/ ){
    print "<textarea name=%\"Q\".$qn.\"%\" rows=%\"\".
            $1.\"%\" cols=%\"\".$2.\"%\"></textarea>%n\";

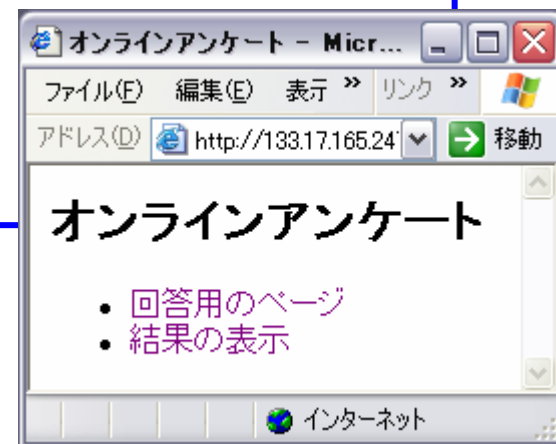
    print "<br>%n"; }
# その他はそのまま出力
else{ print $_.\"<br>%n\"; }
}
close(IN);
```

演習：アンケート生成の作成

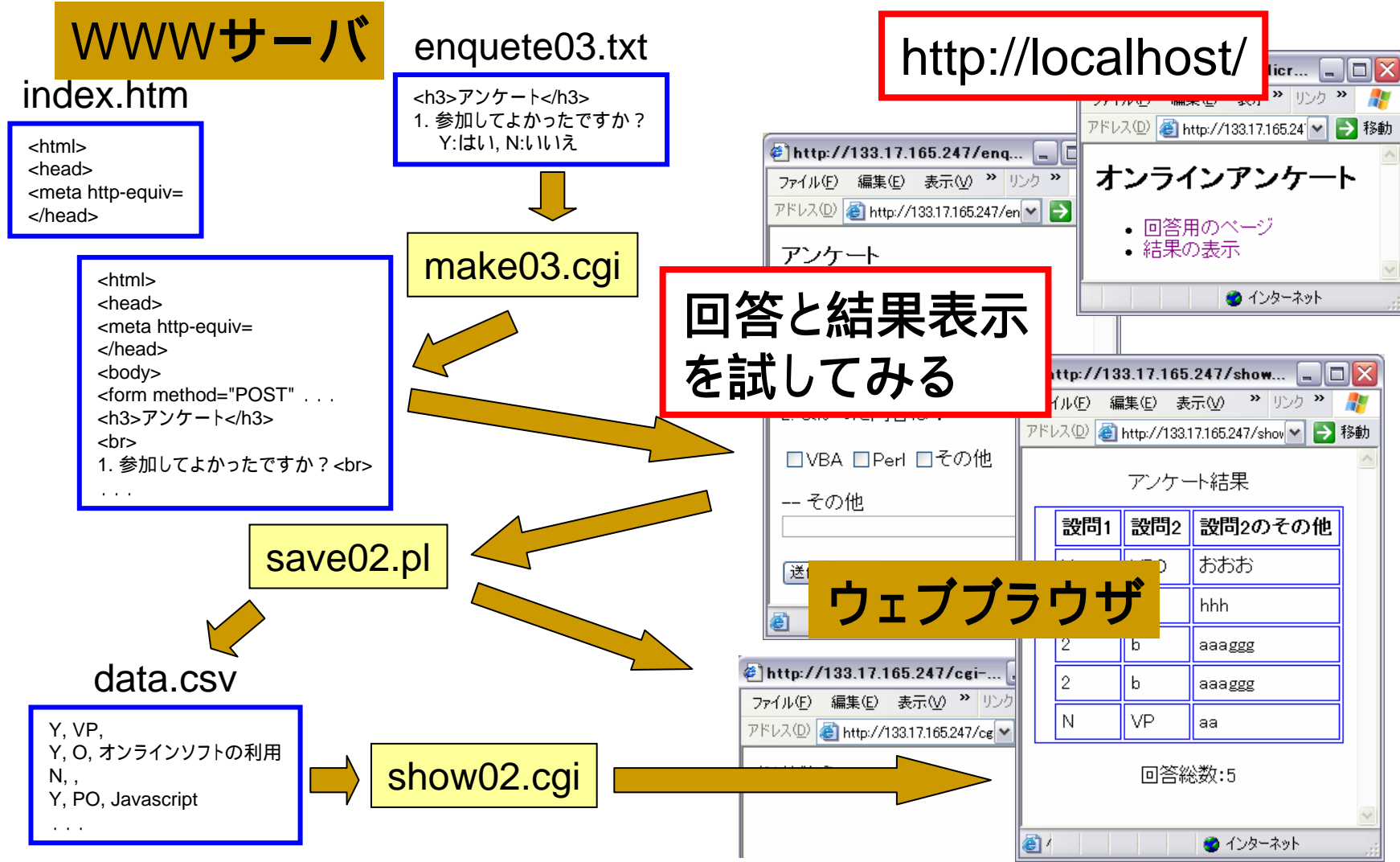
- アンケート生成用perlスクリプト
 - マイドキュメントのmake03.cgiをasperlにドラッグ & ドロップして中身を確認(自分で書いたつもり)。
 - asperlを終了。
 - make03.cgiをC:¥www¥public_htmlへコピー(右ボタンでドラッグ & ドロップ)。
- アンケートのテキストファイル
 - マイドキュメントのenquete03.txtをasperlにドラッグ & ドロップして中身を確認(自分で書いたつもり)。
 - asperlを終了。
 - enquete03.txtをC:¥www¥public_htmlへコピー(右ボタンでドラッグ & ドロップ)。

ホームページの修正 (index.htm)

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=shift_jis">
<title>オンラインアンケート</title>
</head>
<body>
<h2>オンラインアンケート</h2>
<ul>
  <li><a href="make03.cgi">回答用のページ</a>
  <li><a href="show02.cgi">結果の表示</a>
</ul>
</body>
</html>
```



演習: アンケートページの生成の確認



演習：オリジナルアンケートの作成(2)

- public_htmlのenquete03.txtを修正。
 - 質問項目や回答項目の修正・削除・追加
 - 他の文言の修正
- 動作確認 (<http://localhost/>)
 - 回答してみる
 - データの確認
- 他の講習生のオリジナルアンケート
 - 回答してみる
 - データの確認

オンラインテストの紹介

セキュリティの強化

- 認証システムの追加
- 保存の代わりに電子メールを利用

オンラインテストの概要

WWWサーバ

ウェブブラウザ

data.txt

```
[test:1234] naka@kyusan-u.ac.jp
# テスト用
<h3>情報科学部設備等に関するアンケート</h3>
#
# 学籍番号 氏名
#
1. 情報科学部の講義記録システムを知っていますか？
   1:知っている, 2:知らない
2. 講義記録システムを利用したことがありますか？
   a:はい, b:いいえ
3. 利用の目的は？
   a.授業の復習, b.授業の予習, c.他クラス授業参観, d.レポート課題の確認, e.その他
--その他
   1x60
--講義記録システムに対する感想や要望を自由に書いてください。
   5x80
```

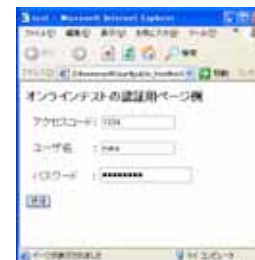
execetest.pl

sendtest.pl

電子メール

Perl & Excel

出題者



	1回	2回	3回	4回	
Aさん	66.7				
Bさん	66.7				
Cさん	33.3				

問題データファイル (testdata.txt)

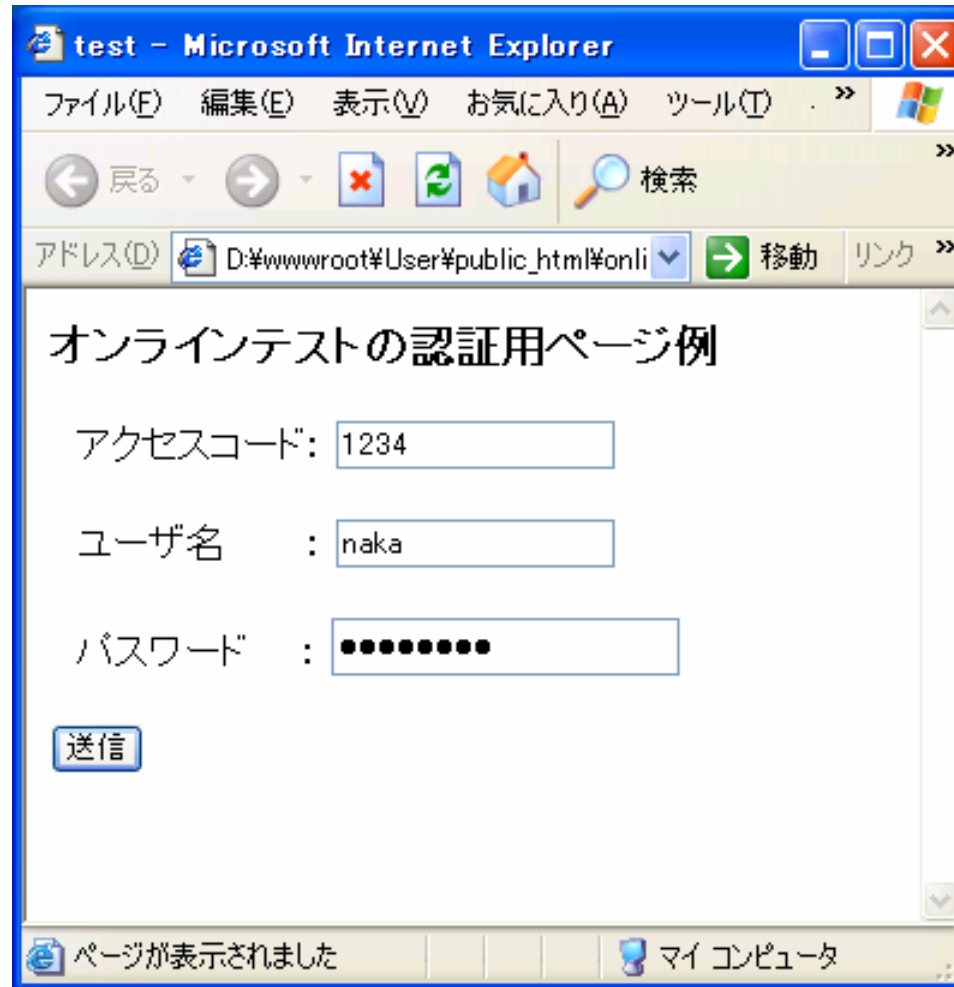
```
[test:1234]naka@kyusan-u.ac.jp
# テスト用
<h3>情報科学部設備等に関するアンケート</h3>
#                               学籍番号       氏名
#
1. 情報科学部の講義記録システムを知っていますか？
   1:知っている, 2:知らない

2. 講義記録システムを利用したことがありますか？
   a:はい, b:いいえ

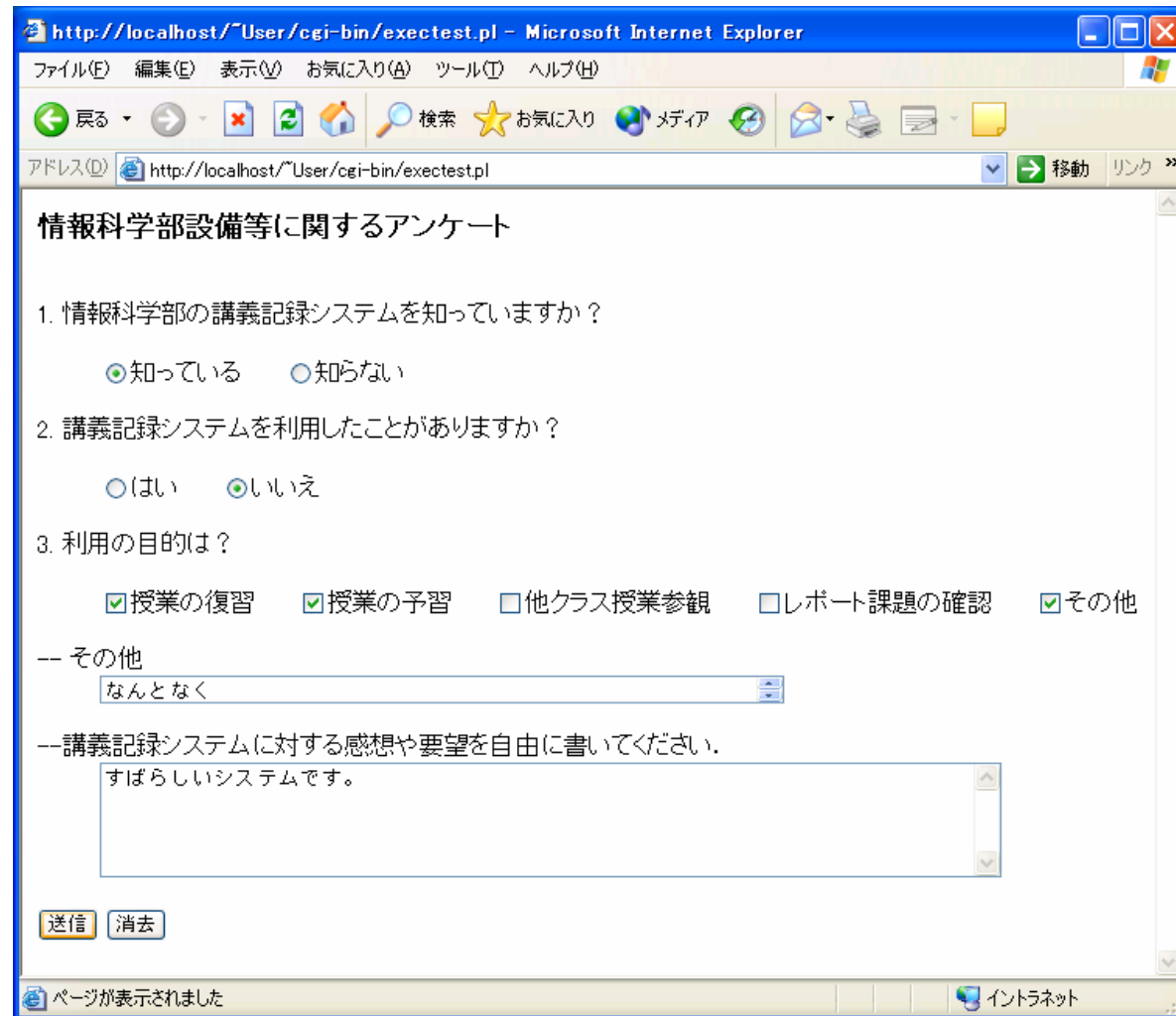
3. 利用の目的は？
   a.授業の復習, b.授業の予習, c.他クラス授業参観, d.レポート課題の確認, e.その他
-- その他
   1x60

--講義記録システムに対する感想や要望を自由に書いてください.
   5x80
```

回答者認証ページ (test.htm)



問題 & 回答ページ (execetest.pl)



The screenshot shows a Microsoft Internet Explorer browser window with the address bar set to `http://localhost/~User/cgi-bin/execetest.pl`. The page content is a survey titled "情報科学部設備等に関するアンケート".

1. 情報科学部の講義記録システムを知っていますか？
 知っている 知らない

2. 講義記録システムを利用したことがありますか？
 はい いいえ

3. 利用の目的は？
 授業の復習 授業の予習 他クラス授業参観 レポート課題の確認 その他

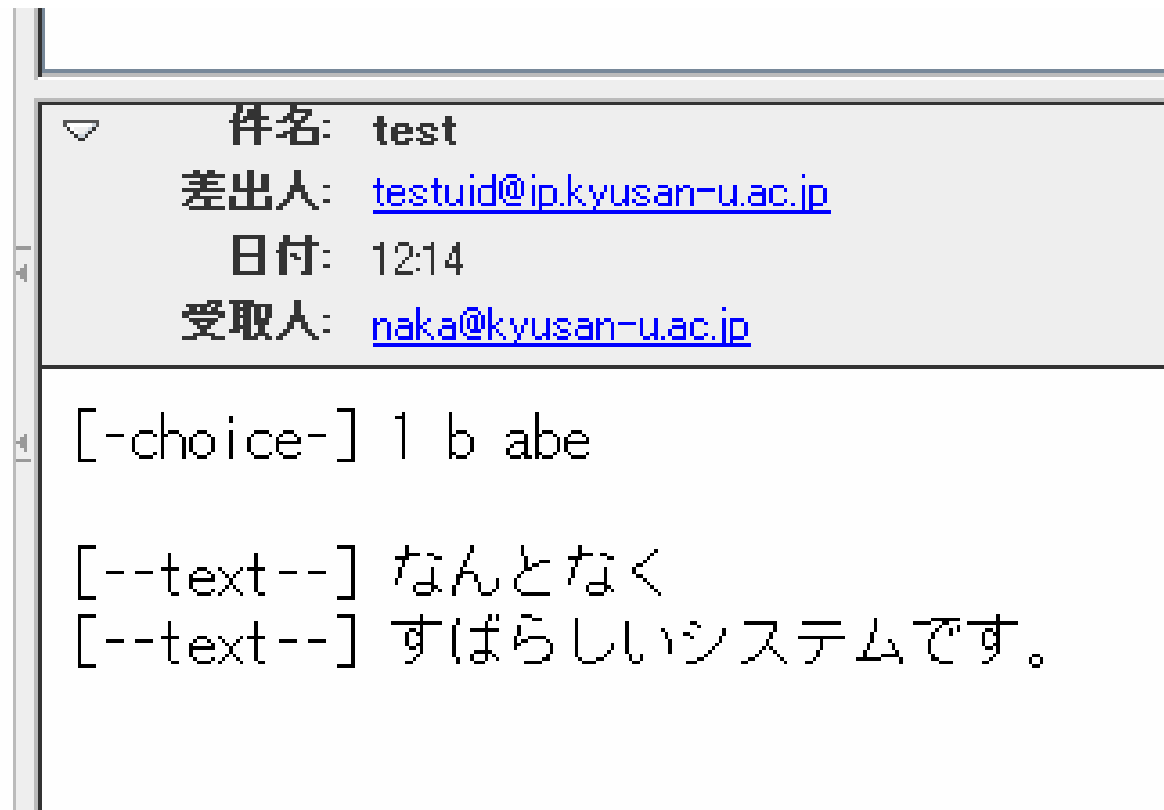
-- その他
なんとなく

-- 講義記録システムに対する感想や要望を自由に書いてください。
すばらしいシステムです。

送信 消去

ページが表示されました イン트라ネット

回答電子メール



メールボックスファイルの内容

...

Received: (qmail 23593 invoked by uid 60001); 23 Jul 2003 03:44:12 -0000

Date: 23 Jul 2003 03:44:12 -0000

Message-ID: <20030723034412.23592.qmail@star2.ip.kyusan-u.ac.jp>

Subject: test

From: testuid@ip.kyusan-u.ac.jp

To: naka@kyusan-u.ac.jp

[-choice-] 1 b abe

[--text--] なんとなく

[--text--] すばらしいシステムです。

...