HTML – CSS – JS BOOTSTRAP

Web Programming

AA 2013-2014 Gino Perna

HTML

- **HyperText Markup Language (HTML) is the main markup language for creating web pages and other information that can be displayed in a web browser.**
- **HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>), within the web page content. HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent empty elements and so are unpaired, for example <img>. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags). In between these tags web designers can add text, further tags, comments and other types of text-based content.**
- **The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page.**

HTML

- **HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as JavaScript which affect the behavior of HTML web pages.**
- **Web browsers can also refer to Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both the HTML and the CSS standards, encourages the use of CSS over explicit presentational HTML.**

HTML

- **Sito di riferimento: http://www.w3schools.com/**
  - **Contiene tutti i tutorial e le lezioni di riferimento aggiornati**

HTML history

HTML History

HTML versions timeline

- 1986 – Tim Berners-Lee proposed at CERN a way for sharing documents inside organization – base of the modern HML
- November 24, 1995 -HTML 2.0 was published as IETF RFC 1866 . Supplemental RFCs added capabilities up to 1997
- January 1997 - HTML 3.2 was published as a W3C Recommendation. It was the first version developed and standardized exclusively by the W3C, as the IETF had closed its HTML Working Group in September 1996.

HTML History

- (cont Jan 1997)Initially code-named "Wilbur", HTML 3.2 dropped math formulas entirely, reconciled overlap among various proprietary extensions and adopted most of Netscape's visual markup tags. Netscape's blink element and Microsoft's marquee element were omitted due to a mutual agreement between the two companies. A markup for mathematical formulas similar to that in HTML was not standardized until 14 months later in MathML.

HTML History

- December 1997 - HTML 4.0[16] was published as a W3C Recommendation . It offers three variations:
  - Strict, in which deprecated elements are forbidden,
  - Transitional, in which deprecated elements are allowed,
  - Frameset, in which mostly only frame related elements are allowed ;

HTML History

- December 1999 - HTML 4.01 was published as a W3C Recommendation. It offers the same three variations as HTML 4.0 and its last errata were published May 12, 2001.
- May 2000 ISO/IEC 15445:2000 ("ISO HTML", based on HTML 4.01 Strict) was published as an ISO/IEC international standard. In the ISO this standard falls in the domain of the ISO/IEC JTC1/SC34.
- As of mid-2008, HTML 4.01 and ISO/IEC 15445:2000 are the most recent versions of HTML. Development of the parallel, XML-based language XHTML occupied the W3C's HTML Working Group through the early and mid-2000s.

HTML History

- May 2011 - On 14 February 2011, the W3C extended the charter of its HTML Working Group with clear milestones for HTML5. In May 2011, the working group advanced HTML5 to "Last Call", an invitation to communities inside and outside W3C to confirm the technical soundness of the specification. The W3C is developing a comprehensive test suite to achieve broad interoperability for the full specification by 2014, which is now the target date for Recommendation

Markup

- HTML markup consists of several key components, including elements (and their attributes), character-based data types, character references and entity references. Another important component is the document type declaration, which triggers standards mode rendering.
- The following is an example of the classic Hello world program, a common test employed for comparing programming languages, scripting languages and markup languages. This example is made using 9 lines of code:

Markup

```
<!DOCTYPE html>
<html>
  <head>
    <title>This is a title</title>
  </head>
  <body>
    <p>Hello world!</p>
  </body>
</html>
```

- (The text between <html> and </html> describes the web page, and the text between <body> and </body> is the visible page content. The markup text '<title>This is a title</title>' defines the browser page title.)
- This Document Type Declaration is for HTML5.

Page debugging

Strumenti per il debugging

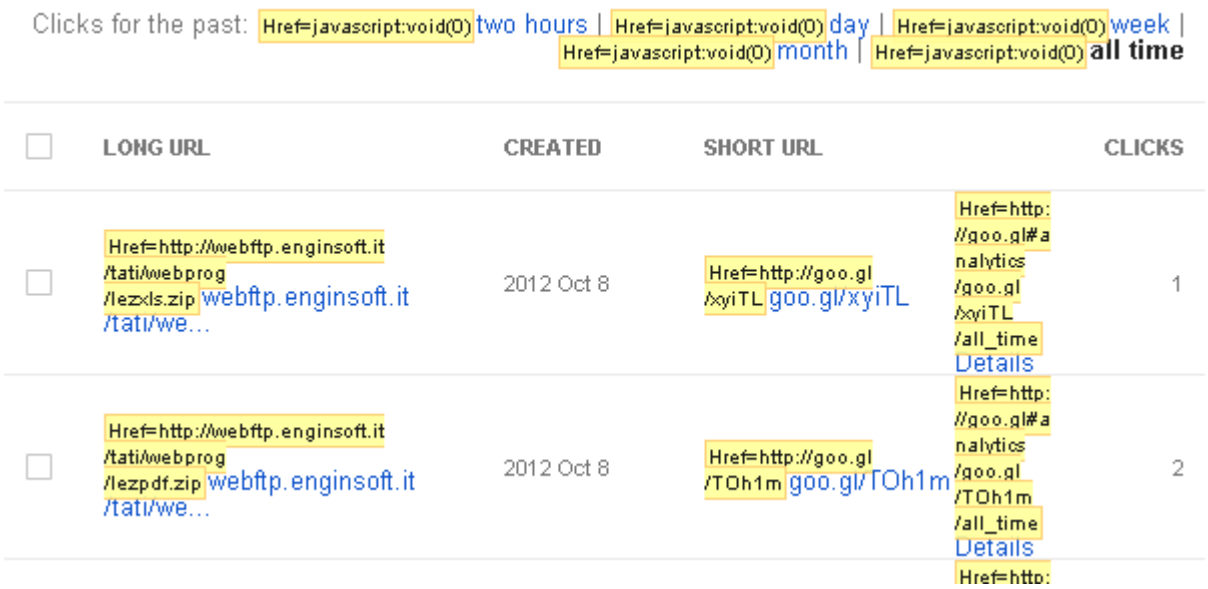
Browser

- Installare almeno due browser tra cui:
  - Firefox
  - Chrome

- Altri browsers possono essere utili per controllare le pagine ma non per il debugging

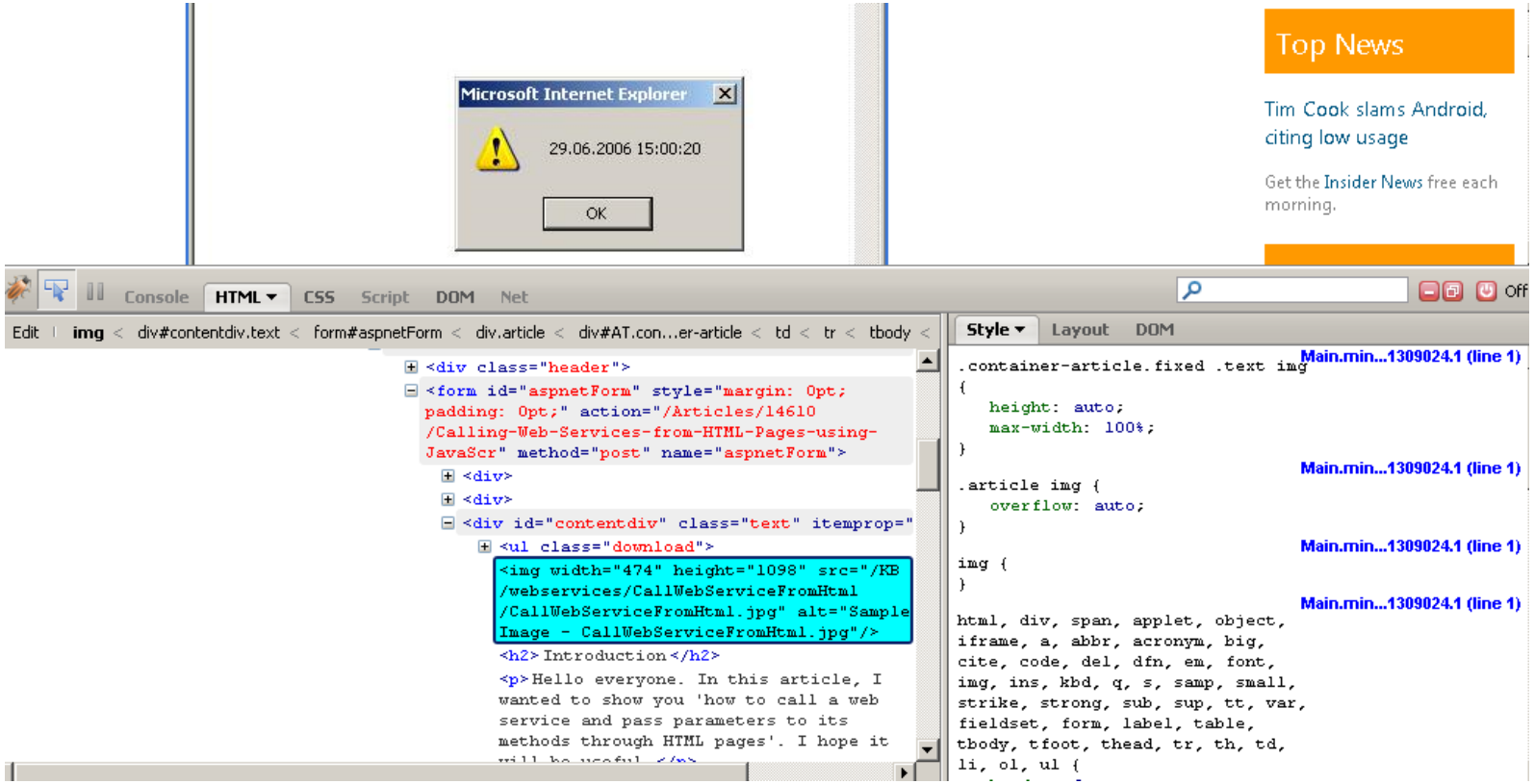
Browser extensions

- Web-developer: utilissimo per cambiamenti globali e visulaizzazione dei tag sulla pagina



Browser extensions

- Firebug: inspector elementi



Editor delle pagine html

- Quello preferito:
  - Notepad++
  - Netbeans
  - Kompozer
  - ….

Esercizi interattivi

- http://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_submit

Un framework per le pagine

Bootstrap

- Bootstrap is a free collection of tools for creating websites and web applications. It contains HTML and CSS-based design templates for typography, forms, buttons, navigation and other interface components, as well as optional JavaScript extensions.
- It has been the most popular project in GitHub and has been used by NASA and MSNBC among others

Bootstrap origin

- Bootstrap was developed by Mark Otto and Jacob Thornton at Twitter as a framework to encourage consistency across internal tools. Before Bootstrap, various libraries were used for interface development, which led to inconsistencies and a high maintenance burden. According to Twitter developer Mark Otto, in the face of those challenges:
- "...[A] super small group of developers and I got together to design and build a new internal tool and saw an opportunity to do something more. Through that process, we saw ourselves build something much more substantial than another internal tool. Months later, we ended up with an early version of Bootstrap as a way to document and share common design patterns and assets within the company."[5]

Bootstrap origin

- The first deployment under real conditions happened during Twitter's first Hackweek." Mark Otto showed some colleagues how to accelerate their project's development with the help of the toolkit. As a result, dozens of teams have moved to the framework.
- In August, 2011 Twitter released Bootstrap as open-source. In February 2012, it was the most popular GitHub development project.

Bootstrap features

- Bootstrap has relatively incomplete support for HTML5 and CSS 3, but it is compatible with all major browsers. Basic information of compatibility of websites or applications is available for all devices and browsers. There is a concept of partial compatibility that makes the basic information of a website available for all devices and browsers. For example, the properties introduced in CSS3 for rounded corners, gradients and shadows are used by Bootstrap despite lack of support by older web browsers. These extend the functionality of the toolkit, but are not required for its use.
- Since version 2.0 it also supports responsive design. This means the layout of web pages adjusts dynamically, taking into account the characteristics of the device used (PC, tablet, mobile phone).

Structure and function

- Bootstrap is modular and consists essentially of a series of LESS stylesheets that implement the various components of the toolkit. A stylesheet called bootstrap.less includes the components stylesheets. Developers can adapt the Bootstrap file itself, selecting the components they wish to use in their project.
- Adjustments are possible to a limited extent through a central configuration stylesheet. More profound changes are possible by the LESS declarations.
- The use of LESS stylesheet language allows the use of variables, functions and operators, nested selectors, as well as so-called mixins.
- Since version 2.0, the configuration of Bootstrap also has a special "Customize" option in the documentation. Moreover, the developer chooses on a form the desired components and adjusts, if necessary, the values of various options to their needs. The subsequently generated package already includes the pre-built CSS style sheet.

Grid system and responsive design

Bootstrap comes standard with a 940 pixel wide, grid layout. Alternatively, the developer can use a variable-width layout. For both cases, the toolkit has four variations to make use of different resolutions and types of devices: mobile phones, portrait and landscape, tablets and PCs with low and high resolution. Each variation adjusts the width of the columns.

CSS and components

- Bootstrap provides a set of stylesheets that provide basic style definitions for all key HTML components. These provide uniform, modern appearance for formatting text, tables and form elements.
- In addition to the regular HTML elements, Bootstrap contains other commonly used interface elements. These include buttons with advanced features (e.g. grouping of buttons or buttons with drop-down option, make and navigation lists, horizontal and vertical tabs, navigation, breadcrumb navigation, pagination, etc.), labels, advanced typographic capabilities, thumbnails, warning messages and a progress bar.

Javascript component

- Bootstrap comes with several JavaScript components in a form of jQuery plugin. They provide additional user-interface elements such as dialog boxes, tooltips, and carousels. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields. In version 2.0, the following JavaScript plugins are supported: Modal, Dropdown, Scrollspy, Tab, Tooltip, Popover, Alert, Button, Collapse, Carousel and Typeahead.
- An implementation of Twitter Bootstrap using the Dojo Toolkit is also available. It is called Dojo Bootstrap and is a port of the Twitter Bootstrap plugins. It uses 100% Dojo code and has support for AMD (Asynchronous Module Definition).

Componenti da scaricare

- https://github.com/twbs/bootstrap/releases/download/v3.0.0/bootstrap-3.0.0-dist.zip
- Lo zip contiene la struttura base di un sito con tutti i suoi componenti (sito completamente vuoto)

```
bootstrap/
├── css/
│   ├── bootstrap.css
│   ├── bootstrap.min.css
│   ├── bootstrap-theme.css
│   ├── bootstrap-theme.min.css
├── js/
│   ├── bootstrap.js
│   ├── bootstrap.min.js
└── fonts/
    ├── glyphicons-halflings-regular.eot
    ├── glyphicons-halflings-regular.svg
    ├── glyphicons-halflings-regular.ttf
    └── glyphicons-halflings-regular.woff
```

Barebone Example

```
<!DOCTYPE html>
<html>
  <head>
    <title>Bootstrap 101 Template</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <!-- Bootstrap -->
    <link href="css/bootstrap.min.css" rel="stylesheet" media="screen">

    <!-- HTML5 shim and Respond.js IE8 support of HTML5 elements and media queries -->
    <!--[if lt IE 9]>
      <script src="../../assets/js/html5shiv.js"></script>
      <script src="../../assets/js/respond.min.js"></script>
    <![endif]-->
  </head>
  <body>
    <h1>Hello, world!</h1>

    <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
    <script src="//code.jquery.com/jquery.js"></script>
    <!-- Include all compiled plugins (below), or include individual files as needed -->
    <script src="js/bootstrap.min.js"></script>
  </body>
</html>
```

Esercizi da reimplementare in locale

http://getbootstrap.com/examples/starter-template/

http://getbootstrap.com/examples/grid/

http://getbootstrap.com/examples/jumbotron/

http://getbootstrap.com/examples/navbar/

http://getbootstrap.com/examples/signin/

http://getbootstrap.com/examples/carousel/

Provare gli esempi in http://getbootstrap.com/components/

Riferimenti

http://getbootstrap.com

HTML5

Html5 selected differences

Introduction

- New features should be based on HTML, CSS, DOM, and JavaScript
- Reduce the need for external plugins (like Flash)
- Better error handling
- More markup to replace scripting
- HTML5 should be device independent
- The development process should be visible to the public

HTML5 is not yet an official standard, and no browsers have full HTML5 support.

New features

Some of the most interesting new features in HTML5:

- The <canvas> element for 2D drawing
- The <video> and <audio> elements for media playback
- Support for local storage
- New content-specific elements, like <article>, <footer>, <header>, <nav>, <section>
- New form controls, like calendar, date, time, email, url, search

Drag and Drop

Any element can be **draggable**.

```
<!DOCTYPE HTML>
<html>
<head>
<script>
function allowDrop(ev)
{
ev.preventDefault();
}

function drag(ev)
{
ev.dataTransfer.setData("Text",ev.target.id);
}

function drop(ev)
{
ev.preventDefault();
var data=ev.dataTransfer.getData("Text");
ev.target.appendChild(document.getElementById(data));
}
</script>
</head>
<body>

<div id="div1" ondrop="drop(event)"
ondragover="allowDrop(event)"></div>

<img id="drag1" src="img_logo.gif" draggable="true"
ondragstart="drag(event)" width="336" height="69">

</body>
</html>
```

Geolocation

The HTML5 Geolocation API is used to get the geographical position of a user.

Since this can compromise user privacy, the position is **not available** unless the user approves it.

Geolocation is also very useful for location-specific information. Examples:

- Up-to-date local information
- Showing Points-of-interest near the user
- Turn-by-turn navigation (GPS)

Example

```
<script>
var x=document.getElementById("demo");
function getLocation()
  {
  if (navigator.geolocation)
    {
    navigator.geolocation.getCurrentPosition(showPosition);
    }
  else{x.innerHTML="Geolocation is not supported by this browser.";}
  }
function showPosition(position)
  {
  x.innerHTML="Latitude: " + position.coords.latitude +
  "<br>Longitude: " + position.coords.longitude;
  }
</script>
```

This example is a simple Geolocation example returning the latitude and longitude of the user's position.

New Form Attributes

<input> list Attribute Since web workers are in external files, they do not have access to the following JavaScript objects:

The list attribute refers to a <datalist> element that contains pre-defined options for an <input> element.

```
<input list="browsers">

<datalist id="browsers">
  <option value="Internet Explorer">
  <option value="Firefox">
  <option value="Chrome">
  <option value="Opera">
  <option value="Safari">
</datalist>
```

*Safari – no support!

New Form Attributes

Regular Expressions How often have you found yourself writing some quickie regular expression to verify a particular textbox. Thanks to the new pattern attribute, we can insert a regular expression directly into our markup.

```
<form action="" method="post">
    <label for="username">Create a Username: </label>
    <input type="text"
       name="username"
       id="username"
       placeholder="4 <> 10"
       pattern="[A-Za-z]{4,10}"
       autofocus
       required>
    <button type="submit">Go </button>
</form>
```

Web Storage

HTML5 web storage is a better local storage than cookies.

web pages can store data locally within the user's browser. Earlier, this was done with cookies. However, Web Storage is more secure and faster. The data is not included with every server request, but used ONLY when asked for. It is also possible to store large amounts of data, without affecting the website's performance.

The data is stored in key/value pairs, and a web page can only access data stored by itself.

localStorage and sessionStorage

There are two new objects for storing data on the client:

- localStorage - stores data with no expiration date
- sessionStorage - stores data for one session

Application Cache

a web application is cached, and accessible without an internet connection.

Advantages:

- Offline browsing - users can use the application when they're offline
- Speed - cached resources load faster
- Reduced server load - the browser will only download updated/changed resources from the server.

<html manifest="demo.appcache">...</html>

- Be careful with what you cache!!!
- Once a file is cached, the browser will continue to show the cached version, even if you change the file on the server. To ensure the browser updates the cache, you need to change the manifest file.
- **Note:** Browsers may have different size limits for cached data (some browsers have a 5MB limit per site).

Web Workers

A web worker is a JavaScript running in the background, without affecting the performance of the page.

When executing scripts in an HTML page, the page becomes unresponsive until the script is finished. A web worker is a JavaScript that runs in the background, independently of other scripts, without affecting the performance of the page. You can continue to do whatever you want: clicking, selecting things, etc., while the web worker runs in the background.

Note: Since web workers are in external files, they do not have access to JavaScript objects like window, document and parent object.

*Internet Explorer 10, Firefox, Chrome, Safari and Opera support Web workers.