

Lesson two: Your first CGI program.

CGI programs must be perfect or they will not work so everything you do from now on cannot have any errors in it and you cannot test your CGI programs in your browser before you load them to the server like HTML programs.

Not only is CGI/Perl the most difficult programming to learn it is also the slowest to complete and make-work because you can't try it out on your computer. You must load it to the server to see if you have done it correctly.

If you didn't read and complete everything in the section, *What you need to use this book*, Go back and do it now.

To make our first CGI program we will mix what you learned in lesson one with lesson two. We will mix HTML with CGI.

Every CGI/Perl program starts with a line that looks something like this:

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

This is why you must follow the instructions in the section, *What you need to use this book*. This is the path to the Perl interpreter. Although it may vary from server to server it tells the program where to find Perl.

Most all lines in Perl must end with a semicolon, “;”.

If your going to make a Perl program that will create a web page you must include the line,

```
print "content-type: text/html\n\n";
```

This is called the *Content header*. It tells the program that it will be viewed in a browser. The “\n\n” at the end of the line tells the program to make two line breaks. It's like pressing the return key twice and it also tells it to not stop, keep going. The word print is a command for Perl. You need to put the content header right after the line to the Perl interpreter.

```
#!/usr/local/bin/perl  
print "content-type: text/html\n\n";
```

Step 1. Start the program Crimson Editor and make sure the screen is blank so it looks like illustration 17.

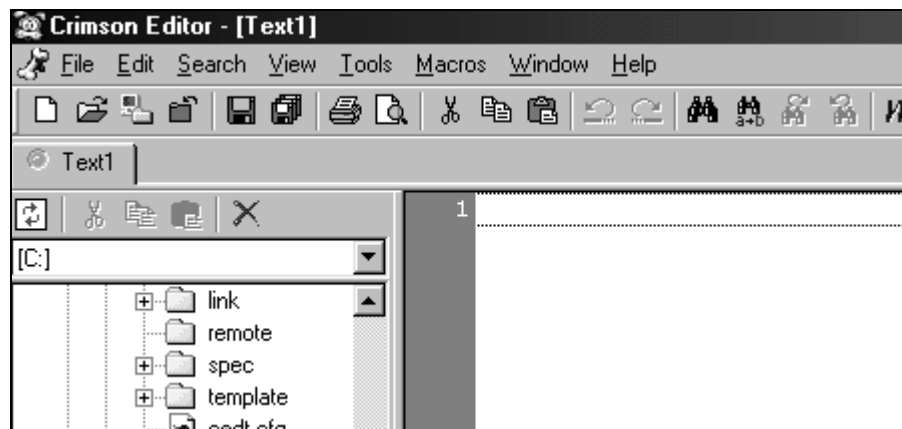


Illustration 17

Step 2. First enter into the editor the path to Perl and the content header like illustration 18. MAKE SURE your path to Perl is the correct one for your server.

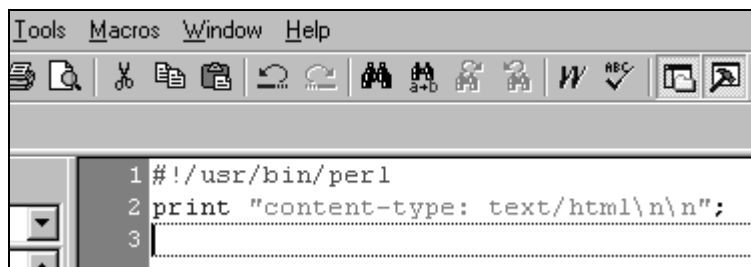


Illustration 18

Step three. Now add the Perl command *print* and add the two words *Hello World* so it looks like illustration 19. The Perl command *print* makes what ever you put after it print to the screen.



Illustration 19

Step 4. Now were going to save the file. I will not go threw how to save a file every time you need to do this so after lesson two you should remember how to do it. If you don't, you can always return to lesson two and review it. Move your mouse pointer up to the top left of your editor and click once on the word that says *File* like illustration 20.



Illustration 20

Step 5. A menu will open like in illustration 21. Move your mouse pointer down to the area that says *Save As*.

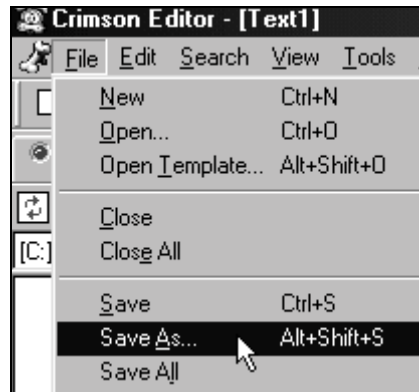


Illustration 21

Step 6. A window will open. Save the file with the name samp1.cgi just like illustration 22. Make sure that you save it to a location that you will remember and easily find.

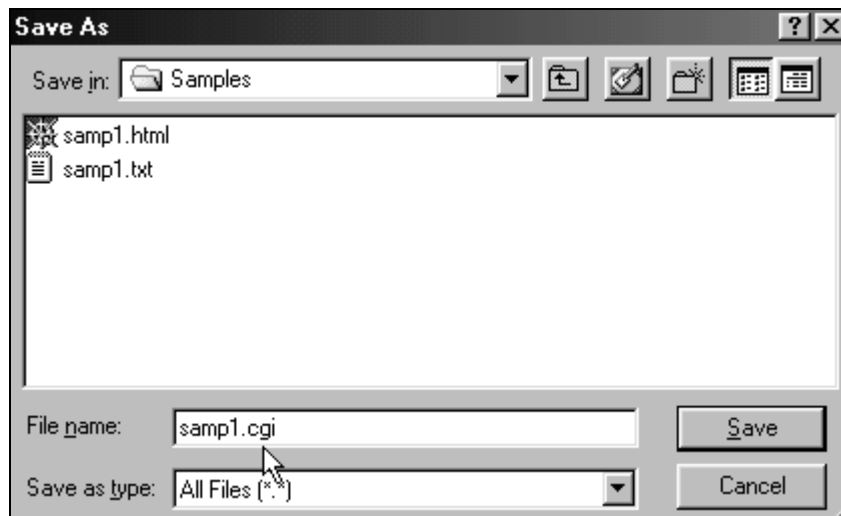


Illustration 22

Step 7. First get connected to the Internet and start the program CuteFTP. A window will open just like illustration 23. Click one on the button that says *Connect*.

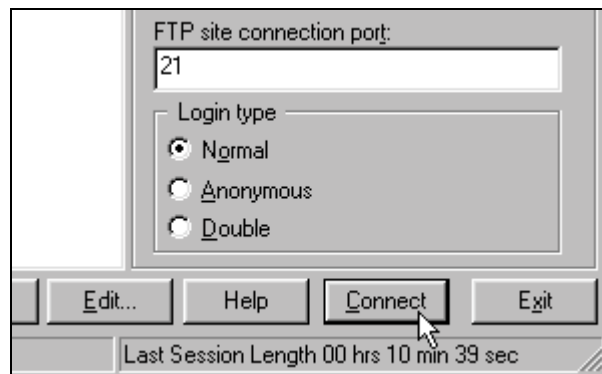


Illustration 23

Step 8. Go to the right side of CuteFTP and line up the directories so you can put your file in your *cgi-bin* directory. Your correct area may be different than illustration 24.

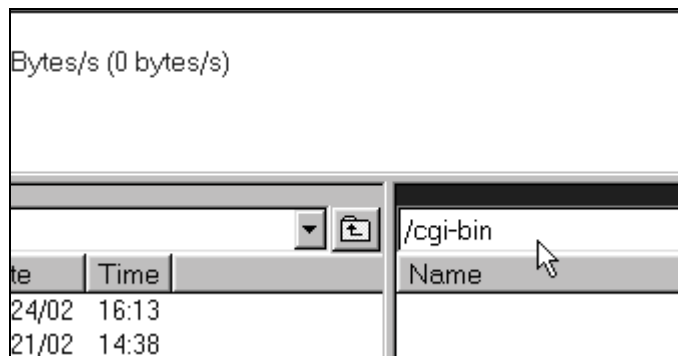


Illustration 24

Step 9. Now go to the left side and line up the directories so that you have the correct path to your file named *samp1.cgi* like illustration 25. Yours will probably be different.

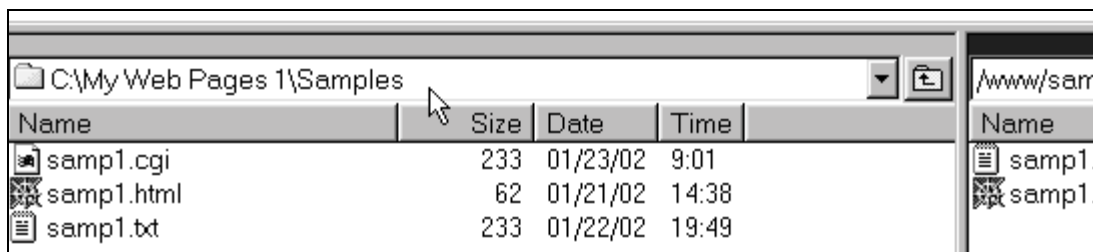


Illustration 25

Step 10. You must load all CGI files to the server as ASCII text. To do this you must first tell CuteFTP to the load it that way. Go up with your mouse pointer to the Edit menu and click once like illustration 26.

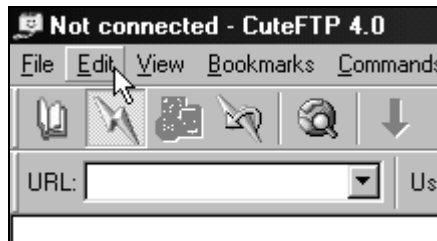


Illustration 26

Step 11. A menu will open. Move your mouse pointer to the area that says *Settings* as in illustration 27 and click once.

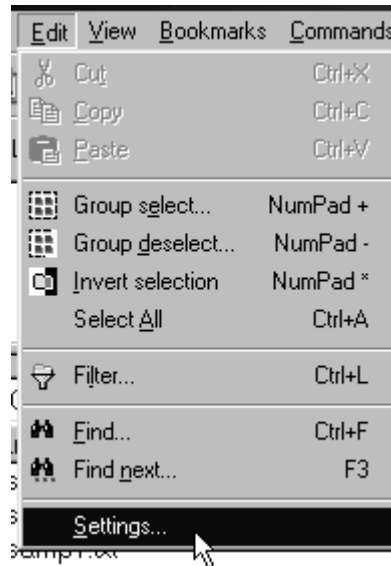


Illustration 27

Step 12. A window will open. Click once on the radio button that says *ASCII* and then click the *Ok* button like illustration 28.

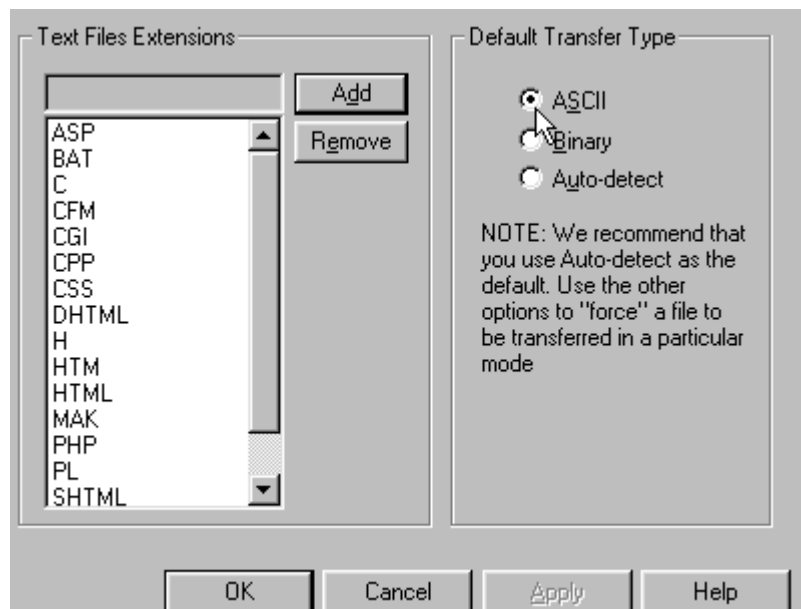


Illustration 28

Step 13. Now you need to load your file to the server. Go to the left side of CuteFTP and left click once on your file named *samp1.cgi*. Hold down the left mouse button and drag and drop the file to the right side of CuteFTP and let go of the mouse button as in illustration 29.

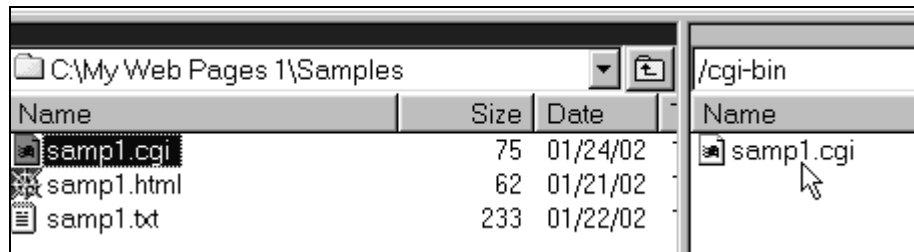


Illustration 29

It is time when you need to learn about file permissions and the CHMOD command. CHMOD stands for “Change Mode”. It is a Unix command that sets the file permission on CGI files. It tells the server how much or how little access a web user should be permitted to a file.

A fair amount of mistakes are caused by not setting your file to the correct permission. Using the CHMOD command makes your CGI file so it can be written to, executed or made so only you can do anything with it and only viewed by web surfers. I will use some charts so you can understand it better.

CHMOD Permission Chart 1				
	Owner (X)	Group (Y)	All (Z)	
Read = 4	X	Y	Z	
Write = 2	X			
Execute = 1	X	Y	Z	
	4+2+1 = 7	4+1 = 5	4+1 = 5	CHMOD=755

Every CGI/Perl file that you load to a UNIX server must have the permissions set on it. The creator of the file can assign different protections on the file either granting or prohibiting access.

The top three boxes of the CHMOD chart are explained like this:

Owner = The person who created the file.

Group = A group of users can be combined in a single user group for the file.
I have never yet had a use for this but it must be set or your program won't work.

All = Means that all web users can execute the program.

Read = A web surfer can read the contents of the file.

Write = A web surfer can change the contents of the file.

Execute = A web surfer can make the file work.

Lets look at your file *samp1.cgi*. You want the owner, “that’s you”, to have read, write and execute permission. You want groups to be able to read it and execute it but not write to it and you want all to read it and execute it but not write to it so you need to set the permission of the file to 755.

Step 14. Put your mouse pointer to the right side of CuteFTP and right click on your file named *samp1.cgi* like illustration 30.

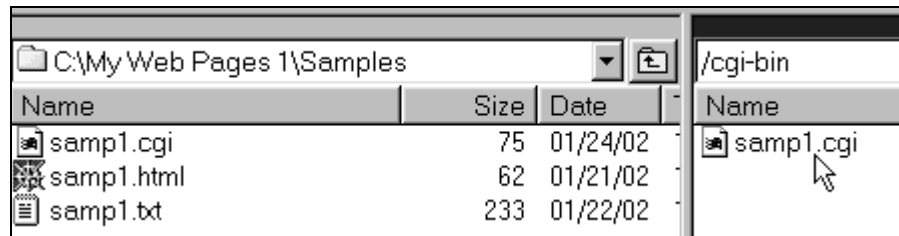


Illustration 30

Step 15. A menu will open as in illustration 31. Move your mouse down to where it says *CHMOD* and left click on it once.

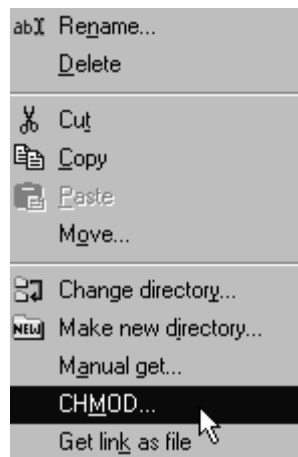


Illustration 31

Step 16. A window will open like in illustration 32. Make sure that the little text box with the numbers in says 755 and click once on the *Ok* button.

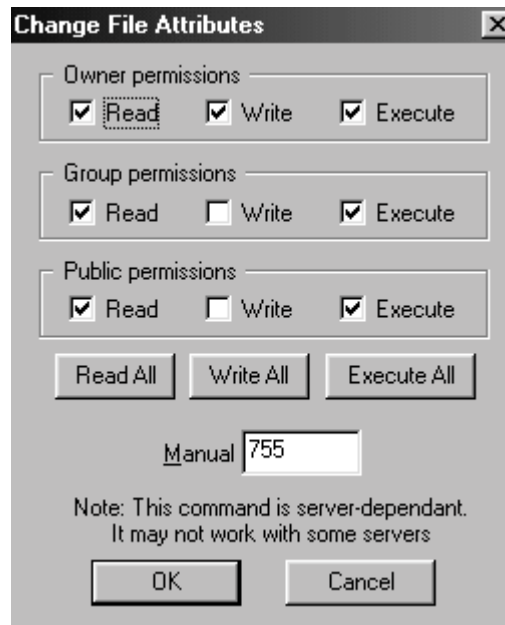


Illustration 32

Step 17. You have finished setting the file permission on the file. Start your browser and fill in the location area so it has a path to your file in the *cgi-bin* directory like illustration 33. Yours won't look just like mine. Now press the *Return* key.



Illustration 33

Your first CGI program should be working now as in illustration 34. If it is not, lets look at some of the reasons it might not have worked.



Illustration 34

1. Is your path to Perl the correct one for your web site?
2. Is your file in the *cgi-bin* directory?
3. Was your file loaded to your web site in ASCII mode?
4. Did you CHMOD the file to 755?
5. Is your program exactly like illustration 19?

Common server error warnings and what they mean:

"403 Forbidden"

This tells you that the file permissions, are most likely, set incorrectly.

"404 File Not Found"

This means that you are trying to access a file that is not there. Double check your URL and where you put the file. Remember some servers are case sensitive

"500 Server Error"

This means that your program is producing output that your server cannot
Make sure that you did not make any typos in your program.