LAUREA
UNIVERSITY OF APPLIED SCIENCES
Together we are stronger

JavaScript BOM

Dynamic Web Apps - 2017

Mika Stenberg / Meija Lohiniva

1. What will be discussed today

1. Variables: Data Types
2. Variables: scope
3. Functions
4. Arrays
5. The BOM– Browser Object Model
6. THE DOM – Document Object Model
7. Using The BOM
8. BOM: Navigator object
9. BOM: Window object
10. BOM: History object
11. BOM: Location object

What did we learn last time?

Scopes

- JavaScript has two scopes:
  - Global Scope
    - Visible and accessible everywhere.
  - Local Scope
    - Visible and accessible within the function.

1. Variables: Data Types

- JavaScript variables are DYNAMIC -> a variable can store different data types.

```
var myVariable;            // datatype: undefined
var myVariable = 2;        // datatype: number
var myVariable = ”2”;      // datatype: String
```

- Operations on variables:
  - Adding a number to a string: JS treats number as a string.
  - JS evaluates expressions from left to right.

```
var myVariable = ”A string” + 2 + 3; // A string23
var myVariable = 2 + 3 + ”A string”; //5A string
```

2. Variables: scope

Name	Scope	Note
	global	*Not advicable*
var	function	
let	block	ES6
const	block	*Cannot be changed later on*

```
function myFunction() {
  if (true) {
    var tmp = 123;
  }
  console.log(tmp); // 123
}
```

```
function myFunction() {
  if (true) {
    let tmp = 123;
  }
  console.log(tmp); // err: undefined
```

```
const foo; // err: missing = in const declaration

const bar = 123;
bar = 456; // err: `bar` is read-only
```

3. Functions

- Functions are defined with keyword `function`.
- Compared to defining functions in Java:
  - No need to define scope.
  - No need to define return value's datatype.
- To call a function you just have the name of the function with parenthesis.

```
function myFunction() {
       //code here
}
myFunction();
```

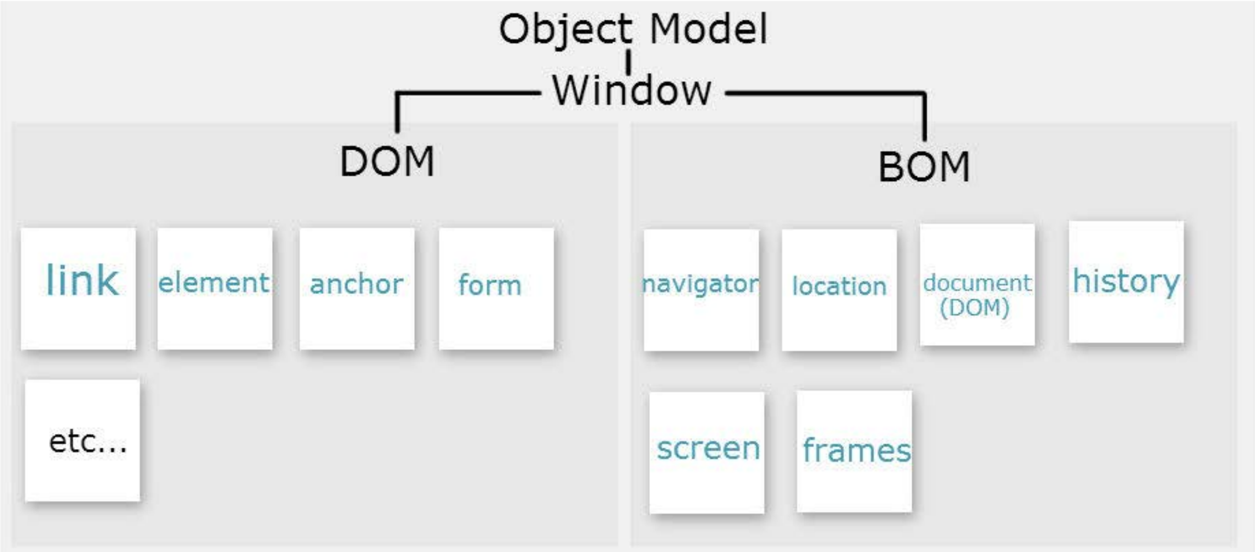
4. Arrays

- Pretty much the same as in other languages:
  - Indexing starts from 0.
  - Items in square brackets ([]), separated by commas.

```
var myArray = [
       "item1",
       "item2",
       "item3"
];
```

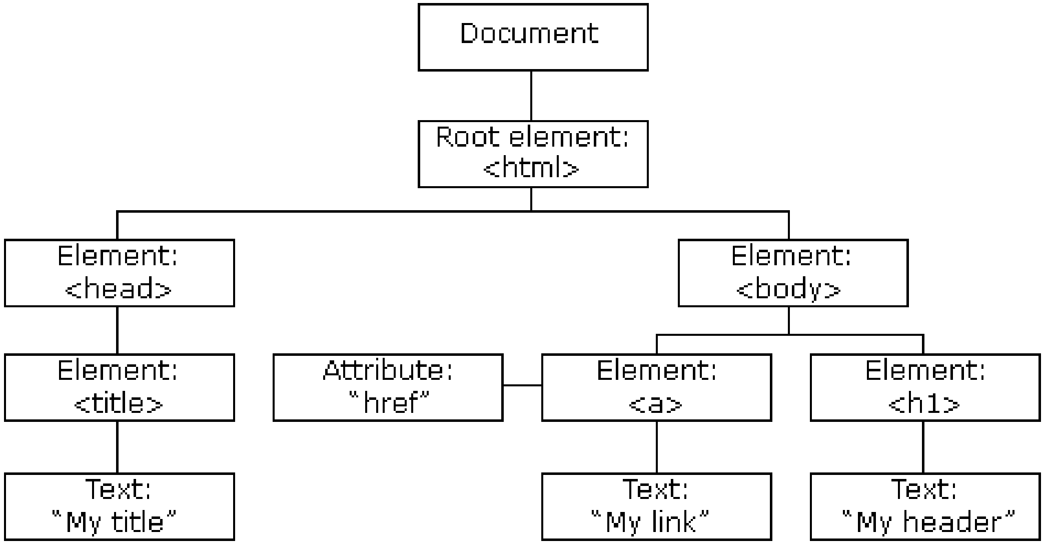
5. The BOM– Browser Object Model

- Methods and properties for JavaScript interactivity
- Modern browsers have implemented (almost) the same methods and properties.



6. THE DOM – Document Object Model

- When a web page is loaded, the browser creates a **D**ocument **O**bject **M**odel of the page.
- The **HTML DOM** model is constructed as a tree of **Objects**:



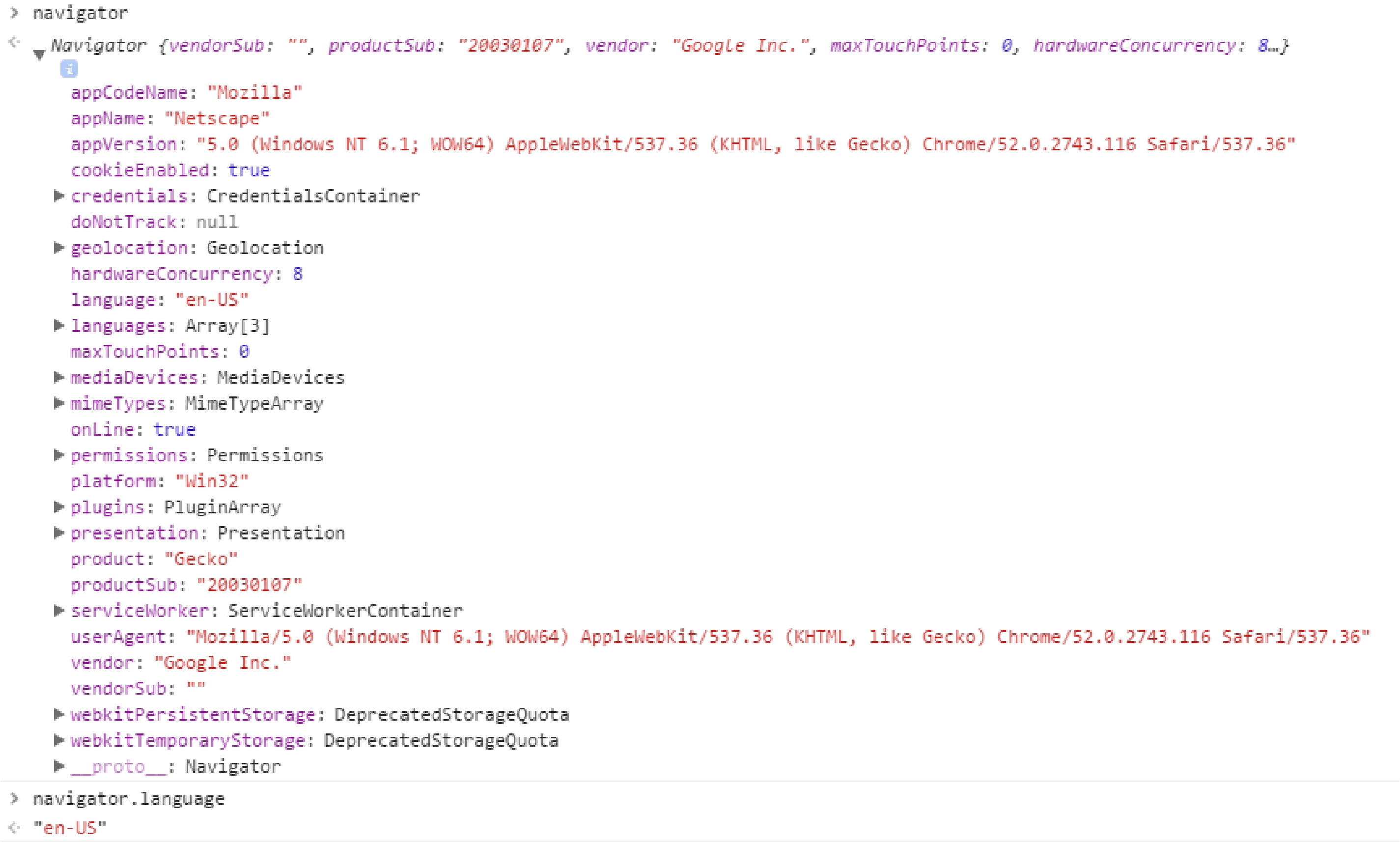
7. Using The BOM

- Allows JS to talk to the browser and get information about:
  - Browser Window contents (DOM)
  - Frames shown in page
  - Screen size, orientation, color depth
  - Navigator: browser specific information
  - History: web site history
  - Location: current web page information

(Picture: http://www.bitmanagement.com/press/cross-browser-installation)

8. BOM: Navigator object

- For example, we can access information about the browser:



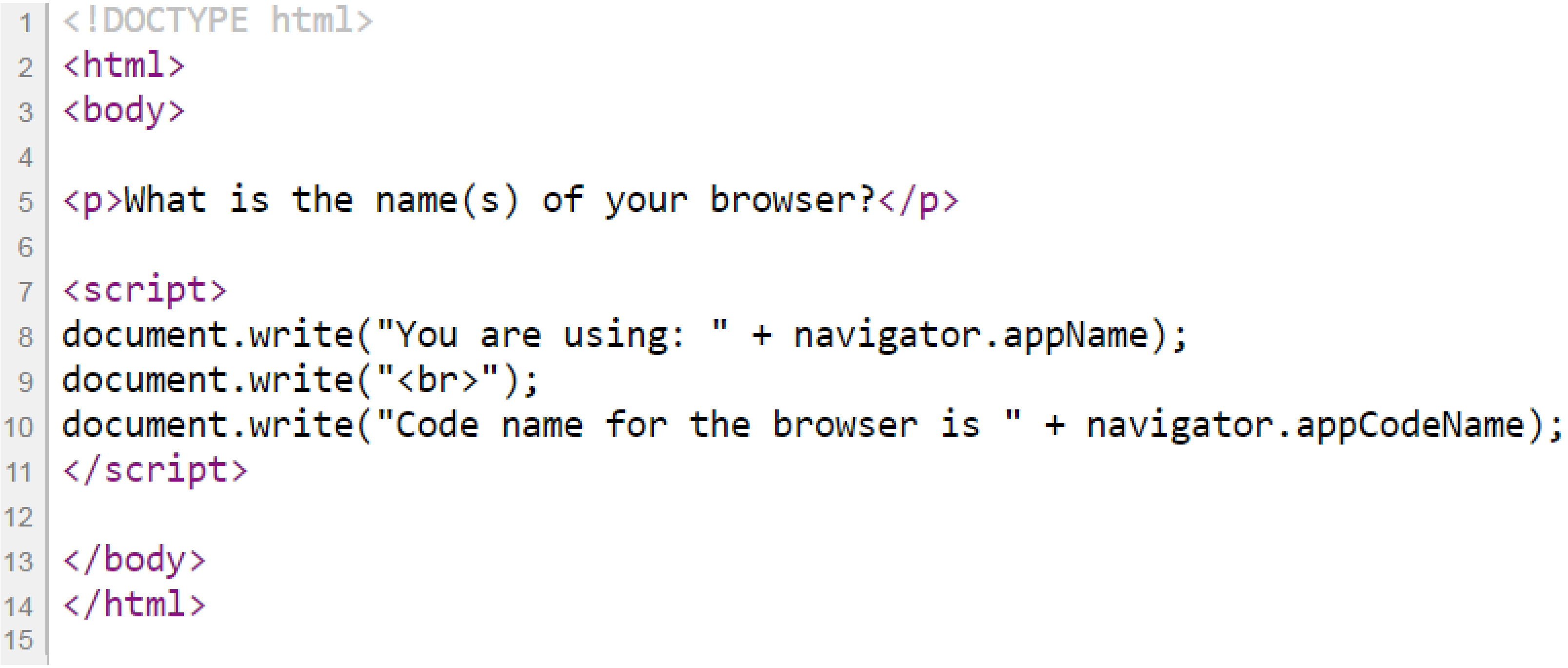
```
> navigator
< ▼Navigator {vendorSub: "", productSub: "20030107", vendor: "Google Inc.", maxTouchPoints: 0, hardwareConcurrency: 8…}
      appCodeName: "Mozilla"
      appName: "Netscape"
      appVersion: "5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
      cookieEnabled: true
    ▶ credentials: CredentialsContainer
      doNotTrack: null
    ▶ geolocation: Geolocation
      hardwareConcurrency: 8
      language: "en-US"
    ▶ languages: Array[3]
      maxTouchPoints: 0
    ▶ mediaDevices: MediaDevices
    ▶ mimeTypes: MimeTypeArray
      onLine: true
    ▶ permissions: Permissions
      platform: "Win32"
    ▶ plugins: PluginArray
    ▶ presentation: Presentation
      product: "Gecko"
      productSub: "20030107"
    ▶ serviceWorker: ServiceWorkerContainer
      userAgent: "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/52.0.2743.116 Safari/537.36"
      vendor: "Google Inc."
      vendorSub: ""
    ▶ webkitPersistentStorage: DeprecatedStorageQuota
    ▶ webkitTemporaryStorage: DeprecatedStorageQuota
    ▶ __proto__: Navigator
> navigator.language
< "en-US"
```

8. BOM: Navigator object

- Developer tools (Ctrl+Shift+I)
- Console on the bottom pane
    - Type navigator
    - Observe all the data that is available there

8. BOM: Navigator object

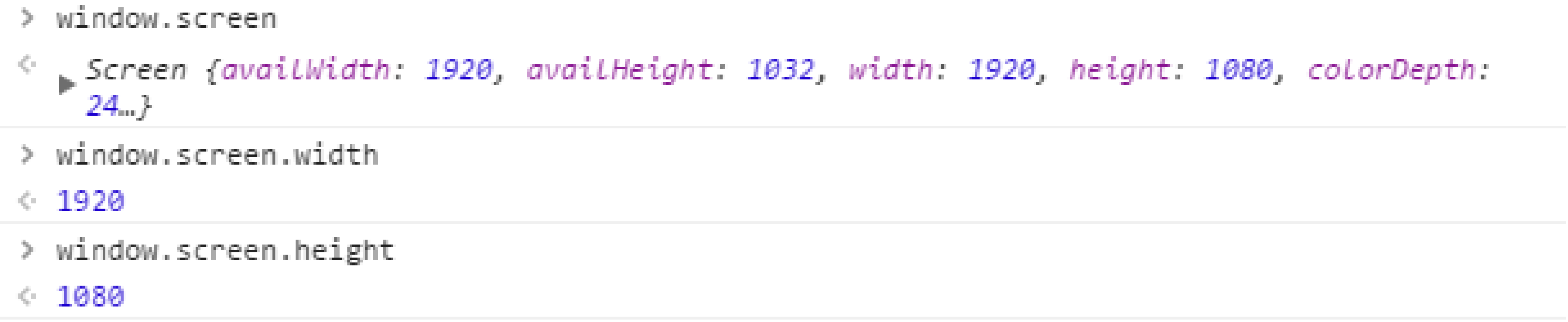
- We can add this to the <script>-tag in the HTML page.
- Code is run when the page is loaded



```
<!DOCTYPE html>
<html>
<body>

<p>What is the name(s) of your browser?</p>

<script>
document.write("You are using: " + navigator.appName);
document.write("<br>");
document.write("Code name for the browser is " + navigator.appCodeName);
</script>

</body>
</html>

```

9. BOM: Window object

- Window-object lets us query the information about screen properties, such as width and height.



```
> window.screen
<· ▶ Screen {availWidth: 1920, availHeight: 1032, width: 1920, height: 1080, colorDepth:
    24…}
> window.screen.width
<· 1920
> window.screen.height
<· 1080
```

10. BOM: History object

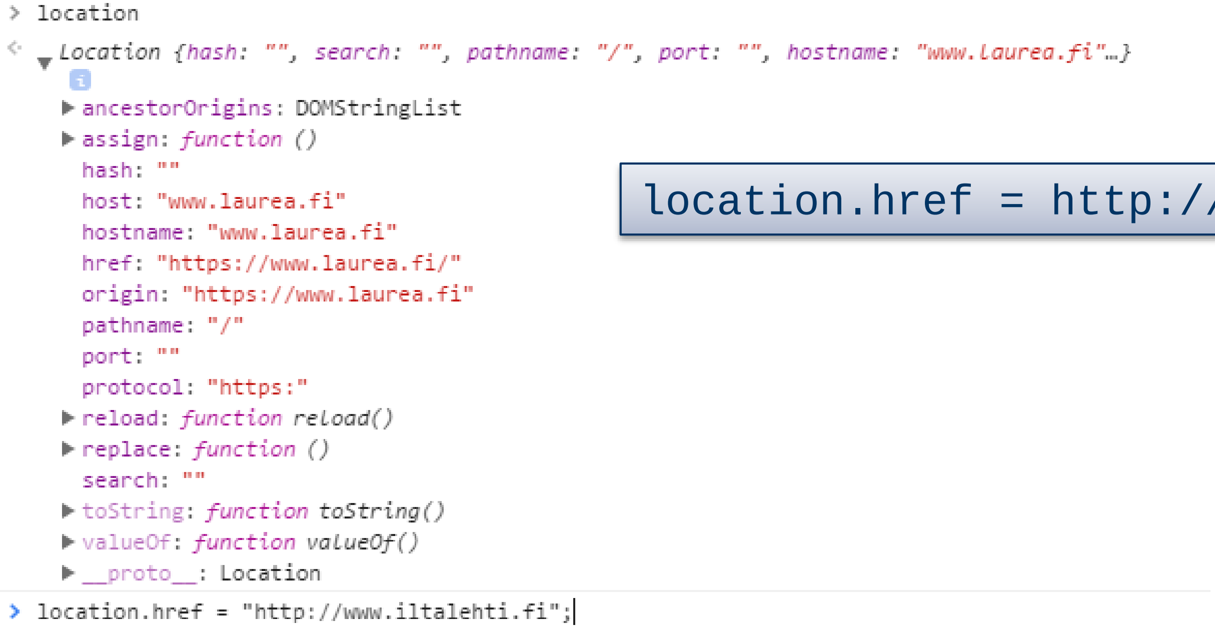
- History-object lets us query the information about browser history.
- We can also control the browser by telling it to go back or forward in history.

```
<script>
  history.back();
</script>
```

NOTE: *History is protected by the browser; JavaScript is not allowed to read the contents of it.*

11. BOM: Location object

- Location-object lets us query the information about current location.
- We can also set the location which causes the browser to load it:



```
> location
<· ▼Location {hash: "", search: "", pathname: "/", port: "", hostname: "www.laurea.fi"…}
    i
   ▶ancestorOrigins: DOMStringList
   ▶assign: function ()
    hash: ""
    host: "www.laurea.fi"
    hostname: "www.laurea.fi"
    href: "https://www.laurea.fi/"
    origin: "https://www.laurea.fi"
    pathname: "/"
    port: ""
    protocol: "https:"
   ▶reload: function reload()
   ▶replace: function ()
    search: ""
   ▶toString: function toString()
   ▶valueOf: function valueOf()
   ▶__proto__: Location
> location.href = "http://www.iltalehti.fi";
```

```
location.href = http://www.iltalehti.fi;
```

Questions or comments?

LAUREA
UNIVERSITY OF APPLIED SCIENCES
Together we are stronger

JavaScript