

JAVASCRIPT PROGRAMMING

- *Assignment statements*
- *Variables*
- *Functions*
- *Examples*
- *Homework assignment*

ASSIGNMENT STATEMENT

JavaScript equal sign stands for the assignment operator:

left hand side = right hand side

“the value of the right hand side is assigned to the left hand side”

“==” means equal to

“=” means assign to

ASSIGNMENT STATEMENT

```
v = 4
```

```
id = "bird"
```

```
my_color = "red"
```

```
getElementById("bird").style.color = my_color
```

PROGRAMMING VARIABLES

VARIABLES—containers for storing data values. In JavaScript, data can be text, css property values, numbers, and refer to images and filenames, etc.

Local and Global variables

LOCAL VARIABLE

LOCAL variable—a variable declared within a function is only accessible within that function

Input parameters work as local variables within a function

GLOBAL VARIABLE

GLOBAL variable—a variable declared outside of a function and accessible by all scripts and functions within that webpage



JAVASCRIPT FUNCTION

```
function u_name_it(  ){
```

Optional
input parameters



BODY OF FUNCTION

```
    return Optional output;  
}
```

FUNCTION NAME

User defined functions

Unique name

- *case sensitive*
- *cannot start with a number or specialty character*
- *not reserved word*

FUNCTION INPUT

Functions have optional inputs.

- Functions **with no input** have the same behavior regardless of who and where they are invoked.
- Functions **with input** parameters perform the same operations usually with characteristic outcomes.

Parameters are the name of the containers that hold arguments

Arguments are the values passed into a function

FUNCTION BODY

The body of the function contains executable code:

- *declare variables*
- *assignment statements*
- *conditional statements (if, while etc)*
- *return statements*

FUNCTION OUTPUT

Optional output designated by the **return** statement

- Functions with no return perform operations located within the webpage.
- Functions with return return a value that is accessible within the webpage.
- Return value can be a value, variable and expression.

START OFF WITH WHAT WE KNOW!

Equations from algebra: Simple addition, multiplication and finding the slope of a line.

Let's use these to illustrate creating JavaScript functions.

- 1. Describe in words the task or job you want your function to complete.*
- 2. Identify the inputs, operations and outputs for each task*
- 3. Write JavaScript function for each equation*
 - a. Pick a name for the function*
 - b. Pick labels for input parameters*
 - c. Write JavaScript code equivalent for this operation, including return statement.*

ADDITION FUNCTION

Addition

- Addends are the input
- Sum is the output

```
function add_2(a1,a2) {  
return a1+a2;  
}
```

MULTIPLY FUNCTION

Multiplication

- Factors are the input
- Product is the output

```
function mult_2(f1, f2) {  
return f1*f2;  
}
```

SLOPE FUNCTION

Slope of a line

- X's and Y's determine the slope
- 2 points are the input
 - (x_1, y_1) (x_2, y_2)
- Slope is the output
 - Rise/Run

```
function find_slope(x1, y1, x2, y2) {  
return (y2 - y1) / (x2 - x1);  
}
```

ASSIGNMENT

Write functions for the following:

1. Area of a circle with radius, r . Remember $A=\pi r^2$ where $\pi = 3.14$
 - a. Input(1): radius
 - b. Return:Area
2. Volume of a box with length, l , width, w , and height, h . Remember $V=l*w*h$
 - a. Input(3):length,width,height
 - b. Return:Volume
3. Change the background color for the webpage
 - a. Input(1): new color
 - b. Return(0): no return
 - c. Body of function: should update `document.body.style.backgroundColor`
4. Change the font color for any element referenced by an id
 - a. Input(2):id and new color
 - b. Return(0):no return
 - c. Body of function: should update `document.getElementById(id).style.color`