Geo-application Development

JavaScript

Client-side Scripting for Dynamic Websites

Bernd Resch

PARIS LODRON UNIVERSITY SALZBURG

Z GIS

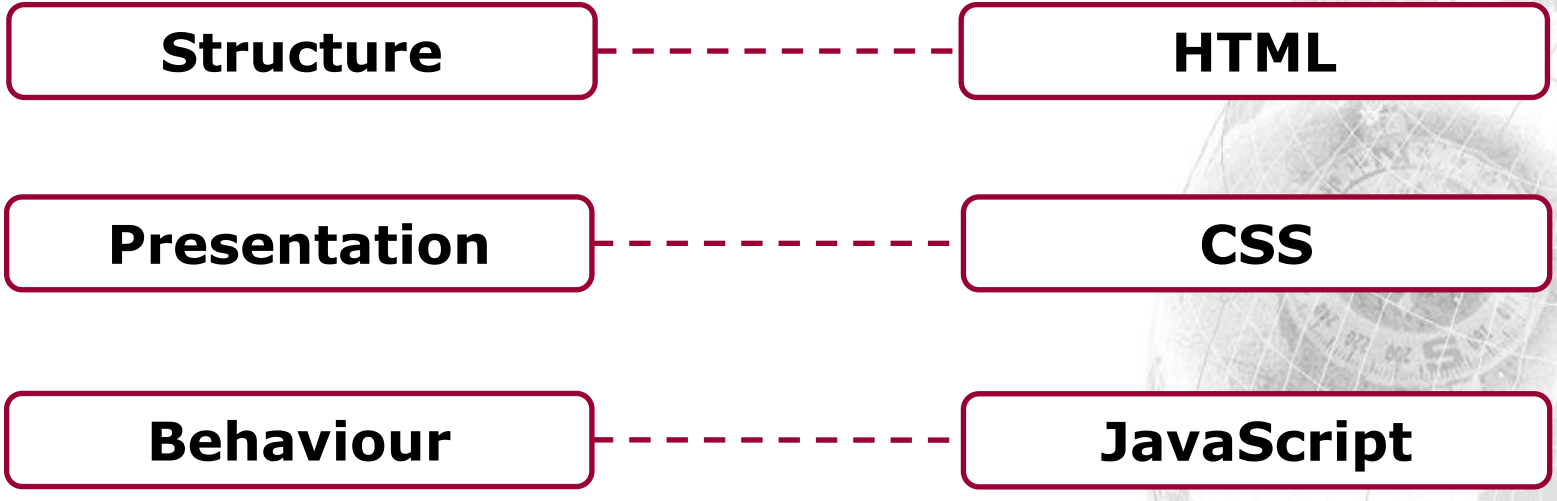
Overview

- Basic Principles
- Integration in HTML
- Functions
- DOM
- OpenLayers

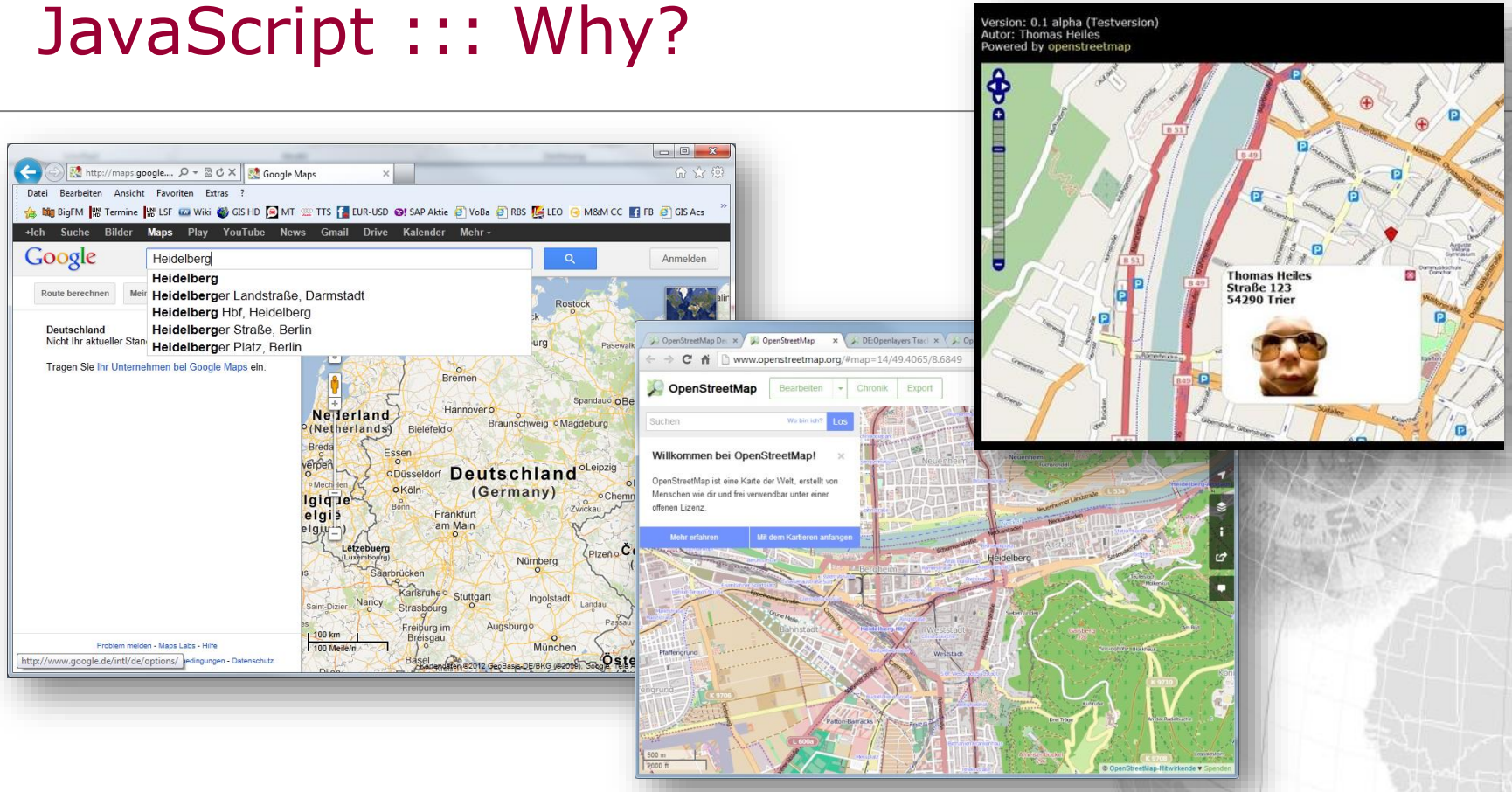
Recap of Assignment 1

- Relative vs. absolute div height/width values
  - Or a mixture? (header/footer vs. body)
- Relative vs. absolute font size values
- Font + div interaction!

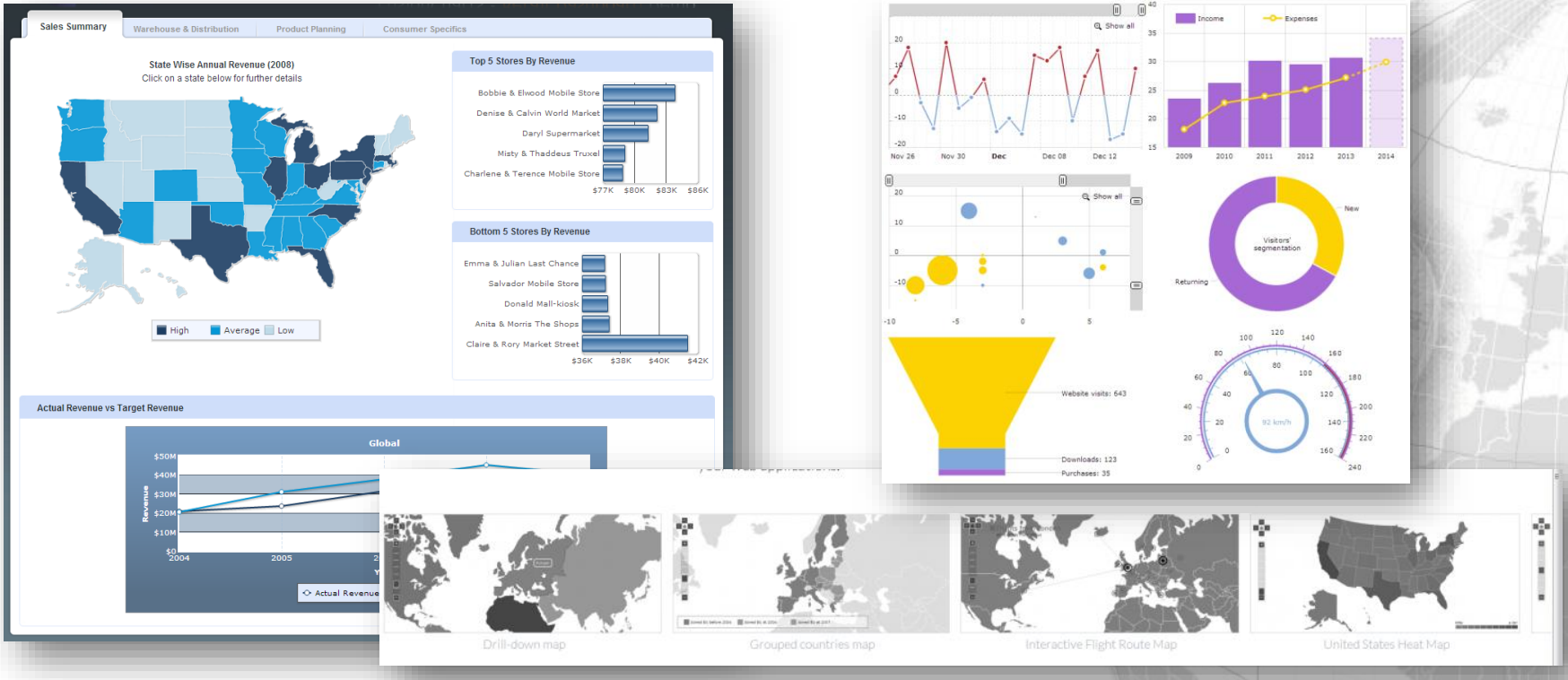
JavaScript ::: Why?



JavaScript ::: Why?



JavaScript ::: Why?



Basic Principles

JavaScript ::: Variables

Keyword var

name of the variable

```
var city;
var street;
var postcode;

city = „Salzburg";
street = „Schillerstraße";
postcode = 5020;
```

setting the variables

Variables – Data Types ::: Basics

```
var a;                  // undefined
var a = 5.5;            // Number
var a = "John";         // String
```

Take care!

```
var city = "Salzburg"
var postCode = "5020";
var street = "Schillerstraße;
var houseNr = 30;
```

JavaScript ::: Text Output

```
document.write("My output");

var text = "This is my output text";
document.write(text);

document.write("My output is: " + text);
```

JavaScript ::: Text Output

```
var text1 = "My output is: ";
var text2 = "Hello World!";

document.write(text1 + text2);
```

JavaScript ::: Conditional Statements

```
var x = 10;
var y = 20;
var max;

if (x > y) {
    max = x;
}
else {
   max = y;
}
```

JavaScript ::: Comments

```
// a comment for one line

/* a comment over
several lines */

/* a very long comment is also
possible. Documentation is always
very important – especially for
your assignments  :-) */
```

JavaScript ::: Functions and Events

```
<head>
    ...
    <script type="text/javascript">
        function zoomIn(){

            // some code
        }
    </script>
    ...
</head>
<body>
    ...
    <a href="#" onclick="zoomIn();">
       <img src="zoom-in.gif" alt="Zoom In" />
    </a>
    ...
</body>
```

Integration in HTML

Integration ::: In HTML

```
<script type="text/javascript">
    var a = 10;
    var b = 7;

    var c = a + b;

    document.write(c);   // Comment: c is 17
</script>
```

Integration ::: External File (*.js)

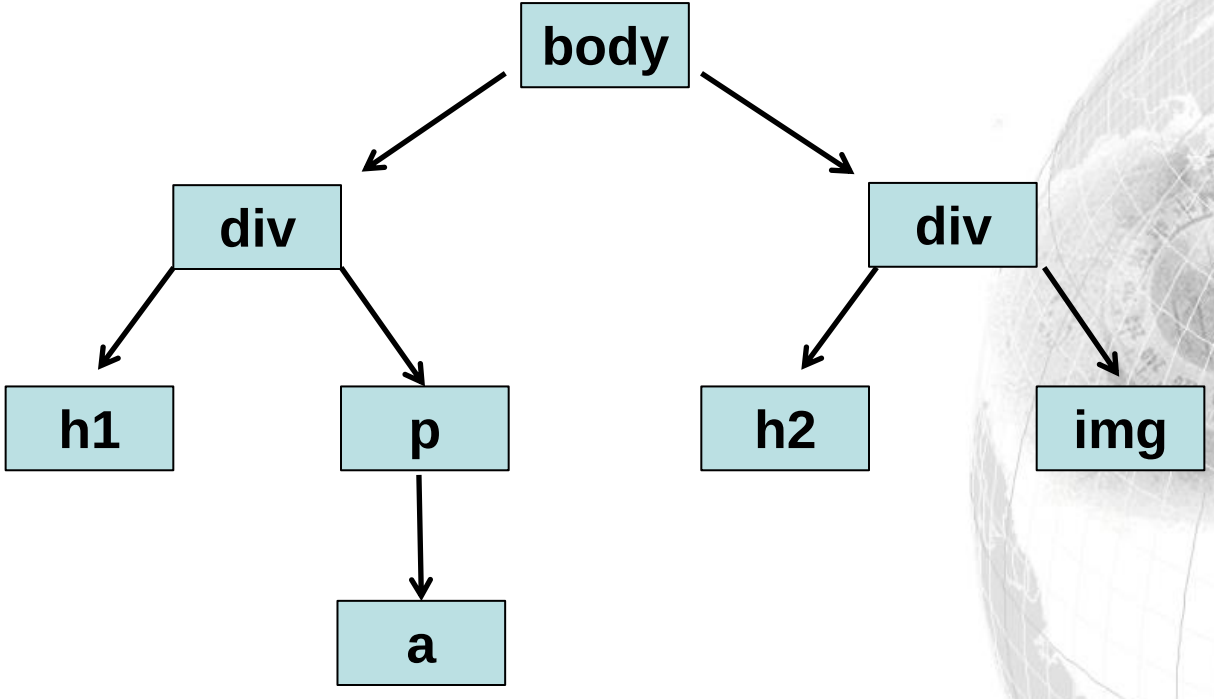
```
<script type="text/javascript" src="script.js"></script>
```

- Functionality in the external file can be used as if it were defined in the same file

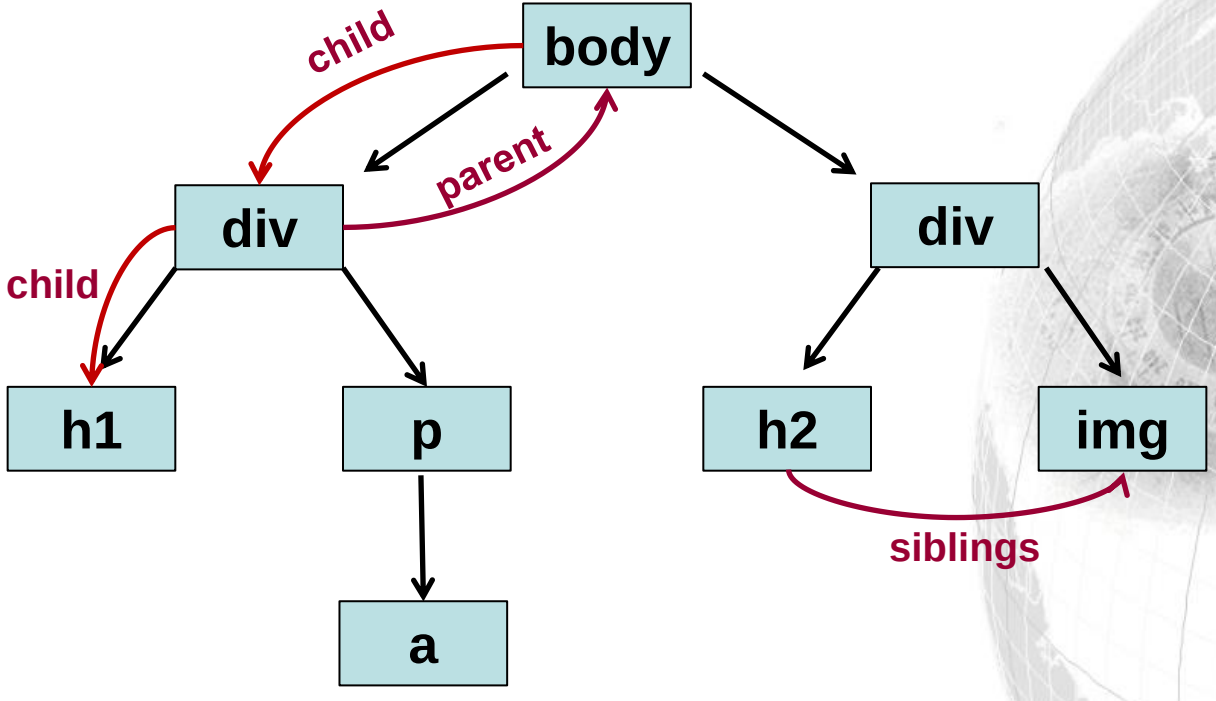
JavaScript ::: DOM Example

```
<body>
  <div id="intro">
    <h1>An OpenStreetMap  Map</h1>
    <p>OpenStreetMap is a Web 2.0 project, aiming at collecting
       free geodata (e.g., Open Data).
       (Source: <a href="http://de.wikipedia.org/wiki/
        OpenStreetMap">Wikipedia</a>).</p>
  </div>
  <div id="map">
    <h2>Map</h2>
    <img src="map.png" id="mapImage" class="maps" alt="OSM Map Salzburg">
  </div>
</body>
```

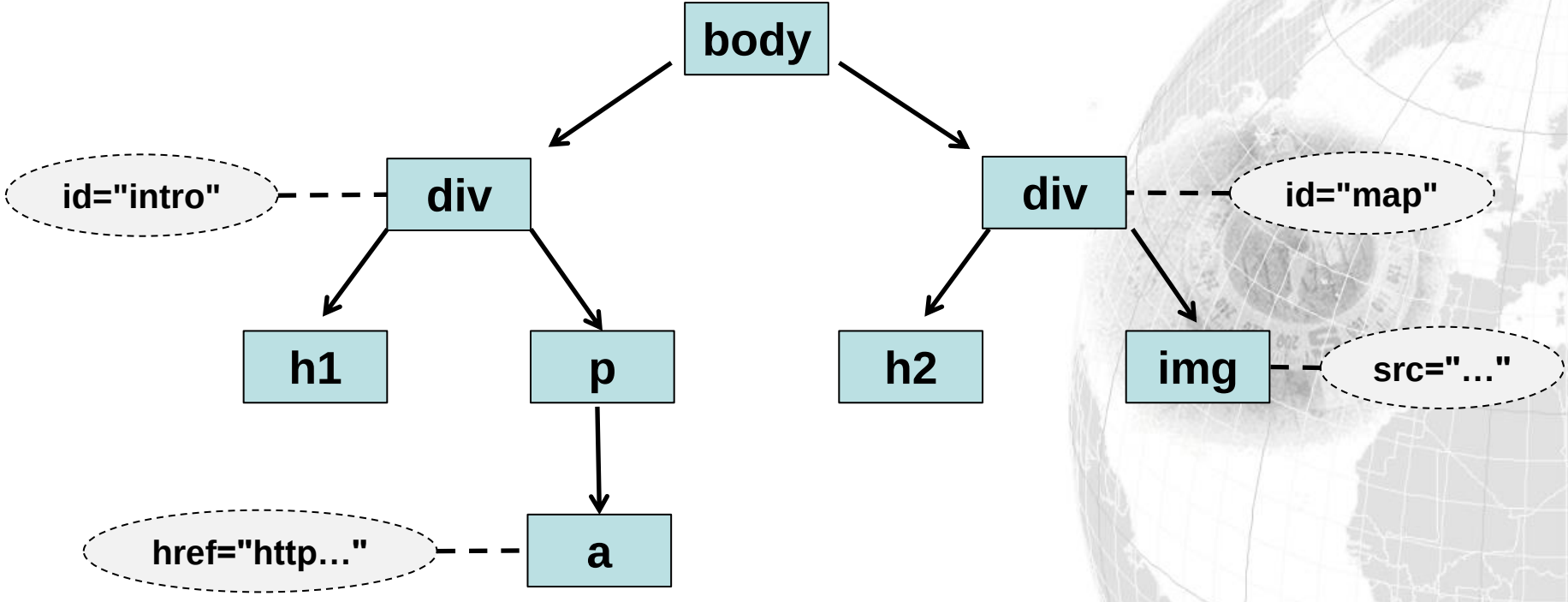
DOM ::: Nodes - Hierarchy



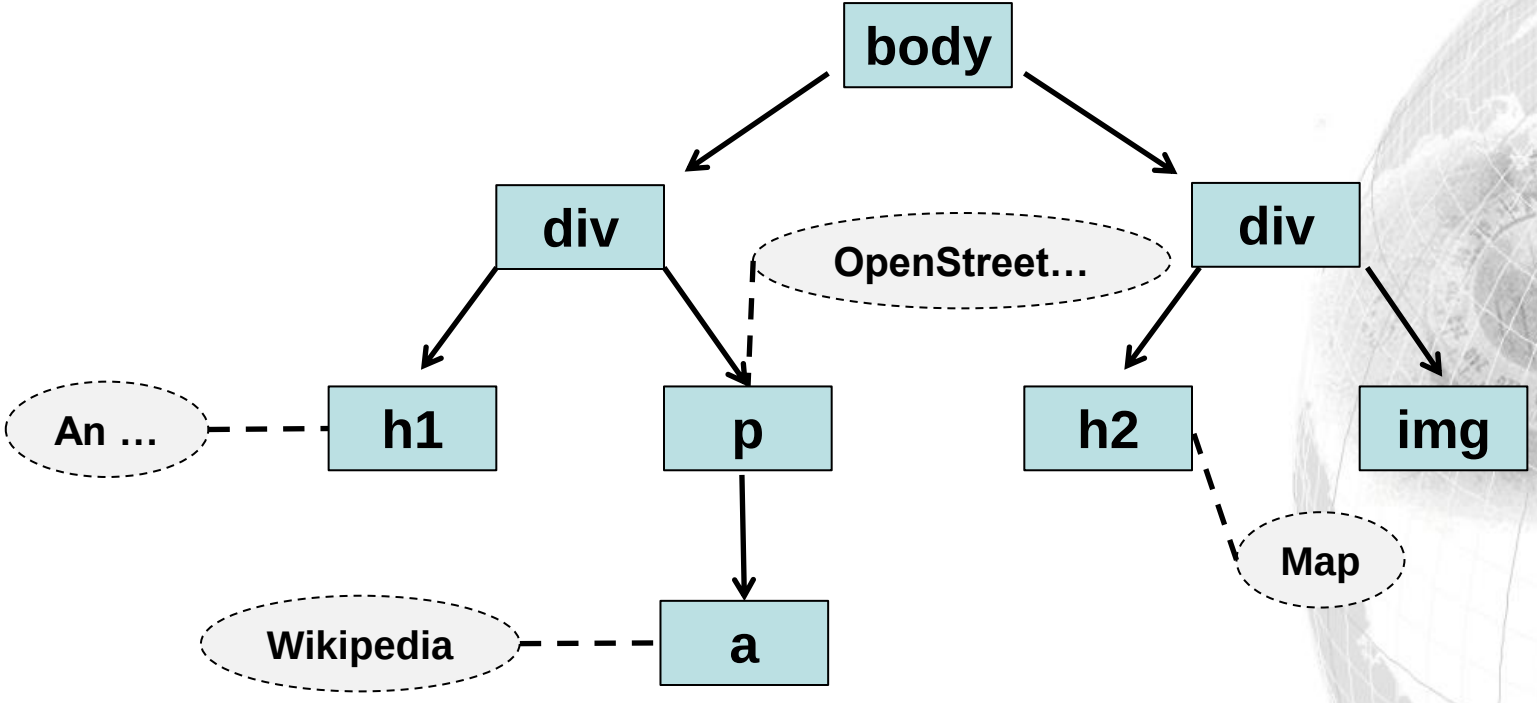
DOM – Nodes ::: Childs, Parents and Siblings



DOM – Nodes ::: Attributes



DOM – Nodes ::: Text



JavaScript ::: DOM Example

```
<body>
  <div id="intro">
    <h1>An OpenStreetMap Map</h1>
    <p>OpenStreetMap is a Web 2.0 project, aiming at collecting
       free geodata (e.g., Open Data).
       (Source: <a href="http://de.wikipedia.org/wiki/
        OpenStreetMap">Wikipedia</a>).</p>
  </div>
  <div id="map">
    <h2>Map</h2>
    <img src="map.png" id="mapImage" class="maps" alt="OSM Map Salzburg">
  </div>
</body>
```

DOM ::: Get Elements and Attributes

```
var imgElement = document.getElementById("mapImage");

var imgSrc = imgElement.getAttribute("src");
// imgSrc now contains „map.png“

var imgClass = imgElement.getAttribute("class");
// imgClass now contains maps

→ if (imgClass == "maps") { … }
```

DOM ::: Set Attributes of an Element

```
var imgElement =
        document.getElementById("mapImage");

// use another map image
imgElement.setAttribute("src", "map2.png");
imgElement.setAttribute("class", "myredmap");
```

DOM ::: Set Attributes of an Element

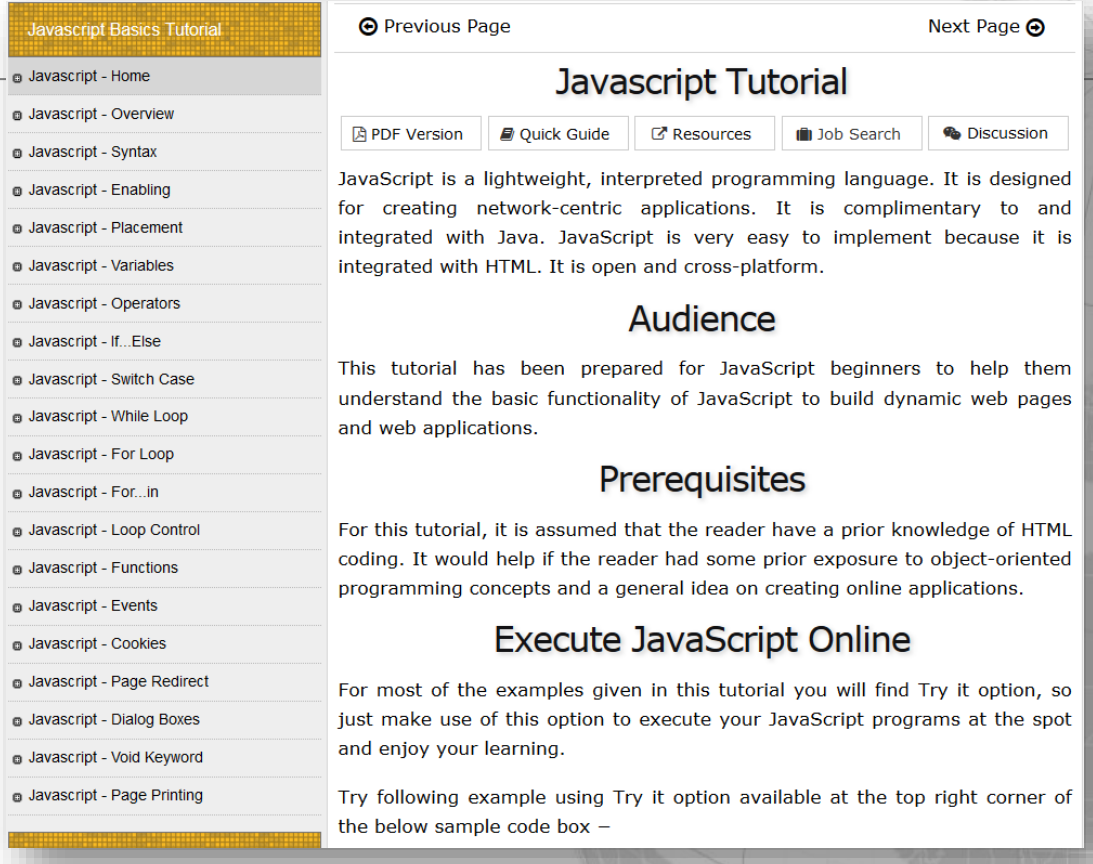
```
var divElement = document.getElementById(„intro");
divElement.style.visibility = "hidden";
// changes visibility of intro (div) to hidden

var divElement2 = document.getElementById("map");
divElement2.style.visibility = "visible";
// changes visibility of map to visible
```

Useful Links

- http://www.w3.org/DOM/
- http://de.selfhtml.org/dhtml/modelle/dom.htm
- **http://www.w3schools.com/js/js_htmldom.asp**
- Online JavaScript Tester
  http://www.webtoolkitonline.com/javascript-tester.html

Home Study

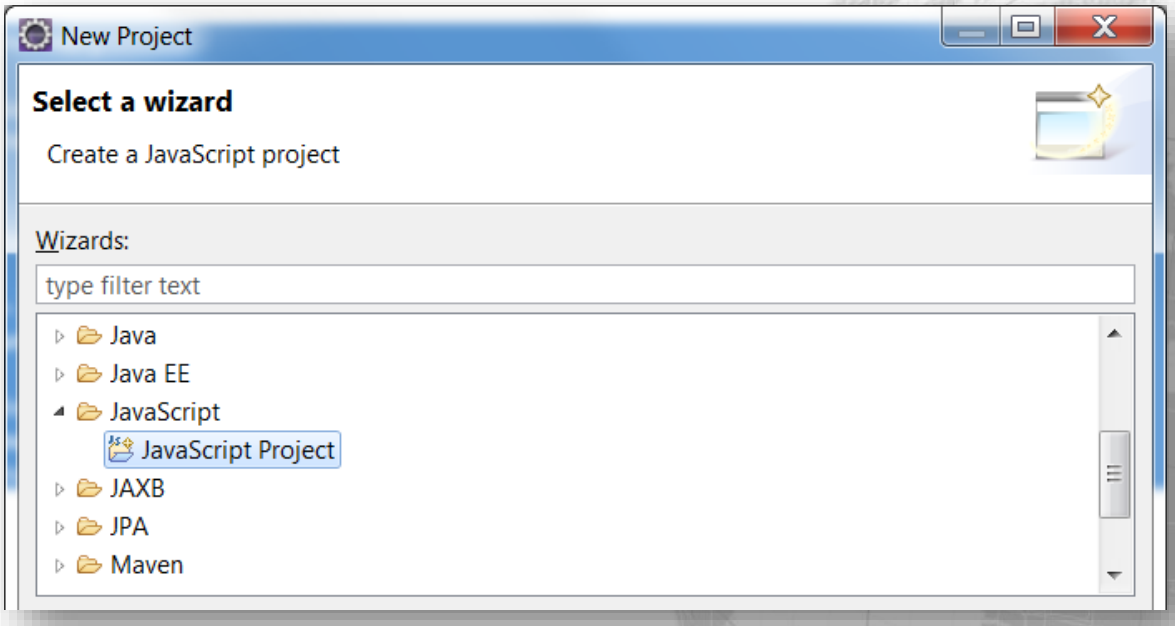
- www.tutorialspoint.com/javascript
  - Go through the entire tutorial (JavaScript Basics: until "Events")



Exercise: Mobile Web Map

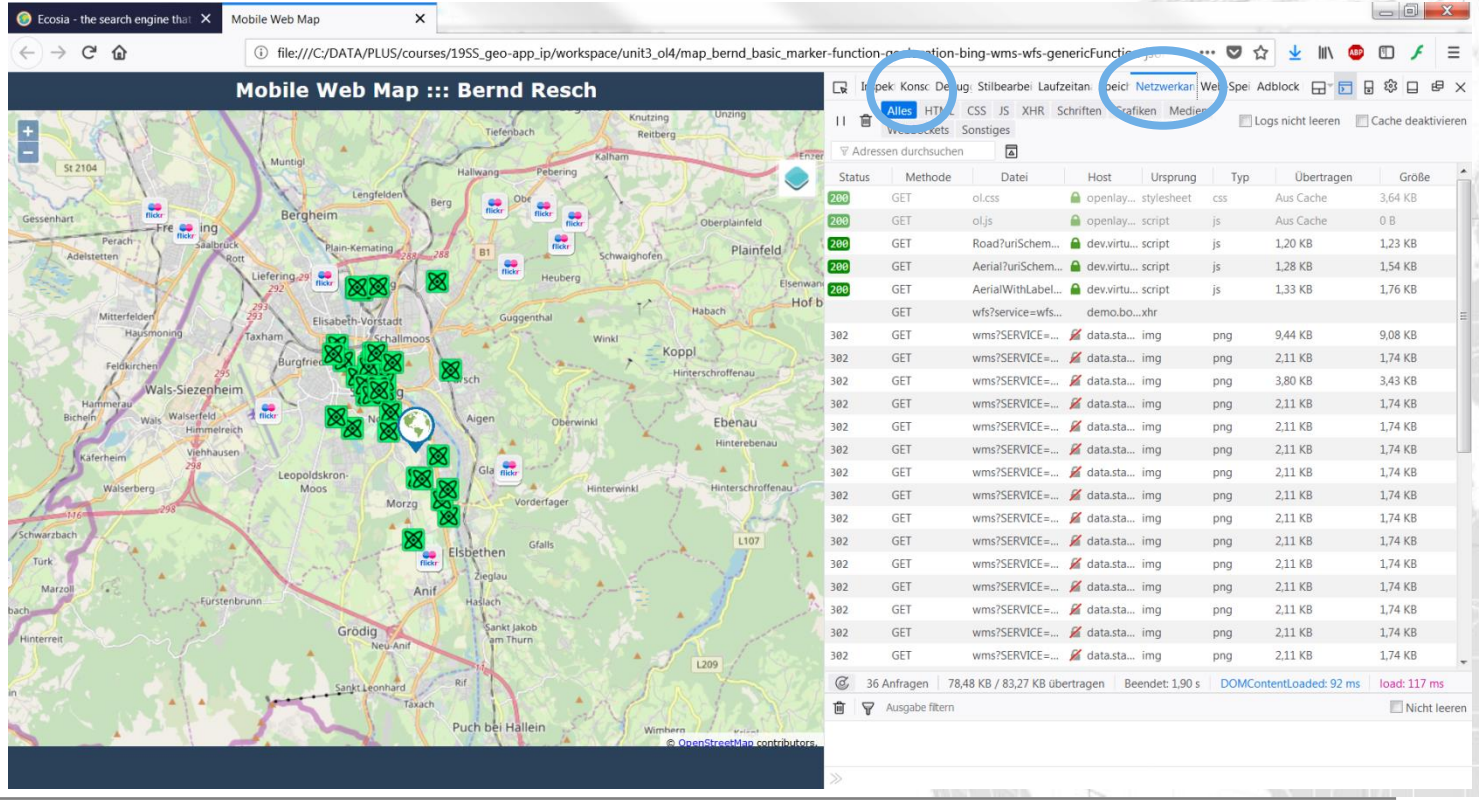
Exercise: Mobile Web Map

- Create a new JavaScript project



JavaScript ::: Debugging!

- F12 in browser



Exercise: Mobile Web Map

- Use the skeleton file you developed earlier:
  - Make a copy
  - Rename the copy to *map_<name>.html*

Exercise: Mobile Web Map

- Extend your file *map_<name>.html* by a simple OpenLayers map: http://openlayers.org/en/latest/apidoc
  - Hint: do not edit many parts in the code at once
    - Add a small part and test it
    - ➔ Otherwise, debugging will be difficult

Exercise: Mobile Web Map

- Use this quickstart guide:
  https://openlayers.org/doc/quickstart.html
- Code to be found here
  https://openlayers.org/download

➔ **Please use API version 5.3.0**
→ (no modules!) and JS tags

Exercise: Mobile Web Map

➔ **Please use exactly this version!**

- <link rel="stylesheet" href="https://openlayers.org/en/v5.3.0/css/ol.css" type="text/css">
- <script src="https://openlayers.org/en/v5.3.0/build/ol.js" type="text/javascript"></script>

Exercise: Mobile Web Map

- Extend your file *map_<name>.html* by a simple OpenLayers map:
  - Put your JS code in a function that is called through the body's "onload" attribute
  - Background map *(e.g., OSM – as an OpenLayers layer, not WMS)*
  - Centre the map at WGS84 position 13.06072 E, 47.78869 N
    - The spatial reference system "EPSG:3857" is used for the map (!)
      → reprojection necessary

Exercise: Mobile Web Map

- Extend your file *map_<name>.html* by a simple OpenLayers map:
  - Add map controls (navigation buttons, overview map, scale bar, etc.)
  - ➔ Approach:
    - ➔ Create a scale line control object (ol.control.ScaleLine)
    - ➔ Add it to the map → extend default controls