

CGI.pm Tutorial

This is a tutorial on CGI.pm which include scripts to let you see the effects. CGI.pm is a Perl module to facilitate the writing of CGI scripts. [Click here](#) to see the details. This tutorial assumes you have a basic knowledge of Perl and HTML.

Table of content

- What is CGI?
 - First program
 - Redirection Request
 - Unpaired Tags
 - Paired Tags
 - Text Field in CGI Form
 - Pop Up Menu in CGI Form
 - Checkbox and Radio Button in CGI Form
-

What is CGI?

The Common Gateway Interface, or CGI, is a standard for external gateway programs to interface with information servers, such as HTTP or Web servers. A plain HTML document that the Web server delivers is static, which means it doesn't change. A CGI program, on the other hand, is executed in real-time, so that it can output dynamic information - perhaps a weather reading, or the latest results from a database query.

[Table of Content]

First Program

The first program just invokes Perl and uses CGI.pm to print the title and heading of the web page.

```
#!/usr/local/bin/perl

use CGI qw(:standard);

print header;
print start_html('This is the title'),
      h1('This is the heading');
print end_html;
```

- The first line specifies where the perl interpreter is.

```
#!/usr/local/bin/perl
```

- The second line includes the CGI Perl module and use standard symbols.

```
use CGI qw(:standard);
```

- The third line prints the HTML header (text/html by default).

```
print header;
```

- The fourth and fifth line prints the HTML title and heading.

```
print start_html('This is the title'),  
      h1('This is the heading');
```

- The last line ends the HTML document by printing the </BODY> and </HTML> tags.

```
print end_html;
```

[[Click here to see the effect.](#) | [Table of Content](#)]

Redirection Request

This program generates a redirection request for the remote browser.

```
#!/usr/local/bin/perl
```

```
use CGI qw(:standard);
```

```
print redirect('http://home.school.net.hk/~hywong/redirect.html');
```

- It will immediately go to the indicated URL, here is,
<http://home.school.net.hk/~hywong/redirect.html>
- Absolute URLs should be used in redirection requests. Relative URLs will not work correctly.

[[Click here to see the effect.](#) | [Table of Content](#)]

Unpaired Tags

This program reveals the use of unpaired tags including
, <P> and <HR>.

```
#!/usr/local/bin/perl
```

```
use CGI qw(:standard);
```

```
print header;  
print start_html(-title=>'This is the title',  
                 -BGCOLOR=>'green');  
print ('This is the first line.');
```

```
print br;  
print ('This is the second line.');
```

```
print p;
```

```
print ('This is the second paragraph');
print hr;
print ('The background color of this page is GREEN')
print end_html;
```

- This lines print out the title of the document and set the background color to green.

```
print start_html(-title=>'This is the title',
                 -BGCOLOR=>'green');
```

- This line prints out the text "
".

```
Print br;
```

- This line prints out the text "<p>".

```
Print p;
```

- This line prints out the text "<hr>".

```
Print hr;
```

[[Click here to see the effect.](#) | [Table of Content](#)]

Paired Tags

This program reveals the use of paired tags including <H1>, , and the like.

```
#!/usr/local/bin/perl

use CGI qw(:standard);

print header;
print start_html(-title=>'Paired Tags',
                 -BGCOLOR=>'yellow');
print h1('The Use of Paired Tags');
print br;
print ('This is an ', i('italic'), 'word.');
```

```
print p;
print ul(
    li('1st item of an unordered list'),
    li('2nd item of an unordered list'),
    li('3rd item of an unordered list')
);
print hr;
print ('The background color of this page is ', em('YELLOW'));
print p;
print a({href=>"http://www.school.net.hk/~hywong/cgipmtut.html#Pair_tag"},
        'GO BACK');
```

```
print end_html;
```

- This line creates the text "<h1>The Use of Paired Tags</h1>".

```
print h1('The Use of Paired Tags');
```

- This line creates the text "This is an *<i>italic</i>* word."

```
print ('This is an ', i('italic'), 'word.');
```

- This line prints out the text:

```
<ul>
```

```
<li>1st item of an unordered list
```

```
<li>2nd item of an unordered list
```

```
<li>3rd item of an unordered list
```

```
</ul>
```

```
print ul(  
    li('1st item of an unordered list'),  
    li('2nd item of an unordered list'),  
    li('3rd item of an unordered list')  
);
```

- This line creates the text "The background color of this page is *YELLOW*".

```
print ('The background color of this page is ', em('YELLOW'));
```

- This line creates the text

```
"<a href="http://www.school.net.hk/~hywong/cgipmtut.html#Pair_tag">GO BACK</a>".
```

```
print a({href=>"http://www.school.net.hk/~hywong/cgipmtut.html#Pair_tag"},  
    'GO BACK');
```

[[Click here to see the effect.](#) | [Table of Content](#)]

Text Field in CGI Form

This program create a CGI form with text fields.

```
#!/usr/local/bin/perl  
  
use CGI qw(:standard);  
  
print header;  
print start_html('CGI Form with Text Field');  
print h1('CGI Form with Text Field');  
print start_form;  
print "What is your name?";  
print textfield(-name=>'field_name',  
    -size=>30);  
  
print p;  
print "What is your comment?";  
print p;  
print textarea(-name=>'large_field_name',  
    -rows=>10,  
    -columns=>50);  
  
print p;  
print submit;
```

```

print end_form;
print hr;

if (param())
{
    print
        "Your name is ", param('field_name'),
        p,
        "Your comment is: ", param('large_field_name'),
        hr;
}

print a({href=>'http://www.school.net.hk/~hywong/cgipmtut.html#Text_field'},
    'Go Back');
print end_html;

```

- This line is used to start a CGI form by creating the <form> tag.

```
print start_form;
```

- These lines create a text field of 30 characters. The name of this text field is "field_name".

```
print textfield(-name=>'field_name',
    -size=>30);
```

- These lines create a large text field with 10 rows and 50 columns. The name of this text field is "large_field_name".

```
print textarea(-name=>'large_field_name',
    -rows=>10,
    -columns=>50);
```

- This line creates the submit button.

```
print submit;
```

- This line is used to end a CGI form by creating the </form> tag.

```
print end_form;
```

- These lines handle the information submitted.

```

if (param())
{
    print
        "Your name is ", param('field_name'),
        p,
        "Your comment is: ", param('large_field_name'),
        hr;
}

```

[[Click here to see the effect.](#) | [Table of Content](#)]

Pop Up Menu in CGI Form

This program create a CGI form with pop up menu.

```
#!/usr/local/bin/perl

use CGI qw(:standard);

print header;
print start_html('CGI Form with Pop Up Menu');
print h1('CGI Form with Pop Up Menu');
print start_form;
print "What is your favourite color?";
print popup_menu(-name=>'color',
                 -values=>['red','orange','yellow',
                           'green','blue','indigo','violet'],
                 -default=>'blue');

print p;
print submit(-name=>'submit button');
print ' ';
print reset;
print end_form;
print hr;

if (param())
{
    print
        "Your favourite color is ", param('color'),
        hr;
}

print a({href=>'http://www.school.net.hk/~hywong/cgipmtut.html#Pop_up_menu'},
        'Go Back');
print end_html;
```

- This line creates a pop up menu of seven values, while "blue" is the default value.

```
print popup_menu(-name=>'color',
                 -values=>['red','orange','yellow',
                           'green','blue','indigo','violet'],
                 -default=>'blue');
```

- This line creates the submit button, with "submit button" as its name.

```
print submit(-name=>'submit button');
```

- This line creates the reset button.

```
print reset;
```

[[Click here to see the effect.](#) | [Table of Content](#)]

Checkbox and Radio Button in CGI Form

This program create a CGI form with checkboxes and radio buttons.

```
#!/usr/local/bin/perl
```

```

use CGI qw(:standard);

print header;
print start_html('CGI Form with Pop Up Menu');
print h1('CGI Form with Pop Up Menu');
print start_form;
print "What is your favourite color?";
print checkbox_group(-name=>'color',
                    -values=>['red ','orange ','yellow ','
                              'green ','blue ','indigo ','violet '],
                    -default=>['red ','blue ']);

print p;
print radio_group(-name=>'color blind',
                 -values=>['Yes','No'],
                 -default=>'No');

print p;
print submit(-name=>'submit button');
print ' ';
print reset;
print end_form;
print hr;

if (param())
{
    print
        "Your favourite color: ", param('color'),
        p,
        "Color blind? ---> ", param('color blind'),
        hr;
}

print a({href=>'http://www.school.net.hk/~hywong/cgipmtut.html#Checkbox'},
      'Go Back');
print end_html;

```

- These lines create seven checkboxes, with "red" and "blue" as the default values.

```

print checkbox_group(-name=>'color',
                    -values=>['red ','orange ','yellow ','
                              'green ','blue ','indigo ','violet '],
                    -default=>['red ','blue ']);

```

- These lines create two radio buttons, with "No" as the default value.

```

print radio_group(-name=>'color blind',
                 -values=>['Yes','No'],
                 -default=>'No');

```

[[Click here to see the effect.](#) | [Table of Content](#)]

This page is brought to you by Wong Ho Yiu Castor.

Created on 24th August, 1997.

Last modified on 30th August, 1997.