Workshop 1: JavaScript Syntax

Let's first get acquanted with Javascript console and its use.

Save example1.html and open it with a web browser (preferably Google Chrome/Firefox). You see it's a plain HTML-file no magic tricks attached. BUT - as a matter of fact a block of JavaScript is being run every time the page loads, but since it doesn't output anything or modify the page, you can't see it.

Press F12 to open Developer tools and select "Console" to see what JavaScript interpreter is doing. Console is the place where possible error messages or console logging will be printed.

After this you can close the window and see the source code of the page with Ctrl+U. There you can see some JavaScript in the <script> -block. Close the source-window.
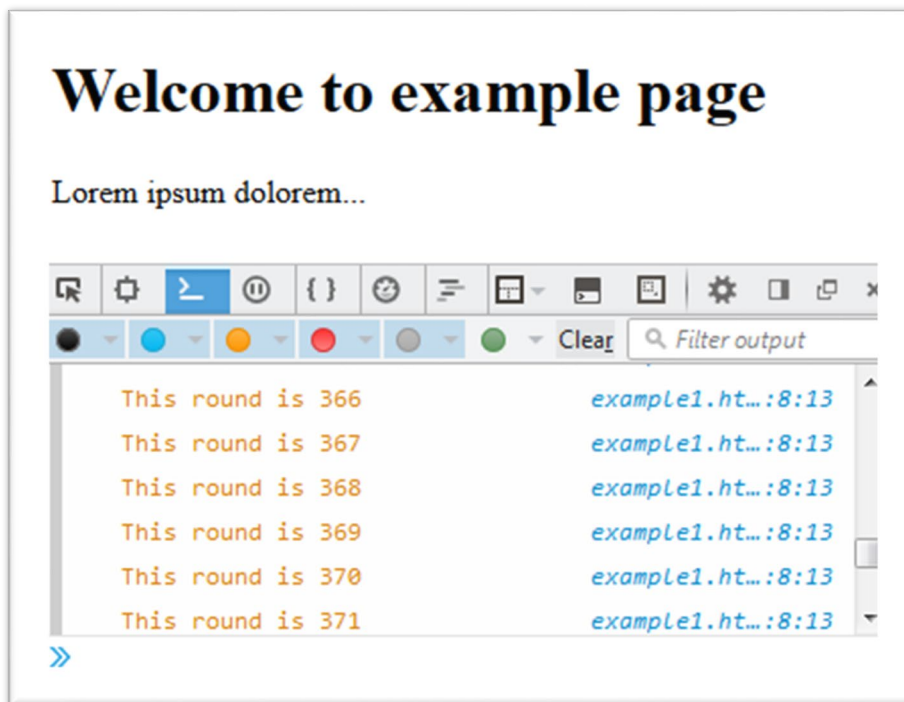
Picture 1: JavaScript console in action

1. JS output and console

Open the JS Console once again (F12). Console is also the place where you can enter and run JavaScript commands to your browser. Lets try it out. You can increase the font size by pressing Ctrl and scrolling the mouse wheel at the same time.

Alert() -function is one of the easiest way to display data to the user. It is also often used when debugging or developing a program in order to see the values of a variable etc.

1. Type in alert(); What happens?

2. Next type in alert("Hello world!"); See? If you wish to use previously entered commands, use arrows keys (up and down) to scroll through console history.

3. Sometimes one has to enter commands which require multiple lines. In these cases, use SHIFT+Enter to change the line without running the command. All the lines will be run consequently.

    Try the following:

    ```
    alert("Hello world!"); // Press Shift+Enter here
    alert("Another hello world"); // Press Shift+Enter here
    alert("Still one more hello world"); // Enter here
    ```

4. Try also typing nonsense to the console. You will see an error message in red.

JS does not actually have any built-in print functions. One can output data in the following ways:

- Writing into an alert box, using window.alert().
- Writing into the HTML output using document.write().
- Writing into the browser console, using console.log().

Try out different variations of your own. Try also adding the code in the SCRIPT-section of the example1.html -page and then loading the page.

2. Defining variables

Lets define some variables and play with them. So get back to the console and clear it if you like. Then type in the following:

```
var firstname = "Your first name";
var lastname = "Your lastname";
alert(firstname);

document.write(lastname);
alert(firstname+" "+lastname);
console.log(lastname+", "+firstname);
```

With console.log -function you can print data to the same Javascript console you've been working on so far. This is very useful if you want to print some debug info without throwing it to the eyes of a regular user.

We will utilize both console.log and alert() -functions heavily during the course, so memorize these if any.

3. Conditionals

If you're done any programming before, conditionals should be fairly familiar to you. It's all about if's, else's and else if's.

In this example you will also familiarize yourself to the *document.write* -function, which is one of the ways to output data from your program.

Try out the following snippet of code in your JS console. Play around with it and see what happens. Remember to use Shift+Enter if you have multiple lines of code you want to enter before running the block.

```
var x = 50;
var order = 'Beer';

if (x > 50) {
    document.write("He's over 50!" );
} else if (x <= 50 && x > 30){
    document.write("Middle aged man, who ordered some "+order);
} else
    document.write("It seems you're bit underaged.");
```

Modify the code so, that it will

* print out HTML-tags to produce formatted output on the page, for example the output could be <h1>He's over 50</h1> or Middle aged man -tags
* have more paths (else if -statements) for different age ranges
* output different messages for each
* output the message to the console as well

4. Loops

Looping is something one has to do when dealing with large amounts of data or when hoping to do an operation multiple times. We will encounter these when we later start utilizing open data available via Web API's.

Try out the following code and play around with it.

```
var arr = [];
for (var i=0; i<15; i++) {
    arr.push( Math.random() );
}
console.log(arr);
```

Modify the code so that it will:

* Output the values generated to the console as they are created
* Output the values using document.write, using HTML-tags to format them
* Randomize numers between 1-100

5. Functions walkthrough

Let's write a small function and run it.

```
// Returns true if age is over 18, otherwise returns false

function areYouOldEnough(age) {
    if (age < 18){
        return false;
    }else
    return true;
}
```

Modify the code so that it will:

- Output true or false on the console before the return statement

Furthermore, we can use the value returned as a parameter to other functions.

```
// Call the function and pass the returned value to the console or to the document

console.log( areYouOldEnough(16) );
document.write( areYouOldEnough(19) );

// Call the value and pass the returned value to alert();

alert(areYouOldEnough(5));
alert(areYouOldEnough(25));
```

Modify the function so, that it will

- output an explanatory text message (not just the value) to the console if the value was NOT over 18.
- output an explanatory text message (not just the value) to the alert box if the value was over 18.

6. Objects

In JS we can build objects and create our own data structures pretty much like in any object-oriented language. Let's try the following approach:

```
// Define an object to hold data for a person
var person = {
    firstName:"John",
    lastName:"Doe",
    age:50,
    eyeColor:"blue"
};
```

console.log(person.age);

Modify the code so, that it will

- have a field for address and telephone.
- have one anonymous method named printInfo, which will print all the information of the person object in the console
- create an array with 4 person objects in it. Then loop through the array and print out the first and lastname for each item

Independent Exercises 1

1. Dogs age

Write a JavaScript function that will return the given man's age as dogs age. It is usually considered that dogs age is man's age multiplied by seven.

2. Debugging a script

Try to run the following piece of code in the console. There are some errors in the code – try to find them? Use Developer Tools to Debug the Code.

```
var i = 5; var j = 10
var k = 25; var l = Joe;

if (i < j) {
 console.log("i");
 else if (j > k) console.log("k");
 }
else console.log("j");
```

```
var arr = [3, 5, 7];
for (var i = 0; i <=3; ++i){
console.log(arr[i]);
}
```