Lesson Four: Operators

Math Operators

We need to look at Perl Operators and how they work. The upcoming lessons will have a lot of operators as control structures. Math Operators are used to create calculations on number. There are more operators than shown here but for now study the Math Operators Chart One.

MATH OPERATOR CHART ONE		
Operator	Function	Example
+	Addition	1 + 2 = 3
-	Subtraction	3 – 2 = 1
*	Multiplication	3 * 2 = 6
/	Division	6 / 2 = 3

These operators are used to help store sums in variables. If you want to use Perl to do arithmetic it would look something like this.

```
$ Cost1 = 10;
$Cost2 = 20;
$Total = $Cost1 + $Cost2;
print $Total;
```

Step 1. We need to make sure this works on your web site so open your editor and fill it in just like illustration 53. Make sure your path to Perl is correct for your server. Building and testing programs is the best way to learn any programming language.

```
1 #!/usr/bin/perl
2 print  "content-type: text/html\n\n";
3 $ Cost1 = 10;
4 $Cost2 = 20;
5 $Total = $Cost1 + $Cost2;
6 print $Total;
7
```

Illustration 53

Step 2. Save the file as samp7.cgi.

Step 3. Connect to the Internet and Start CuteFTP and log in to your web site. If you don't remember how to do this review Lesson 3.

Step 4. Make sure your file transfer protocol is set to ASCII and drag and drop the file, *Samp7.cgi* to your cgi-bin directory.

Step 5. CHMOD the file to 755.

Step 6. Start your favorite browser and line up the URL path to your file and press the Enter key. Your screen should show the number 30 just like illustration 54.

Illustration 54

We need to learn how to percent calculations for use in our programs. The need will arise for finding what the sales tax is and lots of other reasons. The percent sign "%" is used for a lot of things in Perl but we will not go into that now.

Let's say we need to figure out how to find your states sales tax for use in a program. The sales tax in my county in Florida is 6 percent, "6%". 6 percent would be written as point zero six, ".06". To find the sales tax needed to collect from buying a product that cost 30 dollars you multiply 30 dollars times 6 percent, "30.00 times .06" or "30 * .06". It would look like this:

```
$Cost1 = 10;
$Cost2 = 20;
$SubTotal = $Cost1 + $Cost2;
$Tax = .06;
$SalesTax = $SubTotal * $Tax;
$Total = $SubTotal + $SalesTax;
print $Total;
```

Let's break this down for all of math challenged people.

Example	Definition
$Cost1 = 10;	The cost of the first product is 10.00 dollars.
$Cost2 = 20;	The cost of the second product is 20.00 dollars.
$SubTotal = $Cost1 + $Cost2;	The total without adding the sales tax is found by adding $Cost1 to $Cost2.
$Tax = .06;	The sales tax for your state is 6 percent.
$SalesTax = $SubTotal * $Tax;	You find what the sales tax would be by multiplying the sub total times what percent the sales tax is.
$Total = $SubTotal + $SalesTax;	To find what a person would have to pay in your state to buy two products add the sub total plus the amount of the sales tax.
print $Total;	This prints the total amount to the screen.

Not all states or counties have an even number for their sales tax. This is important that you understand sales tax because if your going to program in any computer language you will need to explain it to your clients. The county I used to live in in Florida was Monroe County and they have a sales tax of seven and one half percent, "7.5% or .075".

Step 7. Start a new page in your editor and fill it in just like illustration 55.

```
1  #!/usr/bin/perl
2  print  "content-type: text/html\n\n";
3  $Cost1 = 10;
4  $Cost2 = 20;
5  $SubTotal = $Cost1 + $Cost2;
6  $Tax = .075;
7  $SalesTax = $SubTotal * $Tax;
8  $Total = $SubTotal + $SalesTax;
9  print $Total;
10
```

Illustration 55

Step 8. Save the file as samp8.cgi.

Step 9. Connect to the Internet and Start CuteFTP and log in to your web site. If you don't remember how to do this review Lesson 3.

Step 10. Make sure your file transfer protocol is set to ASCII and drag and drop the file, *Samp8.cgi* to your cgi-bin directory.

Step 11. CHMOD the file to 755.

Step 12. Start your favorite browser and line up the URL path to your file and press the Enter key. Your screen should show the number 32.25 just like illustration 56.

Illustration 56

32.25 means 32 dollars and 25 cents. $Cost1 plus $Cost2 equals 30 dollars. 7and one half percent of 30 dollars is 2 dollars and 25 cents of sales tax. Add 30 and 2.25 together and you get 32.25.

Math Comparison Operators

Comparison Operators are used to compare two numbers with each other. They are not used to compare strings. Study the Comparison Operator Chart One.

MATH COMPARISON OPERATOR CHART ONE		
Operator	Function	Example
<	Less Than	1 < 2
>	Greater Than	2 > 1
= =	Equal To	2 = = 2
! =	Not Equal To	2 ! = 1
>=	Greater Than Or Equal To	2 >= 1
<=	Less Than Or Equal To	1 <= 2

To put Math Comparison Operators to use we need to use the Perl statements “if(), else() And elsif(). Study this example:

```
$Cost1 = 10;
```

```
if($Cost1 = = 10)
{
print “It cost 10 dollars”;
}
```

Notice that line one only uses 1 equal sign and when used in certain Perl statements you must use 2 equal signs side by side with no space in between them. It may look like there is a space between them in the printing of this book but don’t let that confuse you.

This would execute the print statement because the condition was true. If the condition were false it would have been skipped over and nothing would be printed. Nothing is wrong when your programming has to skip over something. In this case it is good. Let’s make a larger program.

```
if(9 < 9) {
print “9 is less than 9\n”;
}

else{
print “9 is not less than 9\n”;
{
```

This program would print what is in the Else Statement. Let’s try adding the elsif() statement.

```
if(9 < 9) {
print “9 is less than 9\n”;
}

else{
print “9 is not less than 9\n”;
}

elsif(9 >= 9) {
print “9 is greater than or equal to 9\n”;
}
```

If you try out this program you will see that it would not work. It should work but it won’t. Why? Perl programming does not like the elsif() statement. I use to do a lot of C programming too and it didn’t either. So you must use your creativity and stay away from the elsif() statement. This program will get around it.

```
if(9 < 9) {
print “9 is less than 9\n”;
}

else{
print “9 is not less than 9\n”;
}
```

```
if(9 >= 9) {
print "9 is greater than or equal to 9\n";
}
```

This program will work.

Step 13. It's time to try some statements. Start your editor or if it is already open start a new page. Fill it in just like illustration 57 and save the file as *samp9.cgi*. Make sure you use your correct path to Perl and if the "\n" doesn't start a new line use the "
" statement. Remember you must get everything perfect or it will not work.

```
1  #!/usr/bin/perl
2  print  "content-type: text/html\n\n";
3
4  if (9 != 9){
5      print "The cost is 9 dollars\n";
6  }
7  else {
8      print "Let's try this again\n";
9  }
10 if (9 == 9){
11     print "The cost is 9 dollars\n";
12 }
13
```

Illustration 57

Step 14. Connect to the Internet and Start CuteFTP and log in to your web site. If you don't remember how to do this review Lesson 3.

Step 15. Make sure your file transfer protocol is set to ASCII and drag and drop the file, *samp9.cgi* to your cgi-bin directory.

Step 16. Start your browser and line up the URL path to your file and press the Enter key. Your screen should show the same as illustration 58.

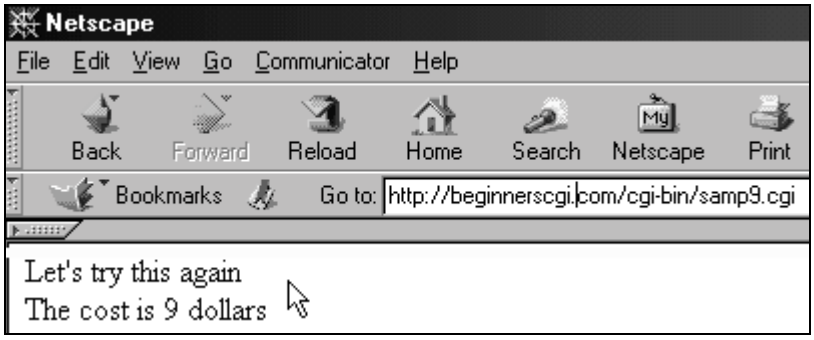

Illustration 58

String Comparison Operators

String Comparison Operators work like Math Comparison Operators but work on strings, (words and sentences). The difference is that letters can be lower case, (small letters), and upper case, (large letters). Anytime you compare strings remember that “abcdefg” is not the same as “Abcdefg”.

Study the String Comparison Operator Chart One.

STRING COMPARISON OPERATOR CHART ONE		
Operator	Function	Example
eq	Equal	door eq door
ne	Not Equal	door ne Door
lt	Less Than	Abc lt abc
gt	Greater Than	abcd gt abc
le	Less Than Or Equal To	ab le abc
ge	Greater Than Or Equal To	abc ge ab

String comparisons are great for password scripts. It is a little strange the way some of the operators work. “ABC” is equal to “ABC” but “ABC” is not equal to “aBc”. The less than and greater than string operators use alphabetical order to compare strings and do it backwards so “A” is greater than “B” and “dude” is greater than “DUDE”.

Step 17. Open your editor or start a new page and fill it in like illustration 59.

```
1 #!/usr/bin/perl
2 print  "content-type: text/html\n\n";
3
4 if("mypassword" eq "mypassword"){
5     print "Password is correct";
6     }
7 else {
8     print "Password is incorrect";
9     }
```

Illustration 59

Step 18. Save the file as *samp10.cgi* and start CuteFTP and connect to your server.

Step 19. Transfer the file in ASCII mode and drag and drop it to your cgi-bin directory.

Step 20. Fire up your favorite browser and connect to your file, *samp10.cgi*, and it should look like illustration 60.

Illustration 60

Step 21. We need to make one more complex so you understand String Operators. Start a new page on your editor and fill it in like illustration 61. Don't just type in all the letters and words; take the time to study the String Comparison Operator Chart One as you type.

```
1  #!/usr/bin/perl
2  print  "content-type: text/html\n\n";
3
4  if("mypassword" eq "mypassword"){
5      print "Password is correct<br>";
6      }
7  else {
8      print "Password is incorrect<br>"
9      }
10
11 if("yourface" gt "Yourface"){
12     print "It is greater than<br>";
13     }
14 else {
15     print "It is not greater than<br>
16     }
17
18 if("Mountain" lt "mountain"){
19     print "It is less than<br>";
20     }
21 else {
22     print "It is not less than<br>";
23     }
24
```

Illustration 61

Step 22. Save the file as *samp11.cgi*.

Step 23. Start CuteFTP and make sure it is set to ASCII mode and drag and drop your file to your cgi-bin directory.

Step 24. CHMOD the file to 755.

Step 25. Start your browser and line up the location bar at the top with the file *samp11.cgi* in your cgi-bin directory and it should read just like illustration 62.

Illustration 62

Logical Operators

We need to have a look at one more Perl Operator the Logical Operator. Study the Logical Operator Chart One.

LOGICAL OPERATOR CHART ONE		
Operator	Function	Example
||	Or	if(($Cost < 20) \|\| ($Cost > 1))
&&	And	if(($Cost <= 20) && ($Cost > 2))

To understand this we need an example.

```
If (($Cost < 20) | | ($Cost > 2)) {
   print "I can afford it";
}
else {
   print " I can not afford it";
}
```

This example says; (if $Cost is less than 20 or $Cost is greater than 2). Make sure that you notice the double parentheses. Double parentheses must be used when this type of operator is used. There are lots of example programs on the Internet that the double parentheses are missing. This tells us that people are putting programs on Web Pages as examples that they never tried out to make sure they worked.

Step 26. Start a new page on your editor and fill it in just like illustration 63. Make sure that the path to Perl is correct for your server. If your web site will work with the new

line statement, "\n" use it otherwise use the "
" statement. Try to figure out what the answers will be before you run the program.

```
1  #!/usr/bin/perl
2  print  "content-type: text/html\n\n";
3  $Cost = 2;
4  if (($Cost >= 10) || ($Cost > 2)) {
5      print "I can afford it<br>";
6  }
7  else {
8      print " I can not afford it<br>";
9  }
10
11 if (($Cost < 10) && ($Cost >= 2)) {
12     print "Now I can afford it<br>";
13 }
14 else {
15     print " I still can't not afford it<br>";
16 }
17
```

Illustration 63

Step 27. Save the file as *samp12.cgi*.

Step 28. Start CuteFTP and connect to your server.

Step 29. Drag and drop your file to your cgi-bin directory.

Step 30. CHMOD it to 755.

Step 31. Start your browser and line up the location bar with your file and press the Enter key. It should read like illustration 64.

Illustration 64