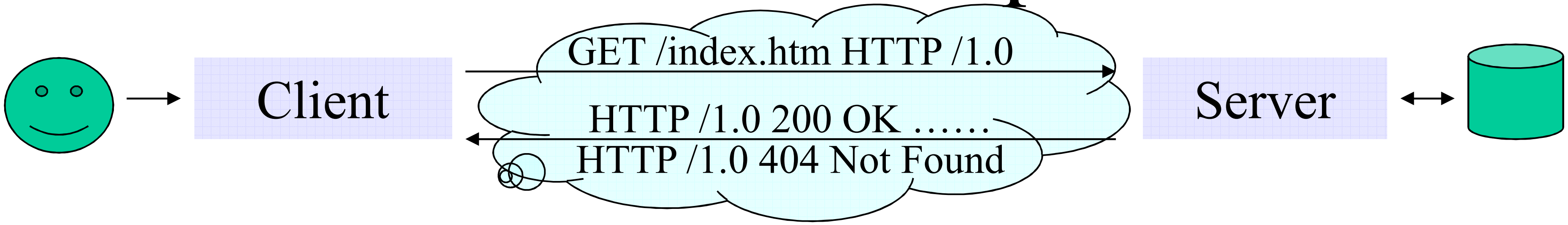
# Perl语言高级编程专题
# Lesson 14

周晓方
courses@xfzhou.homeftp.org

# Web browser & Http Server



1. User opens browser and enters a target address
2. Browser connects to the server (on port 80), and sends request
3. Server finds the web page and returns it with HTTP header.
4. If not found, server returns an error message.
5. Browser transforms the reply and display to user.

- Try these:

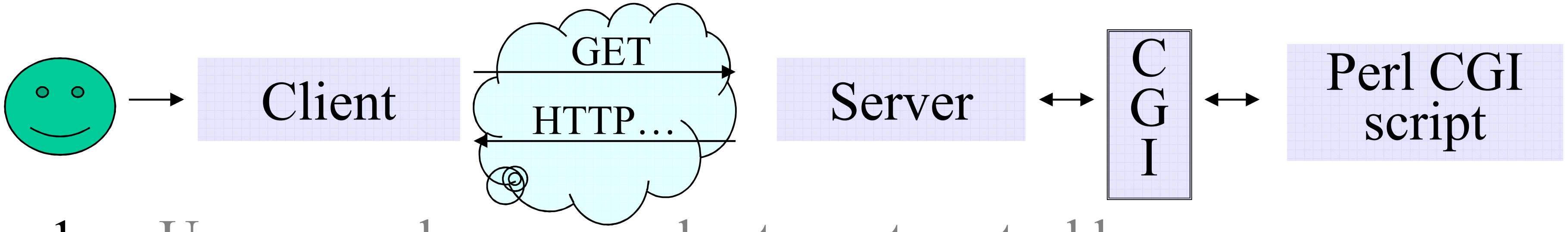
  telnet `www.fudan.edu.cn 80`

  type "`GET / HTTP /1.0`", with 2 ENTER's

  telnet `mail.fudan.edu.cn 25`

  telnet `mail.fudan.edu.cn 110`

# Web Server & CGI Perl



1. User opens browser and enters a target address
2. Browser connects to the server (on port 80), and sends request
3. Server setups CGI environment and starts the Perl CGI script. After script terminats, server collects and returns the result.
4. If Perl CGI script not found, server returns an error message.
5. Browser transforms the reply and display to user.

+ The web page is dynamic, not static.

+ Perl/CGI is almost platform independent (v.s. ASP etc)

+ Client browser is simple (v.s. client side scripts)

- Speed may be slower if server is slow (v.s. ASP).

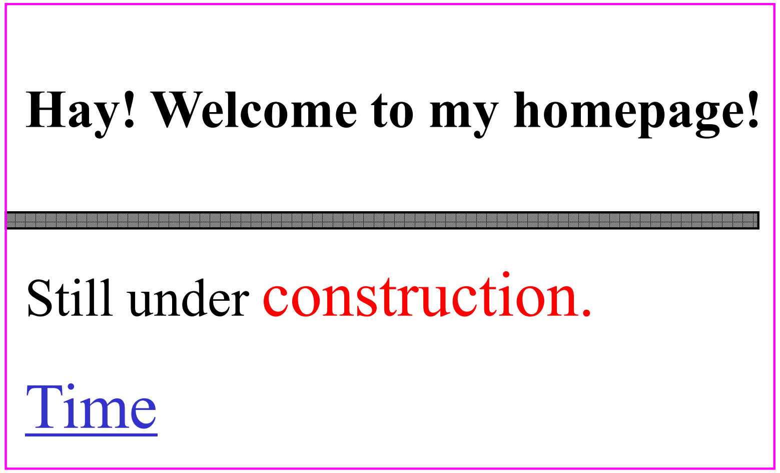
# Install a local Web Server

- Check if tcp/ip protocol is installed on your PC
- 'xitami' free web server (www.xitami.com)
  - xitami/Pro是带SSL的xitami *http://www.xitami.com/download.htm*
  - (2010年)该网站已更新到X5版本，本课件在X2上试用
  - https://en.wikipedia.org/wiki/Xitami
  - https://github.com/imatix-legacy/xitami.com
- Install the GUI version of web server
- Setup http://localhost/admin
- home of webpage: …/xitami/webpages
- home of cgi scripts: …/xitami/cgi-bin
- home of ftp root: …/xitami/ftproot/???/
- test server: http://localhost/
- test cgi: http://localhost/cgi-bin/testcgi

# Write a 1st Web Page

- Add following lines to …/webpages/home.htm
- Open IE & links to http://localhost/home.htm

```
<html>
<head>
<title>My 1st Web Page</title>
</head>
<body>
<h2>Hay! Welcome to my homepage!</h2>
<hr>
Still under
<font color=red>construction</font>.
<br><a href="/cgi-bin/time.pl">Time</a>
</body>
</html>
```



**Hay! Welcome to my homepage!**

Still under construction.

Time

# Write a 1st CGI-Perl script

- Add these lines to …/cgi-bin/time.pl
- Click the "Time" link, and runs the script on server.

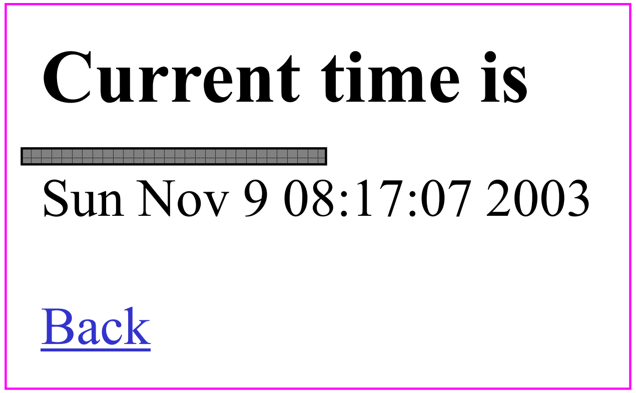
```
#!/usr/bin/perl -w
use strict;

my $time = localtime();

print << "HERE";
Content-type: text/html


<html><head><title>What's the
time?</title>
</head><body><h3>Current time is</h3>
<hr width=50\% align=left>
$time<br><br>
<a href="/home.htm">Back</a>
</body></html>
HERE

1;
```



**Current time is**

Sun Nov 9 08:17:07 2003

Back

# Xitami Server Hints

- 公网网站一定要拿ICP号*http://www.miibeian.gov.cn/*
- 尽量避免开设论坛、博客、新闻等内容
- 解包xiopen_2_5.zip后可直接运行
  - GUI / Console / Server / Control panel版本
- 设置文件
  - 需要简单认证的网页口令和密码：*xitami.aut*
  - 网站和虚拟网站的设置：*default.cfg, *.cfg*
- 配置方法：打开admin口，进入http://127.0.0.1/admin
- 什么是SSI(网页过滤功能)：看一下*perlssi*脚本
- 怎么调试CGI：查看*cgierr.log*
- 在一台主机上建立多个虚拟网站
  - 分目录，绑定不同的ip地址，绑定不同的域名
- 提示：安装路径和perl路径不要包含空格

# HTML formats & tags

- Tags: *<TAG>*………*</TAG>* v.s. *<TAG>*
  - Container tags: <big>……</big>, <h3>…..</h3>
  - Empty tag, open tag, or one-time tag: <p>, <br>, <hr>
  - Tag with arguments: <hr width=50% align=left>
- White spaces ≡ Single space (with exceptions)
- Special symbols ('&' is the "escape-char" in html):
  - &nbsp or &nbsp, an non-breaking-space
  - &lt (<), &gt (>), &amp (& itself), &pi (π), … …
  - &#169 or &copy (©), &reg (®), &#165 (¥)
- 非frame的网页

  <html> <head>…</head> <body>…</body> </html>
- Frame的网页框架（用notepad看一下c:/perl/html/index.html）

  <html> <head>…</head>

  <frameset> … </framset>

  <noframes> … </noframes>

  </html>

# Inside a <head>…</head> section

- <title>…</title>, add a title on browser 不宜过长
- <meta>…</meta>
  - 显示中文<meta http-equiv="Content-Type" content="text/html; charset=gb2312">
- sample:
  - <html><head><title>周晓方复旦大学开课相关网页</title> <meta http-equiv="Content-Type" content="text/html; charset=gb2312"></head> <body bgcolor=#A0FFA0>……
- Good style —— Add **comments** in head section:
  <!- *This is created by Foo. Purpose :… ->*

# Headings and paragraphs

- <h1>…</h1>     <h2>…</h2> ~ <h6>…</h6>
- Paragraphs: <p>A paragraph.</p>
  - The </p> is optional
  - <p> with align=???: <p align=center/left/justify/right>
  - <p> with nowrap: <p nowrap>
- Adding line brakes: <br>
- Preformated paragraphs:      <pre>…</pre>
  - White spaces are significant inside <pre>…</pre>
- horizontal bar: <hr>
  - width=*pixels*     width=*percent*%      size=*pixels*
  - align=*center*     align=*left*      align=*right*
  - color=*#rrggbb*   color=*red/green/lightblue/darkgray*…

# Text formatting

- Bold粗体字<b>…</b>　Italic斜体<i>…</i>
  下划线<u>…</u>　划线<strike>…</strike>
- 大号字体<big>…</big> 小号<small>.</small>
- 字体<font color=?? face=".." size=??>..</font>
- 下标<sub>…</sub>　　　上标<sup>…</sup>
- 逻辑字体
  - 加重<em>…</em> <strong>…</strong>
  - 等宽<code>…</code>, <tt>..</tt>
  - 变量<var>…</var>

# Listing

- ordered list:

```
<ol>
<li>Apply project
<li>Sign contract & budget
<li>Hardworking
<li>Check point
</ol>
```

- unordered list:

```
<ul>
<li>Leaser Jet 6L
<li>Leaser Jet 1010
<ul>
<li>cheeper
<li>faster
</ul>
</ul>
```

1. Apply project
2. Sign contract & budget
3. Hardworking
4. Check point

- Leaser Jet 6L
- Leaser Jet 1010
  - cheeper
  - faster

# Link

- Link to other web pages

```
<a href="home.htm">My home page</a>
<a href="/index.htm">Back to home</a>
<a href="http://www.google.com.hk">Google</a>
<a href="/cgi-bin/time.pl">What's the time now.</a>
```

- Link to named tag

```
<a name="SEC1">Section 1:</a>
……
Goto <a href="#SEC1">section 1</a>
Goto <a href="charp2.html#SEC1">Sec 1 of Ch2"</a>
```

- Open link in new windows

```
<a href="page.htm" target="Name">Here</a>
```

- Invoke your mail client

```
<a href="mailto:courses@xfzhou.homedns.org">Me</a>
```

# Table

- <table>…</table>
- <caption>…</caption>
- row: <tr>…</tr>
- head cell: <th>…</th>
- data cell: <td>…</td>

**Team Statistics**

| **Player** | **Goals** | **Assists** | **Points** |
|---|---|---|---|
| Anne | 7 | 12 | 19 |
| Eric | 4 | 11 | 15 |

```
<TABLE>
  <CAPTION><B>Team Statistics
</B></CAPTION>
  <TR>   <!-- Row 1 -->
    <TH>Player</TH>
    <TH>Goals</TH>
    <TH>Assists</TH>
    <TH>Points</TH>
  </TR>
  <TR>   <!-- Row 2 -->
    <TD>Anne</TD>
    <TD>7</TD>
    <TD>12</TD>
    <TD>19</TD>
  </TR>
  <TR>   <!-- Row 3 -->
    <TD>Eric</TD>
    <TD>4</TD>
    <TD>11</TD>
    <TD>15</TD>
  </TR>
</TABLE>
```

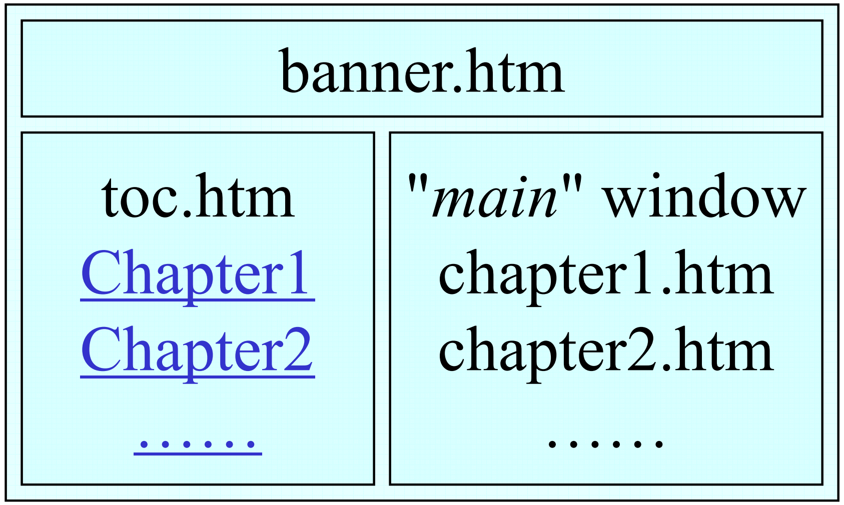
# Image

- Image format: GIF and JPEG
  - GIF: 256 color compressed, LZ-compress
  - JPEG: True color compressed, DCT+Huffman
- <img src="*URL_of_image*">
  - width=*pixel* hight=*pixel* alt="*description text*"
    border=*pixel* vspace=*pixel* hspace=*pixel*
    align=*top/middle/bottom/left/right*
- Image as Bullet Characters

```
<IMG SRC="bullet.gif" WIDTH=12 HEIGHT=12 ALT="*">Item 1<BR>
<IMG SRC="bullet.gif" WIDTH=12 HEIGHT=12 ALT="*">Item 2<BR>
<IMG SRC="bullet.gif" WIDTH=12 HEIGHT=12 ALT="*">Item 3<BR>
```

# Frames

- <frame src="…">
- <frameset cols=…>
- <frameset rows=…>



banner.htm

toc.htm
Chapter1
Chapter2
……

"*main*" window
chapter1.htm
chapter2.htm
……

Inside "*index.html*"

```
<FRAMESET ROWS="80,*">  <!-- Split screen into two rows. -->
    <FRAME SRC="banner.htm">
    <FRAMESET COLS="175,*">  <!-- Split row 2 into two columns. -->
        <FRAME SRC="toc.htm">
        <FRAME SRC="changing_content.html" NAME="main">
    </FRAMESET>
</FRAMESET>
```

Inside "*toc.htm*"

```
<a href="chapter1.htm" target="main">Chapter1</a>
<a href="chapter2.htm" target="main">Chapter2</a>
```

# HTML Forms 发送、接收表单信息

- Client side:
  <form action="*URL_to_script*" method=*POST* | *GET*>
  … … … … </form>
  - action="protocol://server/path/script/extra_path_info"
  - GET(default): append all encoded data at the end of URL
  - POST: sends data to server in a separate HTTP transaction
- Server side (method=GET) CGI:
  - Receive encoded data in $ENV{QUERY_STRING}
  - Use STDOUT to return data
  - **Data can't be too long. Usually < 1K bytes**
- Server side (method=POST) CGI:
  - Receive encoded data from STDIN. Note: no EOF at end!
  - use $ENV{CONTENT_LENGTH} to determine the data size
  - Use STDOUT to return data

# Usefull CGI ENV Variables

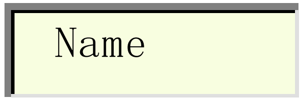
| **Variable** | **Method** | **Description** |
|---|---|---|
| *CONTENT_LENGTH* | POST | length in byte of encoded data |
| PATH_INFO | | extra path information string |
| PATH_TRANSLATED | | physical path/script/extra_path_info |
| *QUERY_STRING* | GET | encoded data |
| REMOTE_ADDR | | IP address of remote computer |
| REMOTE_HOST | | name of the remote computer |
| REMOTE_USER | | authenticated remote user name |
| *REQUEST_METHOD* | | "GET" or "POST" |
| SCRIPT_NAME | | Virtual path of the script |
| SERVER_NAME | | Domain name or IP of the server |
| SERVER_PORT | | Port # of the server (default is 80) |
| SERVER_SOFTWARE | | Name of the server software (Xitami) |

# TEXT, PASSWORD, TEXTAREA

- A single line of text (user-name, phone #, etc)

```
<INPUT [TYPE=TEXT] NAME="id" [SIZE=nn] [MAXLENGTH=nn] [VALUE="default"]>
```

  - SIZE: the windows width show on browser
  - MAXLENGTH can be larger then SIZE (scrolling)

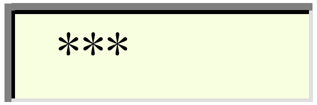


Name

- A single line of password (password)

```
<INPUT [TYPE=PASSWORD] NAME="id" [SIZE=nn] [MAXLENGTH=nn] [VALUE="..."]>
```

  - Display "*" when you typing in PASSWORD line
  - Text is not encrypted when sending to server. Better use with POST method.

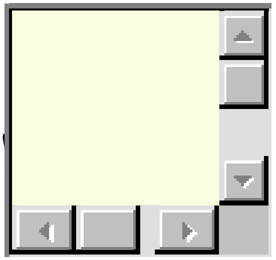


***

- Multiply lines of text (email body, etc)

```
<TEXTAREA NAME="id" [COLS=nn] [ROWS=nn]>default text</TEXTAREA>
```

  - Since the size of text inputed is usually unlimited, you use TEXTAREA with POST method

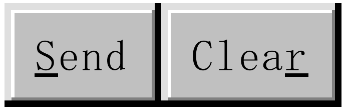


# Action buttons, Image buttons

- Action buttons, "SUBMIT" and "RESET"

  ```
  <INPUT TYPE="SUBMIT" VALUE="Send">
  <INPUT TYPE="RESET" VALUE="Clear">
  ```

  - Don't include multiply SUBMIT or RESET button in one simple form



- Image button sends co-ordinates!

<input name="*LOGO*" type="IMAGE" src="/images/POWERED5.GIF">

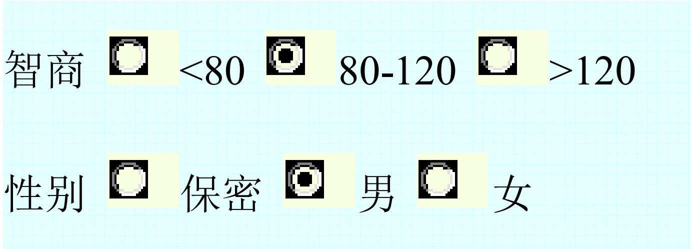
  - Same function as "SUBMIT" button
  - return "*LOGO*.x=??&*LOGO*.y=??" in data
  - left-top corner is (0,0)

# Buttons, Radio or Check Boxes

- Radio button, choose one from many
  - grouped radios shall share the same name

```
智商<input name="IQ" type="RADIO" value="A">&lt80
<input name="IQ" type="RADIO" value="B" CHECKED>80-120
<input name="IQ" type="RADIO" value="C">&gt120
<br>性别<input name="GE" type="RADIO" value="C">保密
<input name="GE" type="RADIO" value="B" CHECKED>男
<input name="GE" type="RADIO" value="A">女
```

IQ=B&GE=B

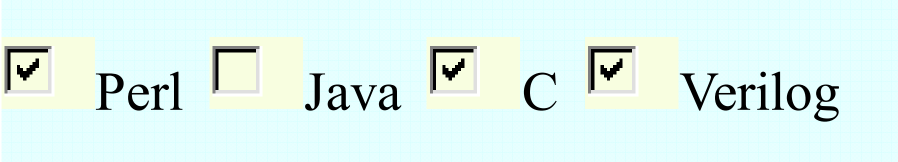


智商 ○ <80 ● 80-120 ○ >120

性别 ○ 保密 ● 男 ○ 女

- Check boxes, on-&-off style
  - Eachshall have a different name

```
<input name="C1" type="CHECKBOX" value="1" CHECKED>Perl
<input name="C2" type="CHECKBOX" value="1">Java
<input name="C3" type="CHECKBOX" value="1" CHECKED>C
<input name="C4" type="CHECKBOX" value="1" CHECKED>Verilog
```



☑ Perl □ Java ☑ C ☑ Verilog

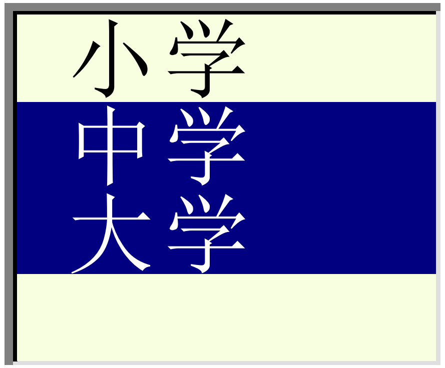
C1=1&C3=1&C4=1

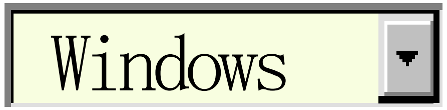
# SELECT/OPTION 单选或多选的列表

- 单选 <SELECT NAME="…">……</SELECT>
- 多选 <SELECT NAME="…" MULTIPLE>……
- 选项 <OPTION VALUE="…" SELECTED>内容

  <OPTION SELECTED>内容

```
<SELECT NAME="platform">
<OPTION>DOS
<OPTION SELECTED>Windows
<OPTION>Linux
</SELECT>
<br>
<SELECT NAME="options" MULTIPLE>
<OPTION>小学
<OPTION>中学
<OPTION>大学
</SELECT>
```





platform=Windows&options=%D6%D0%D1%A7&options=%B4%F3%D1%A7

# Encoding, etc

- HIDDEN, send contextual data

<input type=HIDDEN name="key" value="112233">

- Encoding in GET method
  - "*URL_in_action*" . "*?*" . "*name=val&name=val…*"
  - space converted to "+"
  - special chars (like "+") converted to %??, ?? is the heximal ASCII value of the char. e.g.
    + ➔ %2B
    %➔ %25
    计算机➔%BC%C6%CB%E3%BB%FA

# "POST" method and Decoding

- Useful Environment Variables:
  REQUEST_METHOD & CONTENT_LENGTH
- Data parsing and decoding:

```
$DataLen = $ENV{'CONTEXT_LENGTH'};  #获得数据长度
read(STDIN, $QueryString, $DataLen);    #读入指定长度
@NameValuePairs = split(/&/, $QueryString);
foreach $NameValue (@NameValuePairs) {
   ($Name, $Value) = split(/=/, $NameValue);
   $Value =~ tr/+/ /;            #恢复空格
   $Value =~ s/%([\dA-Fa-f]{2})/pack("C",hex($1))/eg;
                                 #恢复%??编码的字符
   print "Name = $Name, Value = $Value<br>\n";
}
```