

JavaScript

Lecture 3a (Client-side application)

Waterford Institute of Technology

June 9, 2016

John Fitzgerald

JavaScript Introduction

Topics discussed

- Scripts
- Chrome Developer Tools

Script Tags

Inserts program in html document

`<script></script>`

- Allows program in html
- Bad idea to place JavaScript in html
- Instead use source tags

```
<!DOCTYPE html>
<html>
  <head>
  </head>
  <body>
    ...
    ...
    <script src="js/jquery-2.0.0.js"></script>
    <script src="js/reportMap.js"></script>
  </body>
```

Script Tags

Where to locate

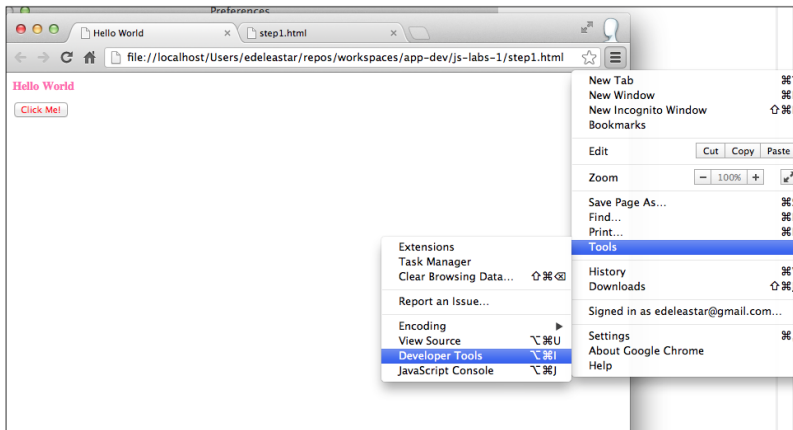
`<script></script>`

- Script files may have big impact on page load
- Place tags close as possible to bottom of body
- Place css `<link>` high as possible in head
- Reduce number of script files as much as possible
- Minify script files in release versions
 - Has big impact on load time

Chrome Dev Tools

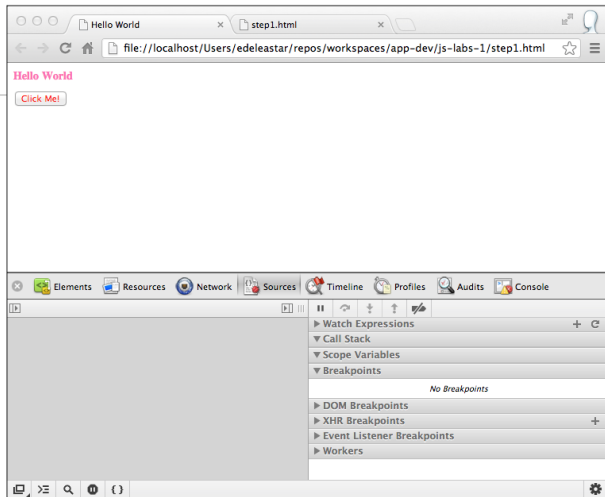
Web authoring & debugging tools

View->Developer Tools



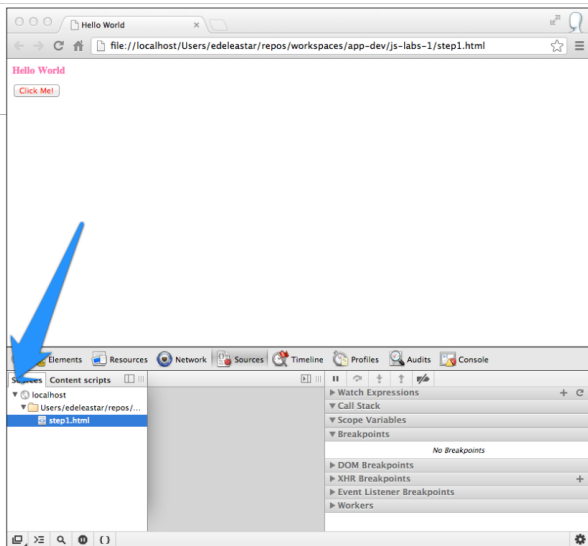
Chrome Dev Tools

Web authoring & debugging tools



Chrome Dev Tools

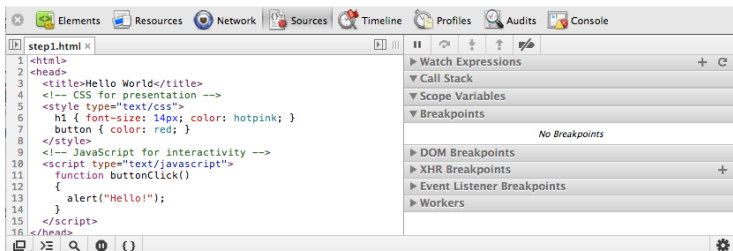
Web authoring & debugging tools



Chrome Dev Tools

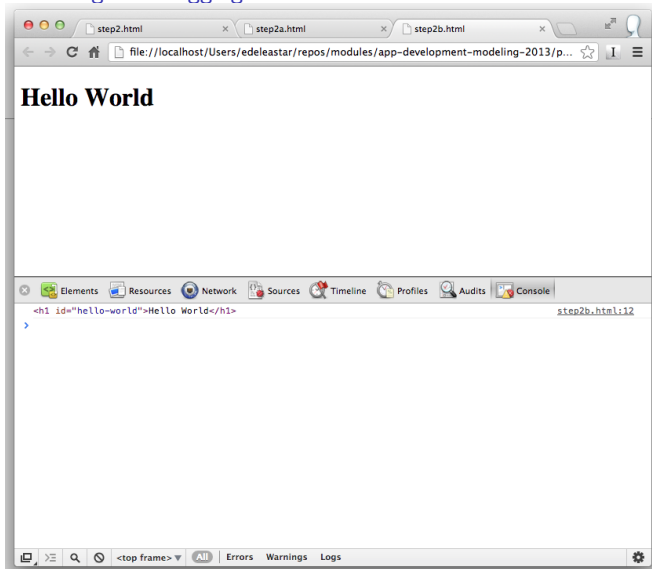
Web authoring & debugging tools

Source view in Developer Tools



Chrome Dev Tools

Web authoring & debugging tools



Example

Button press causes invocation JavaScript function

Page contains:

- paragraph <p>
- input elements <input>
 - text field
 - buttons
- list

This page contains a list, which will be modified by pressing the following button:

1. An Item

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Changing the DOM</title>
  <script type="text/javascript" src="script.js"></script>
</head>
<body>
  <p>
    This page contains a list, which will be modified by pressing the following button:
  </p>
  <input type="text" id="itemtext" />
  <input type="button" value="Add One" onclick="addElementById('itemtext')" />
  <input type="button" value="Clear All" onclick="clearList()" />
  <ol id="list">
    <li> An Item </li>
  </ol>
</body>
</html>
```

Example

JavaScript functions

```
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Changing the DOM</title>
  <script type="text/javascript" src="script.js"></script>
</head>
```

```
<input type="button" value="Add One" onclick="addElementById('itemText')" />
<input type="button" value="Clear All" onclick="clearList()" />
```

```
function addElementById(itemId)
{
  var list = document.getElementById('list');
  var itemText = document.getElementById(itemId);
  var newItem = document.createElement('li');
  newItem.innerHTML = itemText.value;
  list.appendChild(newItem);
}

function clearList()
{
  var list = document.getElementById('list');
  list.innerHTML = "";
}
```

script.js

- The script element identifies a file containing javascript functions
- The button elements identify the functions + parameters, to be called when the buttons are clicked
- The functions directly manipulate the DOM, changing the content of the current page

Static v Dynamic

JavaScript enabled page

Example

This page contains a list, which will be modified by pressing the following button:

1. An Item

This page contains a list, which will be modified by pressing the following button:

1. An Item
2. test one
3. test two

- For a static page, clicking on a link/button takes the browser to a new page (new url)
- With a dynamic page (javascript enabled), clicking on a button may change the *current* pages structure, content or style

Summary

- Scripts
 - How to load - in html or external files
 - Number scripts
 - Minification
- Developer tools
 - Chrome
 - Firefox



Except where otherwise noted, this content is licensed under a Creative Commons Attribution-NonCommercial 3.0 License.

For more information, please see <http://creativecommons.org/licenses/by-nc/3.0/>



Waterford Institute of Technology
INSTITIÚD TEICNEOLAÍOCHTA PHORT LÁIRGE

