

## 6. JavaScript Variables

JavaScript variables are "containers" for storing information:

### Example

```
var x=5;  
var y=6;  
var z=x+y;
```

### 6.1. Much Like Algebra

```
x=5  
y=6  
z=x+y
```

In algebra we use letters (like x) to hold values (like 5).

From the expression  $z=x+y$  above, we can calculate the value of z to be 11.

In JavaScript these letters are called **variables**.



Think of variables as containers for storing data.

### 6.2. JavaScript Variables

As with algebra, JavaScript variables can be used to hold values ( $x=5$ ) or expressions ( $z=x+y$ ).

Variable can have short names (like x and y) or more descriptive names (age, sum, totalvolume).

- Variable names must begin with a letter
- Variable names can also begin with \$ and \_ (but we will not use it)
- Variable names are case sensitive (y and Y are different variables)



Both JavaScript statements and JavaScript variables are case-sensitive.

### 6.3. JavaScript Data Types

JavaScript variables can also hold other types of data, like text values (person="John Doe").

In JavaScript a text like "John Doe" is called a string.

There are many types of JavaScript variables, but for now, just think of numbers and strings.

When you assign a text value to a variable, put double or single quotes around the value.

When you assign a numeric value to a variable, do not put quotes around the value. If you put quotes around a numeric value, it will be treated as text.

## Example

```
var pi=3.14;  
var person="John Doe";  
var answer='Yes I am!';
```

## 6.4. Declaring (Creating) JavaScript Variables

Creating a variable in JavaScript is most often referred to as "declaring" a variable.

You declare JavaScript variables with the **var** keyword:

```
var carname;
```

After the declaration, the variable is empty (it has no value).

To assign a value to the variable, use the equal sign:

```
carname="Volvo";
```

However, you can also assign a value to the variable when you declare it:

```
var carname="Volvo";
```

In the example below we create a variable called carname, assigns the value "Volvo" to it, and put the value inside the HTML paragraph with id="demo":

## Example

```
<p id="demo"></p>
var carname="Volvo";
document.getElementById("demo").innerHTML=carname;
```



It's a good programming practice to declare all the variables you will need, in one place, at the beginning of your code.

## 6.5. One Statement, Many Variables

You can declare many variables in one statement. Just start the statement with **var** and separate the variables by comma:

```
var lastname="Doe", age=30, job="carpenter";
```

Your declaration can also span multiple lines:

```
var lastname="Doe",
    age=30,
    job="carpenter";
```

## 6.6. Value = undefined

In computer programs, variables are often declared without a value. The value can be something that has to be calculated, or something that will be provided later, like user input. Variable declared without a value will have the value **undefined**.

The variable *carname* will have the value *undefined* after the execution of the following statement:

```
var carname;
```

## 6.7. Re-Declaring JavaScript Variables

If you re-declare a JavaScript variable, it will not lose its value:.

The value of the variable *carname* will still have the value "Volvo" after the execution of the following two statements:

```
var carname="Volvo";
```

```
var carname;
```

## 6.8. JavaScript Arithmetic

As with algebra, you can do arithmetic with JavaScript variables, using operators like = and +:

### Example

```
y=5;  
x=y+2;
```

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