8. JavaScript Objects

"Everything" in JavaScript is an Object: a String, a Number, an Array, a Date....

In JavaScript, an object is data, with properties and methods.

8.1. Properties and Methods

Properties are **values** associated with objects.

Methods are **actions** that objects can perform.

8.2. A Real Life Object. A Car:

Properties:

Methods:

car.name=Fiat

car.start()

car.model=500

car.drive()

car.weight=850kg

car.brake()

car.color=white

The **properties** of the car include name, model, weight, color, etc.

All cars have these properties, but the property values differ from car to car.

The **methods** of the car could be start(), drive(), brake(), etc.

All cars have these methods, but they are performed at different times.

8.3. Objects in JavaScript:

In JavaScript, objects are data (variables), with properties and methods.

When you declare a JavaScript variable like this:

```
var txt = "Hello";
```

You actually create a JavaScript String object. The String object has a built-in property called length. For the string above, length has the value 5. The String object also have several built-in methods.

Properties:

txt.length=5

"Hello"

Methods:

txt.indexOf()

txt.replace()

txt.search()

In object oriented languages, properties and methods are often called **object members**.

You will learn more about properties and the methods of the String object in a later chapter of this tutorial.

8.4. Creating JavaScript Objects

Almost "everything" in JavaScript is an object. Strings, Dates, Arrays, Functions.

You can also create your own objects.

This example creates an object called "person", and adds four properties to it:

Example

```
person=new Object();
person.firstname="John";
person.lastname="Doe";
person.age=50;
person.eyecolor="blue";
```

There are many different ways to create new JavaScript objects, and you can also add properties and methods to existing objects.

You will learn much more about this in a later chapter of this tutorial.

8.5. Accessing Object Properties

The syntax for accessing the property of an object is:

```
objectName.propertyName
```

This example uses the length property of the String object to find the length of a string:

```
var message="Hello World!";
var x=message.length;
```

The value of x, after execution of the code above will be:

```
12
```

8.6. Accessing Object Methods

You can call a method with the following syntax:

```
objectName.methodName()
```

This example uses the toUpperCase() method of the String object, to convert a text to uppercase:

```
var message="Hello world!";
var x=message.toUpperCase();
```

The value of x, after execution of the code above will be:

```
HELLO WORLD!
```

8.7. Did You Know?

It is common in object oriented languages, to use camel-case function names.
You will more often see function names like someMethod() instead of some_method().

JavaScript variables are all objects. When you declare a variable you create a new object.