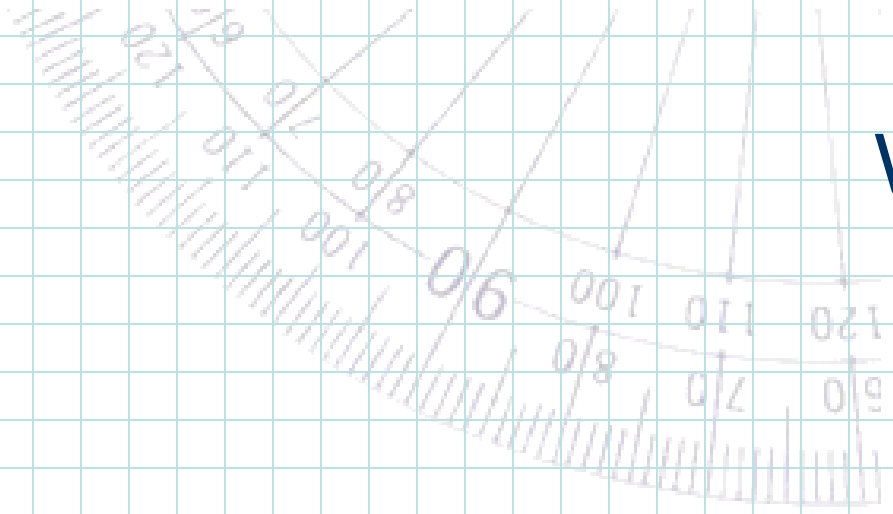
# Intro to Programming: Javascript and Pseudocode

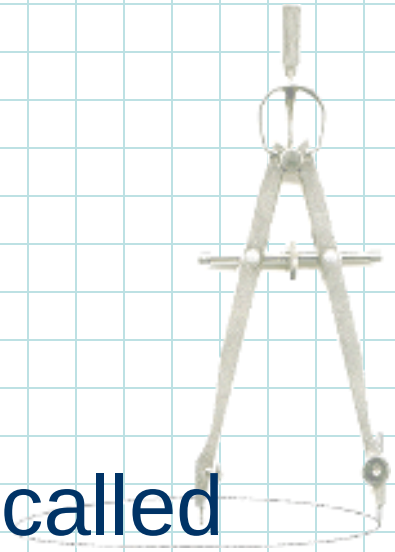


Website Production

# Q: What is Javascript?

- A scripting language from Netscape; only marginally related to Java
- Intended to provide a quicker and simpler language for enhancing Web pages and servers
- Embedded as a small program in a web page, that is interpreted and executed(run) by the Web browser.

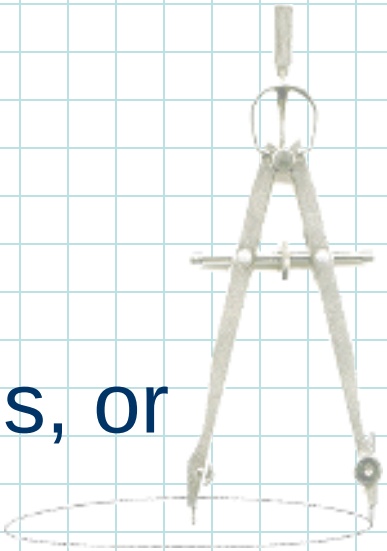
# Q: What is Javascript?



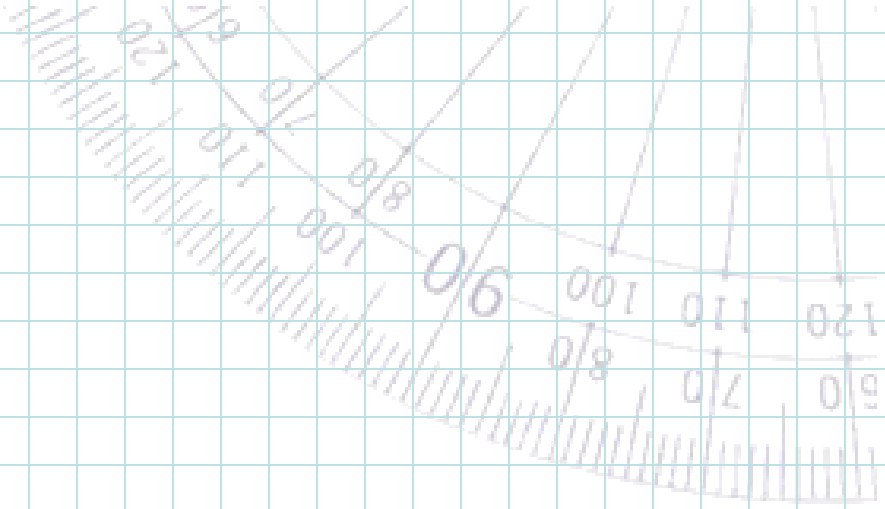
JavaScript functions can be called from within a Web document, often executed by mouse functions, buttons, or other actions from the user

- Can be used to fully control all the familiar browser attributes

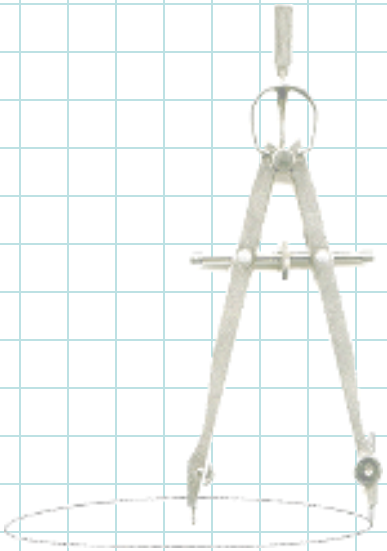
# People & Programs



- *User*: an individual who runs, or executes, a program
- *Programmer*: an individual who creates, or writes,  a program



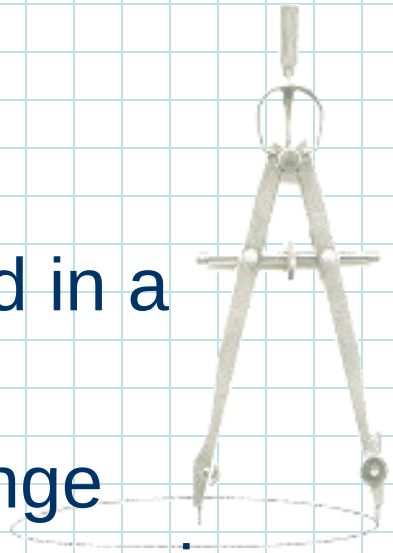
# Program



*Consists of…*

- Declarations
  - Define the use of various identifiers (names), thus creating the elements used by the program (computer)
- Statements
  - Or executable statements, representing actions the computer will take on the user's behalf

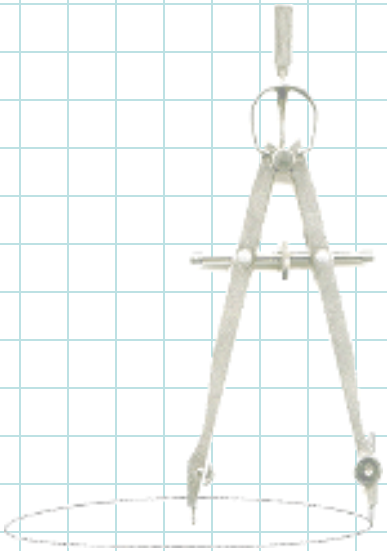
# Identifiers

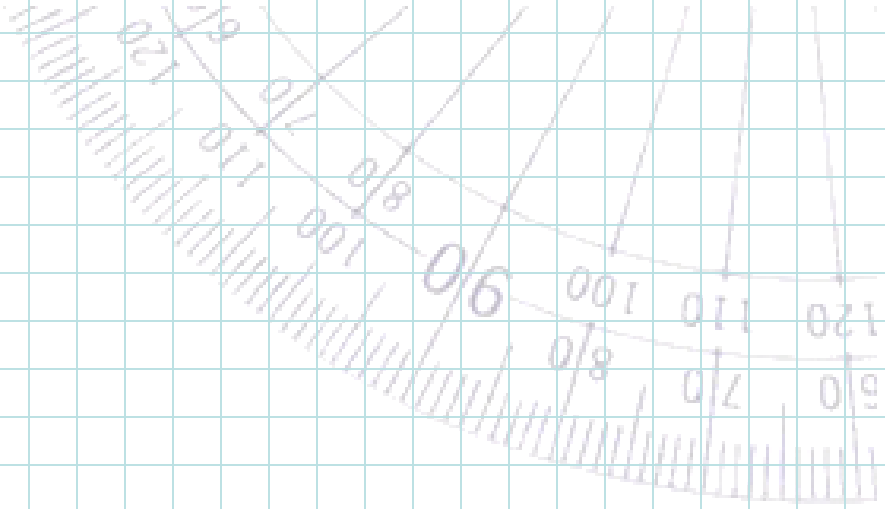


- Names for various entities used in a program; used for...
- *Variables*: values that can change frequently when the program is running
- *Constants*: values that never change when the program is running
- *Functions*: programming units that represents complex operations
- *Parameters*: values that change infrequently and are inputs to functions

# Simple Function

```
DoIt()
{
// Statements
}
```

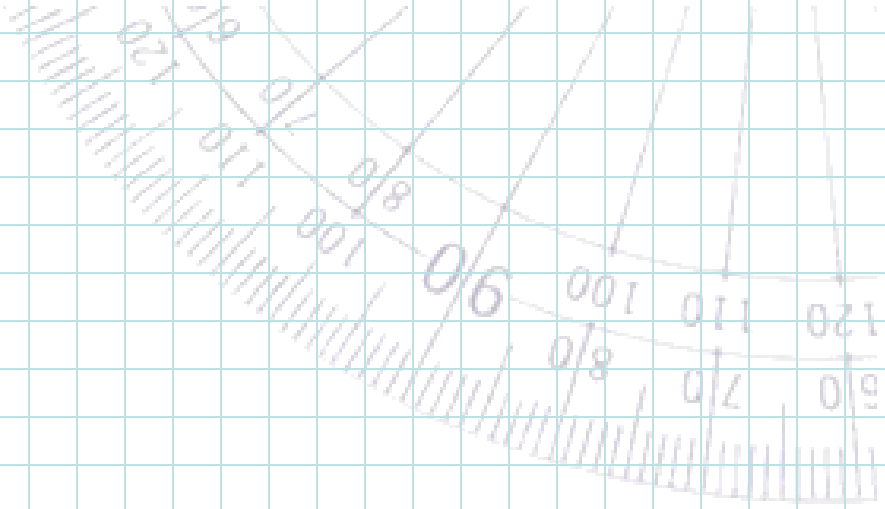




# Simple JavaScript Program (a JavaScript function)

```
function DoIt()
{
// Statements
}
```

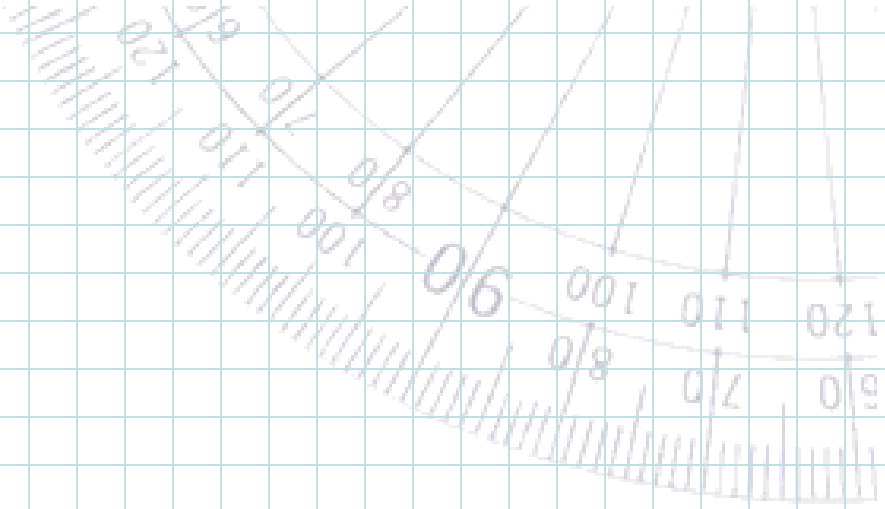
- Header for function
- Consists of…
  - identifier for function
  - list of arguments between parenthesis (none for this function)



# Simple JavaScript Program

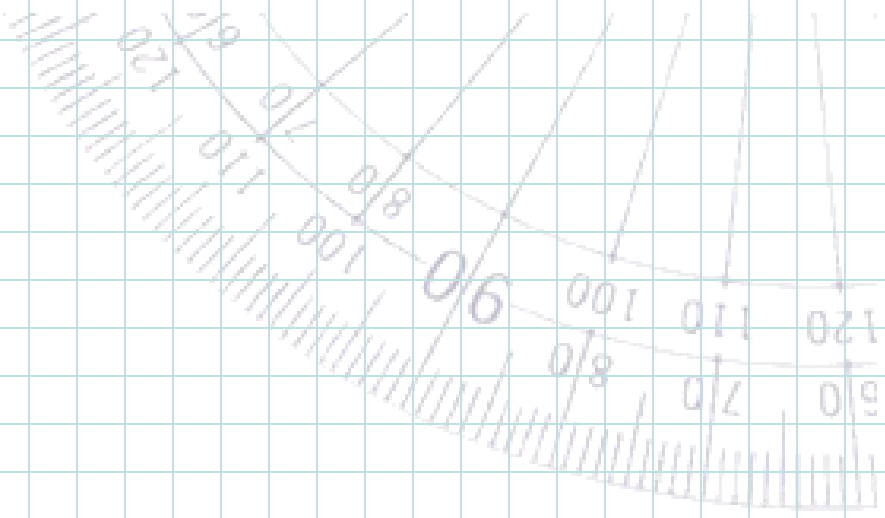
```
function DoIt()
{
// Statements
}
```

- Braces enclose the body of the function
- They represent the start and end of the function



# Simple JavaScript Program

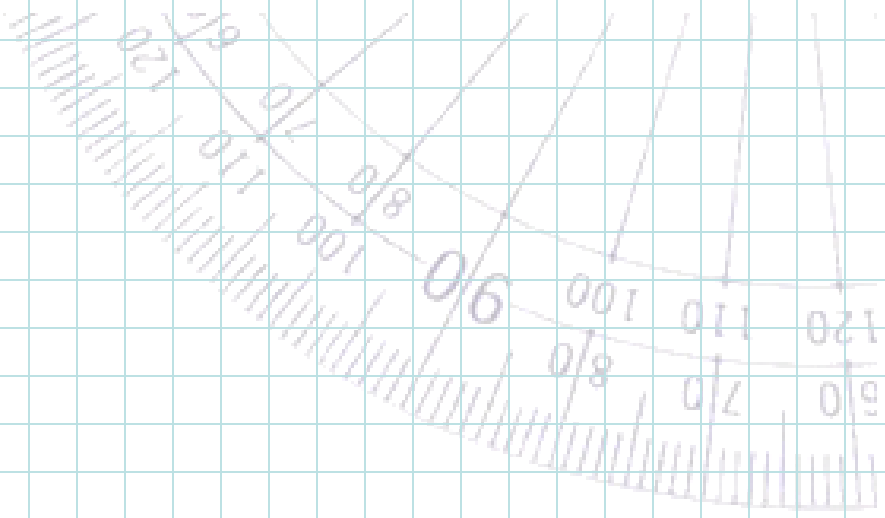
```
function DoIt()
{
// Statements
}
```



- Statements
- Main body of function (or main part)
- “//” represents the start of a comment

# Sample JavaScript Program

```
function DoIt()
{
var number = 5
}
```

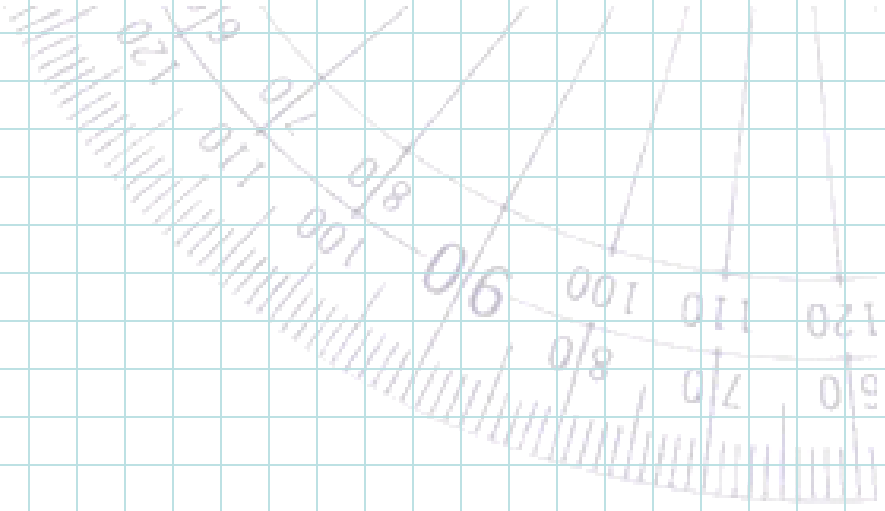


- The identifier `number` is declared as being an integer by its use
- Note: the use of “var” is optional

# Simple JavaScript Program

```
function DoIt()
{
// Statements
}
```

- This program doesn’t do anything!



# Sample JavaScript Program

```
<html>
<head>
<script language="javascript">
<!--
Function DoIt(){
   document.bgColor="#FF0000";
}
//-->
</script>
</head>
<body>
<form>
<input type="button" value="Click"
   onClick="DoIt()">
</form>
</body>
</html>
```

- Here's a short program to try. It is in the file chbackcolor.html
- Try it
- Note: Javascript is case sensitive
- Background is set to red in DoIt() when the button is clicked

# Assignment

- Assignment is an operation that assigns the value of an expression to a variable
- Ex.

  Total = 2 + 3 + 5
- First, the expresssion “2 + 3 + 5” is evaluated
- Then, this value is assigned to the variable “Total”

# Assignment

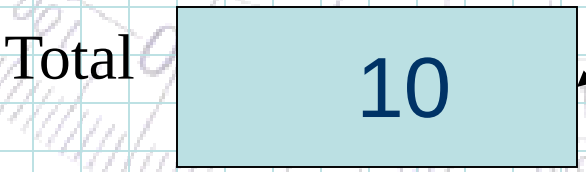
- When a variable is declared, space is allocated in the computer's memory for the variable's value
- Each data type requires a different number of bytes of storage in memory for storing a variable

  int - 2
  float - 4
  double - 8
  char, bool - 1

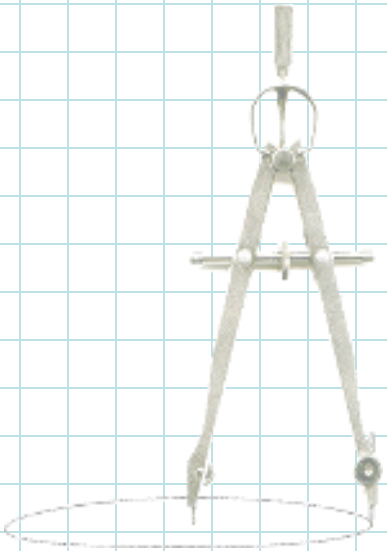
# Assignment

- When a variable is assigned a value, the value is placed into the variable's memory location

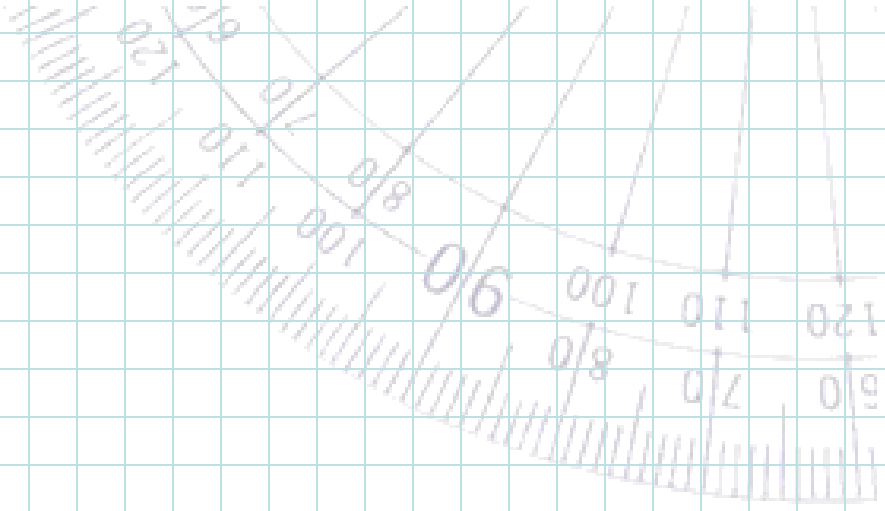
Total = 2 + 3 + 5;



# Arithmetic Operations



- Addition: `2 + 3`
- Subtraction: `5 - 2`
- Multiplication: `10 * 4`
- Division: `12 / 3`



# Order of Operations

- Arithmetic expressions are evaluated according to the following order of operations
- At each level, operations are evaluated left to right
- (1) Parenthesis, Functions
  (2) Multiplication, Division
  (3) Addition, Subtraction

# Parenthesis

- Parenthesis are used to alter the order with which operations are evaluated
- Ex.

  4 + 5 * 2 equals 14  first multiply then add

  (4 + 5) * 2 equals 18  first add then multiply

# Pseudocode

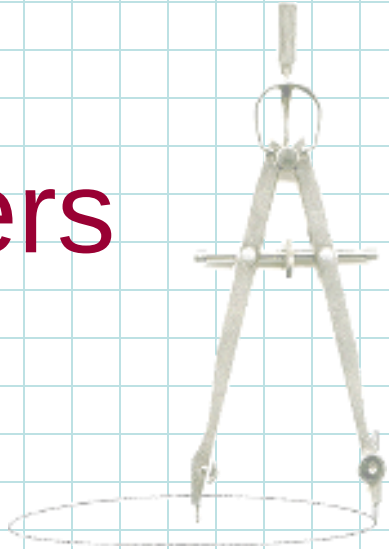
- Language for writing instructions for a computer to follow that is like a programming language, but is informal and not implemented in software. Intended to outline a program as part of development process.

# *Pseudocode*

- Problem: To determine the average of three numbers
- Task: Write step by step instructions to:
  1. Request, from the user, three numbers,
  2. Compute the average of the three numbers,
  3. Print out the original values and the computed average, with text labelling them

Think about this and see the pseudocode on the next slide.

# Pseudocode for the Average of Three Numbers



Get N1, N2, N3

Avg = (N1+N2+N3)/3

Display “Numbers to be averaged.”(literally)

Display N1, N2, N3 (their values)

Display “Average” (literally)

Display Avg (the value of the variable)