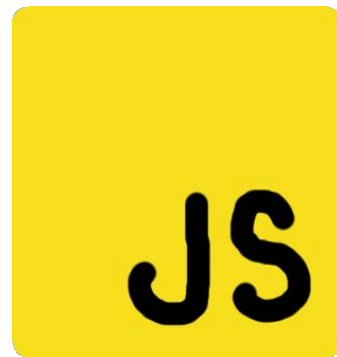


JavaScript



What is JS?

- Interpreted programming language
- The only accepted on client side
- Many features
 - Improves interaction
 - Dynamic requests - AJAX
 - Animations
 - Canvas



JavaScript - Files

Like CSS, it can be imported from a **.js** file

```
<script type="text/javascript" src="main.js"></script>
```

It can be added in the **<script>** tag

```
<script type="text/javascript">  
    window.onload = function () {  
        alert('the game');  
    }  
</script>
```

JavaScript - Files

And it can be coded inside a tag, calling an event like “**onclick**”

```
<button onclick="alert('the game again');">Do NOT click me!!</button>
```

However, this form is **not** recommended

JavaScript - Where

- In the **<head>** so it is ordered and follows logic structure
- Sometimes found inside **<body>** because of loading times
- Loading related events
 - Before loading: the site is not loaded
 - ready: the structure is fully loaded
 - onload: everything is loaded including iframes, photos...

JavaScript - Variables

- Not typed
- Not mandatory to be initialized
- Local / global scope

```
var a, b , c;  
var firstname;  
var nothing = null;  
  
firstname = "Doe";  
a = 1;  
b = 2;  
c = a + b;
```

JavaScript - Variables

Primitive types:

- Undefined: value not assigned
- Null: “no value”. Object.
- Number: any type of number
- Boolean: true or false
- String: any kind of string

```
var num = 1.16;  
var firstname = "Doe";  
var nothing = null;  
var choose = false;  
var idk;  
  
typeof firstname // "string"  
typeof num       // "number"  
typeof nothing   // "null"  
typeof choose    // "boolean"  
typeof idk       // "undefined"
```

JavaScript - Variables

Arrays

```
var bag = ["books", "laptop", 2];  
bag[0] = "big book";
```

Objects

```
var song= {name:"Bohemian Rhapsody",  
           artist:"Queen",  
           album:"A Night at the Opera",  
           fullName: function(){ this.name + " - " + this.artist };  
           duration:5.55 };  
  
alert(song.duration); // 5.55
```


JavaScript - Syntax

Operators	Conditional Statements		Loops
<pre>+ // addition - // subtraction * // multiplication / // division % /* modulus * (remainder) */ ++ // increment -- // decrement</pre>	<pre>if (...) { ... } else if (...) { ... } else { ... }</pre>	<pre>=== // equal valor & type == // equal valor !== // not equal valor & type != // not equal valor > // bigger than >= // bigger or equal than < // smaller than <= // smaller or equal than</pre>	<pre>var i = 0; while (i < 10) { alert("Number " + i); i++; }</pre> <pre>for (var i = 0; i < bag.length; i++) { case += bag[i]; }</pre>

JavaScript - Starting

Different ways to display stuff

```
document.write(2 + 10);
```

```
alert(6 + 6);
```

```
console.log(4 * 3);
```

```
<html>  
  <head>  
    <script>  
      document.getElementById("main").innerHTML = 12 / 1;  
    </script>  
  </head>  
  <body>  
    <div id="main"></div>  
  </body>  
</html>
```

JavaScript - Functions

Named functions

```
function whatTime() {  
    alert("it's time to take a break");  
    return 1;  
}
```

Anonymous functions

```
bagArray.forEach( function() {  
    alert(this);  
});
```

JavaScript - this

- Special keyword that refers to **current** object/function
- Very tricky sometimes

```
var test = {  
  prop: 42,  
  func: function() {  
    return this.prop;  
  },  
};
```

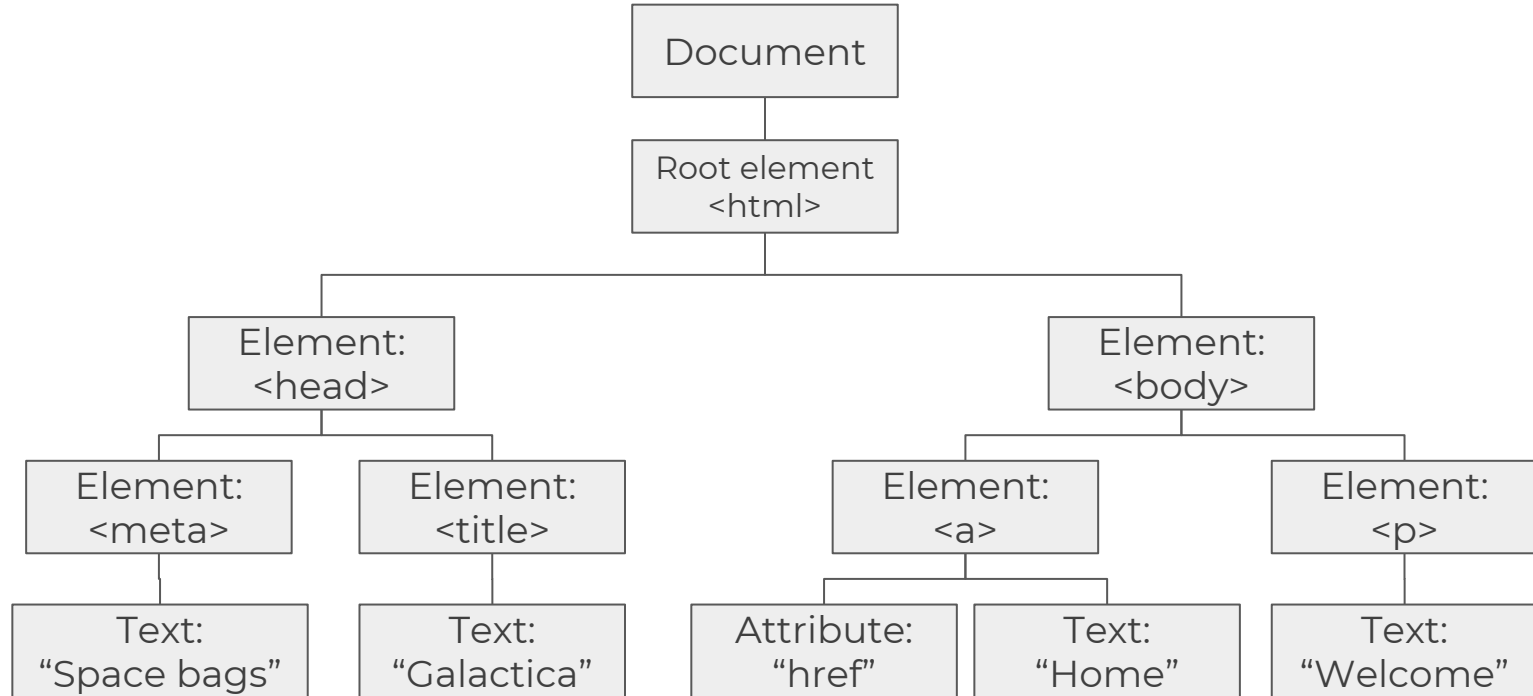
```
document.getElementById("btn").onclick(function () {  
  console.log(this.innerHTML);  
});
```

```
bagMap.forEach( function() {  
  alert(this.numItems);  
});
```

JavaScript - DOM

- Stands for **D**ocument **O**bject **M**odel
- *Platform and language neutral **interface** that allows programs and scripts to **dynamically access and update** the **content, structure** and **style** of a document.*
- JavaScript can, for example:
 - **Add, remove** and **change** all the HTML **elements/attributes**
 - React to existing HTML **events** and create new
 - Change **CSS** styles

JavaScript - DOM



JavaScript - Finding methods

- To get elements by its attributes
 - **getElementById** - first element with specified id
 - **getElementsByClassName** - array of elements by class
 - **getElementsByName** - array of elements by name
- **getElementsByTagName** returns an array of elements by tag
- Other types of methods
 - Adding elements: createElement, appendChild...
 - Deleting elements: deleteChild, replaceChild...

JavaScript - Properties

- To change DOM leaves
 - `element.innerHTML`: to change the HTML
 - `element.attribute`: to change one attribute (like src, href, alt, etc.)
 - `element.setAttribute(attribute, value)`: to add an attribute
 - `element.style.property`: to set a value for a css property like color, font, etc.

```
document.getElementById("guineaPig").innerHTML = "First";  
document.getElementById("guineaPig").name = "second";  
document.getElementById("guineaPig").setAttribute("alt", "Third");  
document.getElementById("guineaPig").style.display = "none";
```


JavaScript - Events

- **onload** - when the site fully loads

```
<!DOCTYPE html>
<html>
  <head>
    <script>
      window.onload = function(){
        alert("loaded!");
      }
    </script>
  </head>
  <body>
  </body>
</html>
```

JavaScript - Events

- **onbeforeunload** - to display a message before leaving the site

```
<!DOCTYPE html>
<html>
  <head>
    <script>
      // Enable navigation prompt
      window.onbeforeunload = function() {
        return true;
      };
      // Remove navigation prompt
      window.onbeforeunload = null;
    </script>
  </head>
  <body>
    <a href="https://www.google.com">
  </body>
</html>
```

JavaScript - Events

- **onchange** -
triggered when
the value changes

```
<!DOCTYPE html>
<html>
  <head>
    <script>
      window.onload = function(){
        document.getElementById("fname").onchange = myFunction; }
      function myFunction() {
        var x = document.getElementById("fname");
        x.value = x.value.toUpperCase();      }
    </script>
  </head>
  <body>
    Enter your name: <input type="text" id="fname">
    <p>When you leave the input field, a function is triggered
      which transforms the input text to upper case.</p>
  </body>
</html>
```

JavaScript - Events

- **onclick** - triggered when the element is clicked (mouse down and up)

```
<!DOCTYPE html>
<html>
  <head>
    <script>
      window.onload = function(){
        var h1 = document.getElementById("h1text");
        h1.onclick = function(){ changeText(h1) };
      }
      function changeText(id) {
        id.innerHTML = "Oops!";
      }
    </script>
  </head>
  <body>
    <h1 id="h1text" >Click on this text!</h1>
  </body>
</html>
```

JavaScript - Events

- **onmouseover** - the mouse is moved over an HTML element
- **onmouseout** - the mouse is moved away an HTML element
- **onkeydown** - a keyboard key is pushed
- **onkeypress** - a key is pressed down and up
- **onfocus** - an element gets focus
- **onpagehide** - similar to onunload