About Functions

Module1 – JS - Handout 7

A function is a block of code that performs a specific task.

Dividing a complex problem into smaller chunks makes your program easy to understand and reusable.

Examples

- a function to find all positive numbers given a list of numbers
- a function to return some of all numbers given a list of numbers

Syntax

```
function nameOfFunction () {

    // function body

}
```

- A function is declared using the function keyword.
- The basic rules of naming a function are similar to naming a variable. It is better to write a descriptive name for your function. For example, if a function is used to add two numbers, you could name the function add or addNumbers.
- The body of function is written within {}.

Example

```
// declaring a function named greet()

function greet() {

    console.log("Hello there");

}
```

Calling a Function

```
// function call

greet();
```

Together –

```
// program to print a text

// declaring a function

function greet() {

    console.log("Hello there!");

}

// calling the function

greet();

//Output

Hello there!
```

Function Parameters

A function can also be declared with parameters. A parameter is a value that is passed when declaring a function.

Note: When a value is passed when declaring a function, it is called parameter. And when the function is called, the value passed is called argument.

Example

```
// program to print the text

// declaring a function

function greet(name) {

   console.log("Hello " + name + ":)");

}

// variable name can be different

let name = prompt("Enter a name: "); // prompt is used to take value from user, when JS is embedded in HTML page


// calling function

greet(name);

//output

Enter a name: Simon

Hello Simon :)
```

Function Return

- The return statement can be used to return the value to a function call.
- The return statement denotes that the function has ended. Any code after return is not executed.
- If nothing is returned, the function returns an undefined value.

Example

```
// program to add two numbers

// declaring a function

function add(a, b) {

   return a + b;

}

// take input from the user

let number1 = parseFloat(prompt("Enter first number: "));

let number2 = parseFloat(prompt("Enter second number: "));

// calling function

let result = add(number1,number2);

// display the result

console.log("The sum is " + result);
```

output:

Enter first number: 3.4

Enter second number: 4

The sum is 7.4

Benefits of Using a Function

- Function makes the code reusable. You can declare it once and use it multiple times.
- Function makes the program easier as each small task is divided into a function.
- Function increases readability.

Function Expressions

In Javascript, functions can also be defined as expressions.

In the below program, variable x is used to store the function. Here the function is treated as an expression. And the function is called using the variable name.

The function below is called an anonymous function.

Example

```
// program to find the square of a number

// function is declared inside the variable

let x = function (num) { return num * num };

console.log(x(4));


// can be used as variable value for other variables

let y = x(3);

console.log(y);
```

Output :

16

9