* Scripts can be placed in the <body>, or in the <head> section of an HTML page, or in both.

### The function is invoked (called) when a button is clicked. Example:

<!DOCTYPE html>  
<html>

<head>  
<script>  
function myFunction() {  
    document.getElementById("demo").innerHTML = "Paragraph changed.";  
}  
</script>  
</head>

<body>

<h1>A Web Page</h1>  
<p id="demo">A Paragraph</p>  
<button type="button" onclick="myFunction()">Try it</button>

</body>  
</html>

* JavaScript files have the file extension **.js**.
* JavaScript can "display" data in different ways:
* Writing into an HTML element, using innerHTML.
* Writing into the HTML output using document.write().
* Writing into an alert box, using window.alert().
* Writing into the browser console, using console.log().
* The document.write() method should only be used for testing.
* Using window.alert()
* You can use an alert box to display data.
* In a programming language, these programming instructions are called **statements**.
* A **JavaScript program** is a list of programming **statements**.
* JavaScript statements are composed of:
* Values, Operators, Expressions, Keywords, and Comments.
* Semicolons separate JavaScript statements.
* When separated by semicolons, multiple statements on one line are allowed
* A good practice is to put spaces around operators ( = + - \* / ):
* JavaScript statements can be grouped together in code blocks, inside curly brackets {...}.
* The purpose of code blocks is to define statements to be executed together.
* JavaScript statements often start with a keyword to identify the JavaScript action to be performed.
* Keyword Description
* break Terminates a switch or a loop
* continue Jumps out of a loop and starts at the top
* debugger Stops the execution of JavaScript, and calls (if available) the debugging function
* do ... while Executes a block of statements, and repeats the block, while a condition is true
* for Marks a block of statements to be executed, as long as a condition is true
* function Declares a function
* if ... else Marks a block of statements to be executed, depending on a condition
* return Exits a function
* switch Marks a block of statements to be executed, depending on different cases
* try ... catch Implements error handling to a block of statements
* var Declares a variable

JavaScript syntax is the set of rules, how JavaScript programs are constructed:

var x, y;

JavaScript Values

The JavaScript syntax defines two types of values: Fixed values and variable values.

Fixed values are called literals. Variable values are called variables.

JavaScript Literals

The most important rules for writing fixed values are:

Numbers are written with or without decimals:

Strings are text, written within double or single quotes:

JavaScript Variables

In a programming language, variables are used to store data values.

JavaScript uses the var keyword to declare variables.

An equal sign is used to assign values to variables.

In this example, x is defined as a variable. Then, x is assigned (given) the value 6: