



FUNCTIONS, METHODS & OBJECTS

- Complex scripts can run to hundreds (even thousands) of lines.
- Programmers use functions, methods, and objects to organize their code.

FUNCTIONS, METHODS & OBJECTS

- **Functions & Methods**
 - Functions consist of a series of statements that have been grouped together because they perform a specific task.
 - A method is the same as a function, except methods are created inside (and are part of) an object.
- **Objects**
 - Programmers use objects to create models using data, and that objects are made up of **properties** and **methods**.
- **Built-in Objects**
 - The browser comes with a set of objects that act like a toolkit for creating interactive web pages.

FUNCTION

- What is a function?

```
FUNCTION KEYWORD  FUNCTION NAME  
function sayHello() {  
  document.write('Hello!');  
}  
CODE BLOCK (IN CURLY BRACES)
```

- Calling a function

```
FUNCTION NAME  
sayHello();
```

FUNCTION

- Declaring function that needs information

```
PARAMETERS  
function getArea(width, height) {  
  return width * height;  
}  
THE PARAMETERS ARE USED LIKE  
VARIABLES WITHIN THE FUNCTION
```

```
getArea(3, 5);
```

FUNCTION

- Getting a single value out of a function

```
function calculateArea(width, height) {  
  var area = width * height;  
  return area;  
}  
var wallOne = calculateArea(3, 5);  
var wallTwo = calculateArea(8, 5);
```

FUNCTION

- Getting multiple values out of a function

```
function getSize(width, height, depth) {  
  var area = width * height;  
  var volume = width * height * depth;  
  var sizes = [area, volume];  
  return sizes;  
}  
var areaOne = getSize(3, 2, 3)[0];  
var volumeOne = getSize(3, 2, 3)[1];
```

ANONYMOUS FUNCTIONS & FUNCTION EXPRESSIONS

- Function Declaration

```
function area(width, height) {  
  return width * height;  
};  
  
var size = area(3, 4);
```

- Function Expression
Anonymous Function

```
var area = function(width, height) {  
  return width * height;  
};  
  
var size = area(3, 4);
```

IIFE

- Immediately Invoked Function Expressions (IIFE)
 - The **final parentheses** (shown on green) tell the interpreter to call the function immediately.
 - The **grouping operators** (shown on pink) are parentheses there to ensure the interpreter treats this as an expression.

```
var area = (function() {  
  var width = 3;  
  var height = 2;  
  return width * height;  
})();
```

IIFE

- When to use anonymous functions and IIFEs?
 - As an argument when a function is called (to calculate a value for that function).
 - To assign the value of a property to an object.
 - In event handlers and listeners to perform a task when an event occurs.
 - To prevent conflicts between two scripts that might use the same variable names.

VARIABLE SCOPE

- Local Variables
- Global Variables

```
function getArea(width, height) {  
  var area = width * height;  
  return area;  
}  
  
var wallSize = getArea(3, 2);  
document.write(wallSize);
```

- LOCAL (OR FUNCTION-LEVEL) SCOPE
- GLOBAL SCOPE

OBJECT

- What is an object?
- Objects group together a set of variables and functions to create a model of a something you would recognize from the real world.



OBJECT

- In an object, **variables** and **functions** take on new names.
- **Variables** become known as **properties**.
 - If a variable is part of an object, it is called a **property**.
- **Functions** become known as **methods**.
 - If a function is part of an object, it is called a **method**.

REAL LIFE OBJECT

Object



Properties

car.name = Fiat
car.model = 500
car.weight = 850kg
car.color = white

Methods

car.start()
car.drive()
car.brake()
car.stop()

OBJECT

- `var person = "John";`
- `var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};`
- Key/value pair

Property	Property Value
firstName	John
lastName	Doe
age	50
eyeColor	blue

OBJECT

- Methods are **actions** that can be performed on objects.
- Methods are stored in **properties** as **function definitions**.

Property	Property Value
firstName	John
lastName	Doe
age	50
eyeColor	blue
fullName	function() { return this.firstName + " " + this.lastName; }

OBJECT

- **Object Definition**

```
var person = {  
  firstName: "John",  
  lastName: "Doe",  
  age: 50,  
  eyeColor: "blue"  
  fullName : function() {  
    return this.firstName + " " + this.lastName;  
  }  
};
```

OBJECT

- Hotel Object:
- Key/value pairs

```
var hotel = {
```

```
  name: 'Quay',  
  rooms: 40,  
  booked: 25,  
  gym: true,  
  roomTypes: ['twin', 'double', 'suite'],
```

```
  checkAvailability: function() {  
    return this.rooms - this.booked;  
  }
```

```
};
```

● KEY
● VALUE

PROPERTIES
These are variables

METHOD
This is a function

OBJECT

- Hotel object contains the following key/value pairs:
 - Variables have a name and can assign them a value of a string, number, or Boolean.
 - Arrays have a name and a group of values. (Each item in an array is a name/value pair because it has an index number and a value.)
 - Named functions have a name and value that is a set of statements to run if the function is called.
 - Objects consist of a set of name/value pairs (but the names are referred to as keys).

PROPERTIES:	KEY	VALUE
	name	string
	rooms	number
	booked	number
	gym	Boolean
	roomTypes	array
METHODS:	checkAvailability	function

