



CGI

Common Gateway Interface

- Georgia Tech 1995 Web Usage Survey
 - Perl - 46.7%
 - C - 12.5%
 - Shell Scripts - 8.1%
 - Tcl - Tool Commercial Language
 - Visual Basic
 - Java
 - C++
 - AppleScript

Example Shell Script

```
#!/bin/sh  
#Filename: hello.sh.cgi  
echo "Content-type: text/html"  
echo  
echo  
"<html><head><title>Hello!</title></  
head>"  
echo "<body> Hello World! </body>  
</html>"
```


Hello World! in Perl

```
#!/usr/bin/perl  
print "Content-type: text/html\n\n";  
print  
    "<html><head><title>Hello!</title>  
    </head> \n";  
print "<body> Hello, World! </body>  
    <html>\n";
```


Hello World! in C

```
#include <stdio.h>
main()
{
    printf("Content-type: text/html\n\n");
    printf("<html><head><title>Hello!</title>
    </head>\n");
    printf("<body>Hello,
    World!</body><html>\n");
}
```


Environment Variables Using a Shell Script

```
#!/bin/sh
```

```
echo Content-type: text/plain  
echo
```

```
echo GATEWAY_INTERFACE =  
  $GATEWAY_INTERFACE  
echo REQUEST_METHOD = $REQUEST_METHOD  
echo SCRIPT_NAME = $SCRIPT_NAME  
echo QUERY_STRING = $QUERY_STRING  
echo SERVER_SOFTWARE = $SERVER_SOFTWARE  
echo SERVER_NAME = $SERVER_NAME  
echo SERVER_PROTOCOL = $SERVER_PROTOCOL  
echo SERVER_PORT = $SERVER_PORT  
echo HTTP_USER_AGENT = $HTTP_USER_AGENT  
echo HTTP_ACCEPT = $HTTP_ACCEPT
```


Environment Variables Using a Shell Script

```
echo PATH_INFO = $PATH_INFO  
echo PATH_TRANSLATED = $PATH_TRANSLATED  
echo REMOTE_HOST = $REMOTE_HOST  
echo REMOTE_ADDR = $REMOTE_ADDR  
echo REMOTE_USER = $REMOTE_USER  
echo REMOTE_IDENT = $REMOTE_IDENT  
echo AUTH_TYPE = $AUTH_TYPE  
echo CONTENT_TYPE = $CONTENT_TYPE  
echo CONTENT_LENGTH = $CONTENT_LENGTH
```


Environment Variables in Perl

```
#!/usr/bin/perl
print "Content-type: text/plain\n\n"

print "GATEWAY_INTERFACE =
      $ENV{'GATEWAY_INTERFACE'}\n"
print "REQUEST_METHOD =
      $ENV{'REQUEST_METHOD'}\n"
print "SCRIPT_NAME = $ENV{'SCRIPT_NAME'}\n"
print "QUERY_STRING =
      $ENV{'QUERY_STRING'}\n"
print "SERVER_SOFTWARE =
      $ENV{'SERVER_SOFTWARE'}\n"
print "SERVER_NAME = $ENV{'SERVER_NAME'}\n"
print "SERVER_PROTOCOL =
      $ENV{'SERVER_PROTOCOL'}\n"
```


Environment Variables in Perl

```
print "SERVER_PORT = $ENV{'SERVER_PORT'}\n"
print "HTTP_USER_AGENT =
  $ENV{'HTTP_USER_AGENT'}\n"
print "HTTP_ACCEPT = $ENV{'HTTP_ACCEPT'}\n"
print "PATH_INFO = $ENV{'PATH_INFO'}\n"
print "PATH_TRANSLATED =
  $ENV{'PATH_TRANSLATED'}\n"
print "REMOTE_HOST = $ENV{'REMOTE_HOST'}\n"
print "REMOTE_ADDR =
  $ENV{'REMOTE_ADDR'}\n"
print "REMOTE_USER = $ENV{'REMOTE_USER'}\n"
print "REMOTE_IDENT =
  $ENV{'REMOTE_IDENT'}\n"
```


Environment Variables in Perl

```
print "AUTH_TYPE = $ENV{'AUTH_TYPE'}\n"  
print "CONTENT_TYPE =  
    $ENV{'CONTENT_TYPE'}\n"  
print "CONTENT_LENGTH =  
    $ENV{'CONTENT_LENGTH'}\n"
```


Environment Variables in C

```
#include <stdio.h>
#include <stdlib.h>
char *cp;
char *empty = "[empty]"
main()
{
printf ("Content-type: text/plain\n\n");

#define safenv(a) ((cp = getenv(a)) ? cp : empty)

printf("GATEWAY_INTERFACE = %s\n",
    safenv("GATEWAY_INTERFACE"));
```


Environment Variables in C

```
printf("REQUEST_METHOD = %s\n",  
      safenv("REQUEST_METHOD"));  
printf("SCRIPT_NAME = %s\n",  
      safenv("SCRIPT_NAME"));  
printf("QUERY_STRING = %s\n",  
      safenv("QUERY_STRING"));  
printf("SERVER_SOFTWARE = %s\n",  
      safenv("SERVER_SOFTWARE"));  
printf("SERVER_NAME = %s\n",  
      safenv("SERVER_NAME"));  
printf("SERVER_PROTOCOL = %s\n",  
      safenv("SERVER_PROTOCOL"));  
printf("SERVER_PORT = %s\n",  
      safenv("SERVER_PORT"));  
printf("HTTP_USER_AGENT = %s\n",  
      safenv("HTTP_USER_AGENT"));
```


Environment Variables in C

```
printf("HTTP_ACCEPT = %s\n",  
      safenv("HTTP_ACCEPT"));  
printf("PATH_INFO = %s\n",  
      safenv("PATH_INFO"));  
printf("PATH_TRANSLATED = %s\n",  
      safenv("PATH_TRANSLATED"));  
printf("REMOTE_HOST = %s\n",  
      safenv("REMOTE_HOST"));  
printf("REMOTE_ADDR = %s\n",  
      safenv("REMOTE_ADDR"));  
printf("REMOTE_USER = %s\n",  
      safenv("REMOTE_USER"));
```


Environment Variables in C

```
printf("REMOTE_IDENT = %s\n",  
      safenv("REMOTE_IDENT"));  
printf("AUTH_TYPE = %s\n",  
      safenv("AUTH_TYPE"));  
printf("CONTENT_TYPE = %s\n",  
      safenv("CONTENT_TYPE"));  
printf("CONTENT_LENGTH = %s\n",  
      safenv("CONTENT_LENGTH"));  
}
```


Standard CGI Variables

- User information
 - HTTP_USER_AGENT
 - Name and version of user's browser
 - *name/version library/version*
 - Contains information about any proxy gateway
 - Used when firewalls exist

Standard CGI Variables

- User information
 - HTTP_USER_AGENT
 - Examples
 - Mozilla/2.0b1J (Windows; I; 32 bit)
 - Mozilla/1.22 (Windows; I; 16 bit)
 - Lynx/2.3.7 BETA libwww/2.14
 - Microsoft Internet Explorer/4.40.308 (Windows 95)

Standard CGI Variables

- User information
 - HTTP_ACCEPT
 - Gives MIME formats that the browser can accept
 - REMOTE_HOST
 - IP address of user
 - Always given
 - REMOTE_ADDR
 - Text-equivalent host name of user IP address
 - Not always given

Standard CGI Variables

- Server information
 - SERVER_SOFTWARE
 - Name and version of server software
 - *name/version*
 - Apache/0.6.4b
 - SERVER_NAME
 - - www.cs.wayne.edu
 - GATEWAY_INTERFACE
 - CGI revision
 - CGI/*revision*
 - CGI/1.1

Standard CGI Variables

- Request-specific information
 - QUERY_STRING
 - Very important
 - Most common method of passing information to server
 - Suppose `http://www.cs.wayne.edu/cgi/code.cgi?x,y` URL is submitted
 - All characters after the **?** will be put into QUERY_STRING

Standard CGI Variables

- Request-specific information
 - SCRIPT_NAME
 - File name of the CGI program
 - SERVER_PROTOCOL
 - Name and revision of the protocol that request came in from
 - *protocol/revision*
 - HTTP/1.0

Standard CGI Variables

- Request-specific information
 - SERVER_PORT
 - Port on which request came in
 - Usually 80
 - PATH_INFO
 - One can pass another file path to the CGI program by appending it to the URL
 - <http://www.cs.wayne.edu/cgi/cgi-code.cgi/a/b/c>
 - PATH_INFO will contain the extra path /a/b/c

Standard CGI Variables

- Request-specific information
 - PATH_TRANSLATED
 - Contains PATH_INFO appended to the document root path of the server
 - /x/y/z/a/b/c
 - CONTENT_TYPE
 - Used for queries that have attached information, such as POST requests
 - MIME content-type of data
 - *type/subtype*
 - application/x-www-form-urlencoded

Standard CGI Variables

- Request-specific information
 - CONTENT_LENGTH
 - Number of bytes of data
 - AUTH_TYPE
 - Contains authentication method used to validate user
 - REQUEST_METHOD
 - Method used for request
 - Usually either *post* or *get*