Cognex Designer Scripting/Expression Examples:

```
if (Count>0 && Score > .85)
return true;
else
return false;
```

If the variable COUNT > 0 and the variable SCORE > .85, then the expression = true (otherwise false)

```
if (StringRead == "ABCDE")
return true;
else
return false;
```

If the string variable STRINGREAD = ABCDE, then the expression = true else false

---

```
$Components.Filmstrip.Add(Image);
```

Used in a ScriptBlock to add an image to the Image Shift Register called FILMSTRIP

```
$Components.ShiftColor.Add(Result);
```

Used in a ScriptBlock to add a Boolean result to the Boolean Shift Register called SHIFTCOLOR

```
($System.Users.CurrentAccessLevel > 1)
```

Used to make a button visible (or any other parameter) only when the current logged in user has Access Level > 1

```
$HMI.ShowPage("Login");
```

Display the HMI page called Login.

---

*Query Text: SELECT * FROM [PMScoreTime] ORDER BY Time DESC LIMIT 4*

Used in Query Builder to show latest 4 records of the table definition called PMScoreTime

---

```
if (Selector == 1)
   return Chocolate;
else if (Selector == 2)
   return Pyramid;
else if (Selector == 3)
   return Tire;
else if (Selector == 4)
   return Bottle;
```

Used in SciptBlock to check the value of an Integer variable and pass one of 4 script I/O parameters out of the Script Block.

```
(($Result.Tilt / 3.14159) * 180 ) FORMAT "#0.00")
```

Allows you to display a Radians value ($Result.Tilt) in degrees

---

Use the FORMAT option to display a floating value as needed:

Example: $FloatVar = 1543.456321

To only three decimal places: (F = Fixed Decimal)
$FloatVar FORMAT "F3" = 1543.456

To display in Percentage: (P = Percentage)
$FloatVar FORMAT "P2" = 154,345.632 %

To display in specific digits:
$FloatVar FORMAT "#00.0" = 1543.5

To display rounded to whole number: (P = Percentage)
$FloatVar FORMAT "F0" = 1543

Cognex C# Reference

		Example
Data Types	byte[], char, int, long, double, string	

Variable Declaration	<type> <name>;	double result; // creates floating point variable 'result'
		int count = 0; // creates integer variable count and initializes with 0

Array Declaration	<type>[] <name>	string[] day = { "Mon", "Tue", "Wed", "Thu", "Fri"};
		int[] counts = new int[5];
		// integer array with 5 elements, 0-4

Comment	//	*// this is a comment*
Multi-Line Comment	/* */	/* this is commented */
Arithmetic Operators	+ (Addition), - (Subtraction), * (Multiplication), / (Division), % (Modulus)	a = b + c; minutes = hours * 60;

Increment operator	++
	+=
		i ++; // same as i = i +1
		i += 5; // same as i = i + 5

String Concatenation	+	string msg; msg = "hello" + " " + "world!"; // output: "hello world!"
Relational Operators	< (Less Than), <= (Less Than or Equal To), > (Greater Than), >= (Greater Than or Equal To), == (Equal To), ! = (Not Equal To)	if (result <= 5) return true;
Logical Operators	&& (And), || (Or)	if ((a < 5) || (a > 10)) return false;
String Manipulation	.Substring(<start>,[<length>]) .Trim() <removes all leading and trailing blanks> .ToLower() <to lower case> .ToUpper() <to upper case> .Replace(<find>,<replace>) .Equals(<str>) <compares two strings> .Contains(<str>) <check if it contains str> .Compare(<string1>,<string2>) <for sorting>	string s; string txt = "One line of text"; s = txt.Substring(0,2); -> "One" s = txt.ToUpper(); -> "ONE LINE OF TEXT" if (txt.Contains("line")) return true;

If Else Statement	if (expression)
	{
	<statements>;
	}
	else
	{
	<statements>;
	}
		if (a == b)
		{
		return true;
		}
		else
		{
		return false;
		}

For Loop	for (statement)
	{
	<statements>;
	}
		string[] day = { "Mon", "Tue", "Wed", "Thu", "Fri"};
		for(int i = 0; i<=4; i++)
		{
		$System.Log.Write(day[i]);
		} // outputs: Mon, Tue, Wed, Thu, Fri

For Each Loop	foreach(<variable> In <object>)
	{
	<statements>;
	}
		string txt = "One line of text";
		foreach(char c in txt)
		{
		$System.Log.Write(c);
		} // outputs: o,n,e, ,l,i,n,e, ,o,f, ,t,e,x.t

While Loop	while(<expression>)
	{
	<statements>;
	}
		int sum = 0;
		int i = 1;
		while(i < 5)
		{
		sum = sum + i;
		$System.Log.Write(sum);
		i ++;
		} // outputs: 1, 3, 6, 10