

## JavaScript M5-6 Notes

In the previous lesson, we created an object using the object literal (or initializer) syntax.

```
var person = {  
  name: "John", age: 42, favColor: "green"  
};
```

This allows you to create only a single object.

Sometimes, we need to set an "object type" that can be used to create a number of objects of a single type.

The standard way to create an "object type" is to use an object constructor function.

```
function person(name, age, color) {  
  this.name = name;  
  this.age = age;  
  this.favColor = color;  
}
```

The above function (person) is an object constructor, which takes parameters and assigns them to the object properties.

The this keyword refers to the current object.

Note that this is not a variable. It is a keyword, and its value cannot be changed.

Methods are functions that are stored as object properties.

Use the following syntax to create an object method:

```
methodName = function() { code lines }
```

Access an object method using the following syntax:

```
objectName.methodName()
```

A method is a function, belonging to an object. It can be referenced using the this keyword.

The this keyword is used as a reference to the current object, meaning that you can access the objects properties and methods using it.

Defining methods is done inside the constructor function.

For Example:

```
function person(name, age) {  
  this.name = name;  
  this.age = age;  
  this.changeName = function (name) {  
    this.name = name;  
  }  
}  
  
var p = new person("David", 21);  
p.changeName("John");
```

//Now p.name equals to "John"

You can also define the function outside of the constructor function and associate it with the object.

```
function person(name, age) {  
  this.name= name;  
  this.age = age;  
  this.yearOfBirth = bornYear;  
}  
function bornYear() {  
  return 2016 - this.age;  
}
```

As you can see, we have assigned the object's yearOfBirth property to the bornYear function.

The this keyword is used to access the age property of the object, which is going to call the method.

Note that it's not necessary to write the function's parentheses when assigning it to an object.

Arrays store multiple values in a single variable.

To store three course names, you need three variables.

```
var course1 ="HTML";  
var course2 ="CSS";  
var course3 ="JS";
```

But what if you had 500 courses? The solution is an array.

```
var courses = new Array("HTML", "CSS", "JS");
```

This syntax declares an array named courses, which stores three values, or elements.