Lezione 2: HTML

Linguaggio di markup

- Linguaggio di markup modellato per rendere esplicita una particolare interpretazione di un testo
  - Per esempio, tutte quelle aggiunte al testo scritto che permettono di renderlo più fruibile
  - Oltre a rendere il testo più leggibile, il markup permette anche di specificare ulteriori usi del testo
- Con il markup per sistemi informatici, specifichiamo le modalità esatte di utilizzo del testo nel sistema stesso
- I linguaggi di markup sono i linguaggi più opportuni per strutturare e marcare i documenti in maniera indipendente dall'applicazione, favorendo la riusabilità, la flessibilità e la apertura ad applicazioni complesse

HyperText Markup Language (HTML)

- HTML linguaggio per la creazione di pagine Web
  - Standard per la rappresentazione
  - Descrive la struttura della pagina
  - Non è un linguaggio di programmazione, è un linguaggio di markup
  - Descrive dati e regole su come mostrarli
  - Poche e semplici regole sintattiche

- File di testo

Caratteristiche

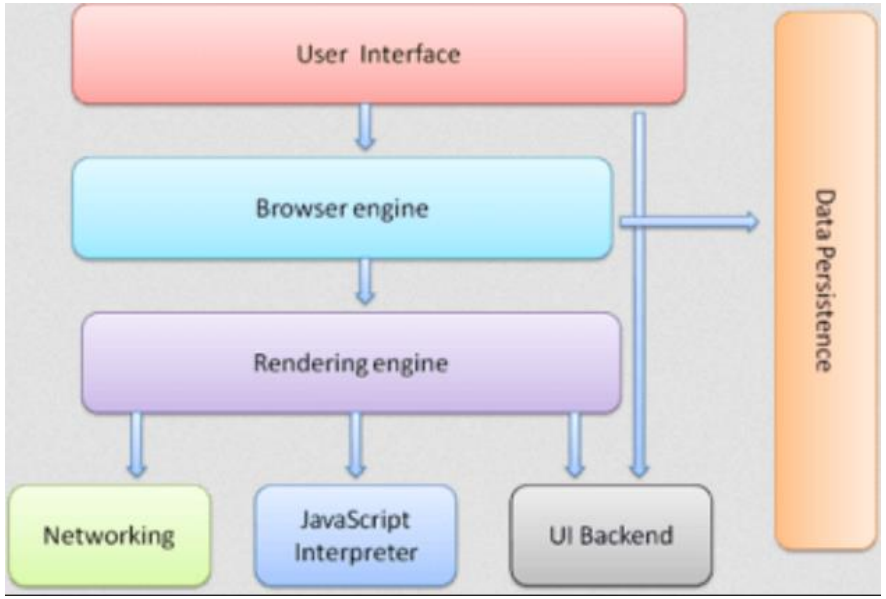
- Ipertesto
- Multimedia
- Ipermedia

Browser - 1

- Programma per la navigazione nel Web
- Richiede risorse attraverso il web (all'indirizzo richiesto) e le mostra nella finestra
- Legge e interpreta i documenti HTML, CSS, JavaScript, immagini sulla base di specifiche e standard
- Browser: Internet Explorer, Mozilla Firefox, Chrome, Opera, Safari

Browser - 2

- **Layout Engine –** riceve input dal browser (URL bar, search box, mouse clicks and key presses) e li passa al rendering engine
- **Rendering Engine** – riceve il codice HTML e lo interpreta visualizzandolo a video. Ad esempio, testo in bold
- **User Interface** – controlli del browser, ad esempio, bottoni per andare avanti e indietro, bookmark
- **JavaScript Engine** – engine che si prende cura di parsare ed eseguire codice JavaScript per poi ritornare il risultato dell'esecuzione
- **Network Layer** – gestisce funzioni di rete, ad esempio, criptazione, richieste http e tutte le configurazioni di rete
- **Storage** – porzione di memoria dove il browser memorizza cached file, cookie e oggetti e dati create con JavaScript
- **Operating System Interface** – componente che interagisce con il sistema operativo per disegnare diversi elementi e per gestire la finestra (chiudi, massimizza, minimizza)



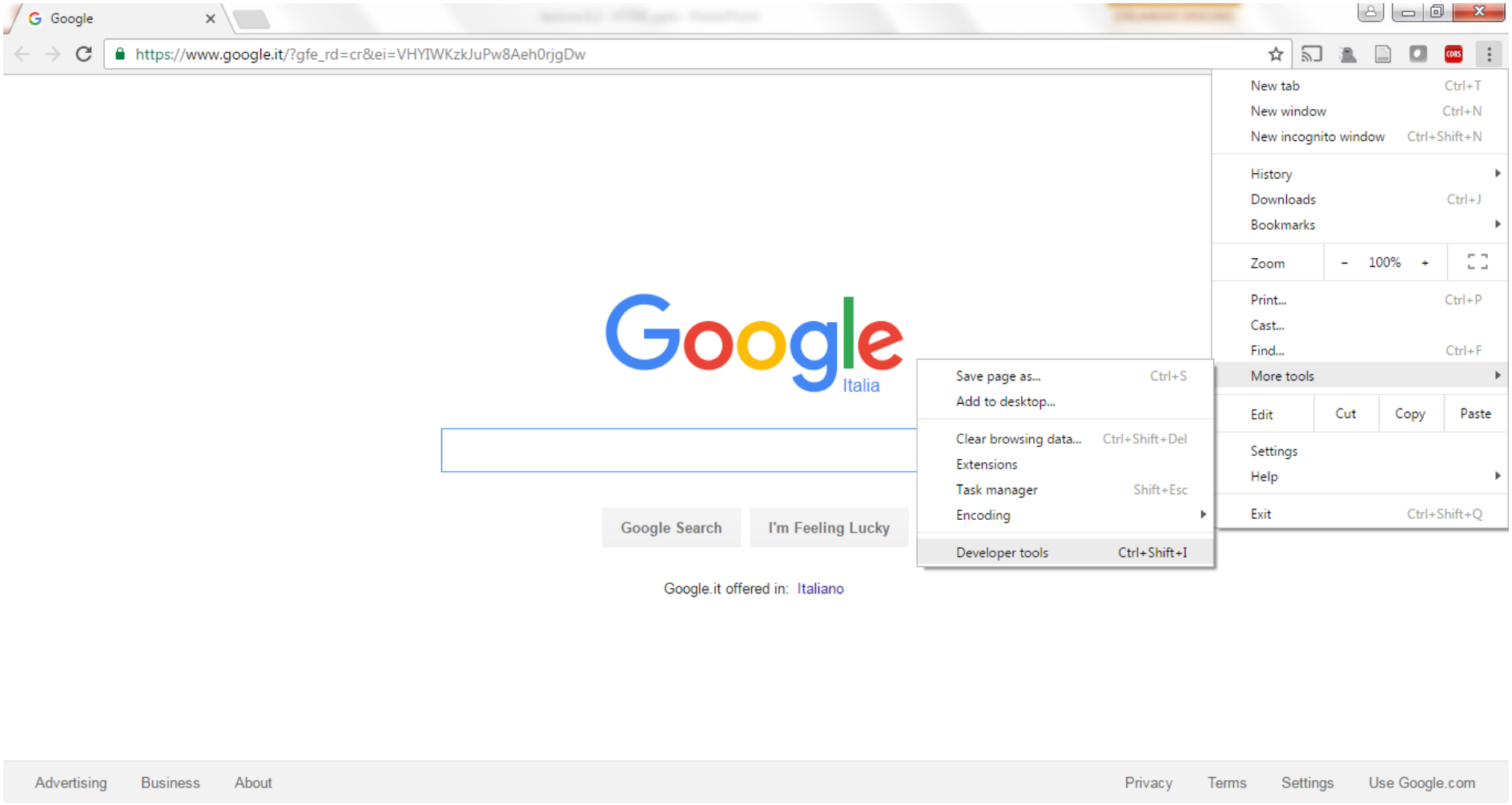
Browser - 3

- Mentalità multi-browser
- Obiettivo: rendere visibile la propria pagina su tutti i browser
- Rendering

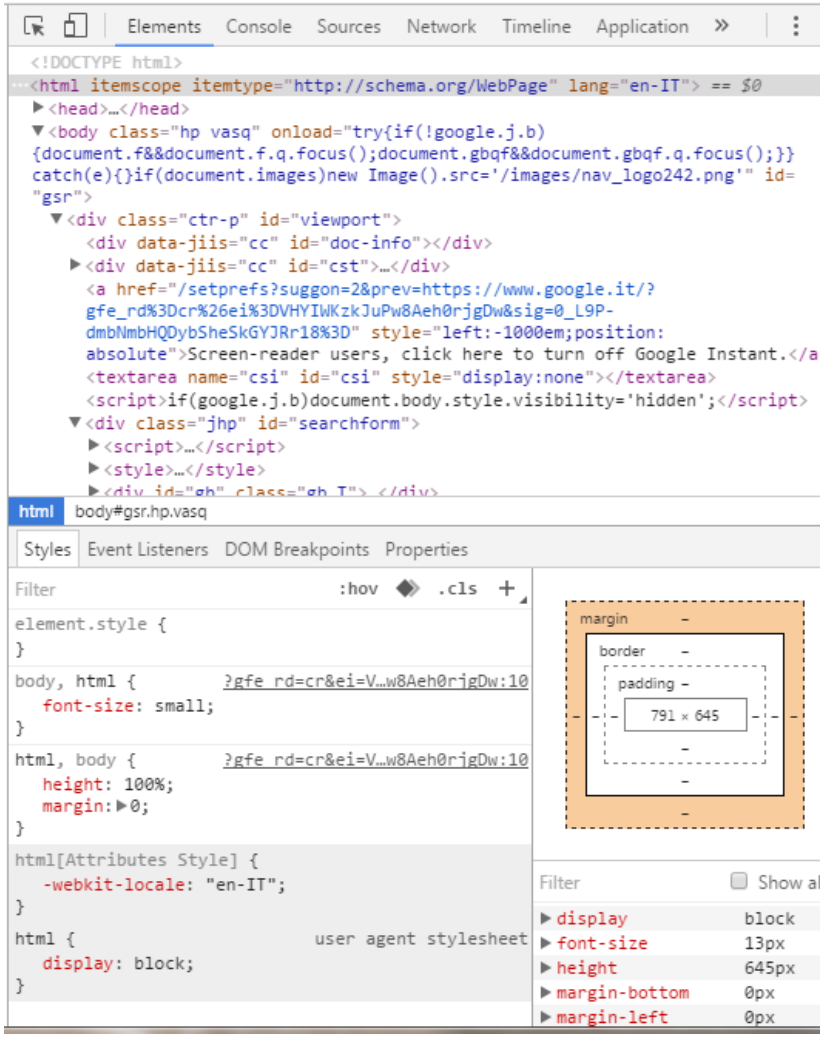
Browser - 4

- Contiene tool per gli sviluppatori
  - Debug pagina
  - Analisi stili e sorgenti dei file
  - Visualizzazione HTTP request e response
  - Accesso a cookie e storage
  - Molto altro…

Browser - 5


Google
https://www.google.it/?gfe_rd=cr&ei=VHYIWKzkJuPw8Aeh0rjgDw
New tab Ctrl+T
New window Ctrl+N
New incognito window Ctrl+Shift+N
History
Downloads Ctrl+J
Bookmarks
Zoom - 100% +
Print... Ctrl+P
Cast...
Find... Ctrl+F
More tools
Edit Cut Copy Paste
Settings
Help
Exit Ctrl+Shift+Q
Save page as... Ctrl+S
Add to desktop...
Clear browsing data... Ctrl+Shift+Del
Extensions
Task manager Shift+Esc
Encoding
Developer tools Ctrl+Shift+I
Google
Italia
Google Search
I'm Feeling Lucky
Google.it offered in: Italiano
Advertising
Business
About
Privacy
Terms
Settings
Use Google.com

Browser - 6


Elements
Console
Sources
Network
Timeline
Application
<!DOCTYPE html>
<html itemscope itemtype="http://schema.org/WebPage" lang="en-IT"> == $0
<head>…</head>
<body class="hp vasq" onload="try{if(!google.j.b)
{document.f&&document.f.q.focus();document.gbqf&&document.gbqf.q.focus();}}
catch(e){}if(document.images)new Image().src='/images/nav_logo242.png'" id=
"gsr">

…
<a href="/setprefs?suggon=2&prev=https://www.google.it/?
gfe_rd%3Dcr%26ei%3DVHYIWKzkJuPw8Aeh0rjgDw&sig=0_L9P-
dmbNmbHQDybSheSkGYJRr18%3D" style="left:-1000em;position:
absolute">Screen-reader users, click here to turn off Google Instant.</a
<textarea name="csi" id="csi" style="display:none"></textarea>
<script>if(google.j.b)document.body.style.visibility='hidden';</script>

<script>…</script>
<style>…</style>
html
body#gsr.hp.vasq
Styles
Event Listeners
DOM Breakpoints
Properties
Filter
:hov
.cls
element.style {
}
body, html {
?gfe_rd=cr&ei=V…w8Aeh0rjgDw:10
font-size: small;
}
html, body {
?gfe_rd=cr&ei=V…w8Aeh0rjgDw:10
height: 100%;
margin:0;
}
html[Attributes Style] {
-webkit-locale: "en-IT";
}
html {
user agent stylesheet
display: block;
}
margin
border
padding
791 × 645
Filter
Show al
display
block
font-size
13px
height
645px
margin-bottom
0px
margin-left
0px

Scrivere una pagina Web

- Creazione file .html o .htm
- Editor Testuale vs Editor Visuale
  - Text editor come Notepad++, Editplus, Textmate, Dream weaver
- Visualizzazione pagina tramite browser
  - Browser moderni come ad esempio Firefox, Chrome, Safari

HTML

- HTML non è un linguaggio di programmazione
- Linguaggio di markup (tag)
- Nessun meccanismo di decisione e iterazione
- Poche regole sintattiche

Da HTML a XHTML…

- HTML
  - W3C (WWW Consortium): HTML 2.0, HTML 3.2, HTML 4.01
- XHTML è quasi identico a HTML
  - Più stringente di HTML
  - Supportato dai principali browser
  - Introdotto per ridurre l'impatto delle pagine web errate sulla visualizzazione delle pagine stesse e sul parsing dei browser
    - Interpretare un markup sbagliato richiede maggiori risorse o potenza computazionale (più complicato per i device più piccoli)
- W3C fornisce un meccanismo per la validazione delle pagine web
  - http://validator.w3.org

… fino a HTML5

- HTML5 è lo standard attuale per strutturare e presentare il contenuto del World Wide Web

- Mira a risolvere le **limitazioni** di HTML4.01 e precedenti che hanno limitato l'evoluzione del web
  - Bisogno di flash/quicktime per eseguire audio/video

  - Impossibilità di memorizzazione dati lato client se non con linguaggi di scripting o altre tecnologie

  - Mancanza di un formato nativo per il disegno
    - Grafica o animazioni fornite come immagini o con Flash, Java, Silverlight

HTML5

- HTML5 è una cooperazione tra World Wide Web Consortium (W3C) e Web Hypertext Application Technology Working Group (WHATWG)
- Gennaio 2010, W3C introduce un logo HTML5 logo per uso pubblico
  - Promuove le nuove funzionalità di HTML5 e le tecnologie corrispondenti
- Logo disponibile ai siti web per mostrare il loro supporto per questa nuova tecnologia
- W3C usa HTML5 come contenitore di altre tecnologie

HTML5

- HTML5 è uno standard per strutturare e presentare il contenuto del World Wide Web

- Cos'è HTML5 (specification)?
  - Successore di HTML 4.01 e XHTML 1.1
  - Aggiunge nuovi tag, funzionalità e API

- Cos'è HTML5 (family)?
  - Una suite di tool per il markup (HTML 5)
    - Presentazione (CSS 3), interazione (DOM, Ajax, API)
  - HTML5 ≈ HTML5 + CSS3 + JavaScript
    - Una combinazione di HTML5 markup tag, proprietà CSS3, JavaScript e tecnologie di supporto
  - Introdotto per catturare l'evoluzione nell'utilizzo del web

HTML5

- Pragmatico
  - Parte dalla considerazione che i browser decidono cosa implementare e cosa no
  - Non ha senso tradurre in specifica qualcosa che non viene utilizzato
- Fornisce supporto per tutto ciò che è JavaScript
  - L'uso di JavaScript mostrava le lacune di HTML
- Aggiunge semantica
  - Migliore definizione delle componenti delle pagine web
  - Tag HTML forniscono informazioni aggiuntive utili per motori di ricerca

HTML5

- 91,0% dei siti web adotta funzionalità HTML5, 94,9% adotta HTML (W3Techs – November 2022)
- HTML5 non è una, ma un gruppo di tecnologie
  - Sviluppatori decidono cosa usare e cosa no di HTML5
- La potenza di HTML5 si vede dal fatto che HTML5 è usata all'interno del sistema operativo Microsoft Windows (a partire dalla versione 8)
- Mobile device e smartphone usano HTML5

Storia degli standard web

- Working Draft (WD)
  - Documento che il W3C pubblica per ottenere una revisione dalla comunità (include W3C Member, pubblico e altre organizzazioni tecniche)
- Candidate Recommendation (CR)
  - Documento revisionato in dettaglio che il W3C ritiene soddisfi i requisiti tecnici del Group
  - W3C pubblica una Candidate Recommendation per raccogliere esperienza implementativa
- Proposed Recommendation (PR)
  - Un technical report maturo che, dopo una dettagliata revisione che valuta l'implementabilità e la correttezza della soluzione, W3C invia al W3C Advisory Committee per un endorsement finale
- W3C Recommendation (REC)
  - Una specifica o insieme di guideline che, dopo aver raccolto un ampio consenso, ha ricevuto l'endorsement dei W3C Member e del W3C Director
  - W3C raccomanda il deployment e l'utilizzo della Recommendation
  - W3C Recommendation sono simili a standard pubblicati da altre organizzazioni

Storia degli standard web

91-92	93-94	95-96	97-98	99-00	01-02	03-04	05-06	07-08	09-10	11-12	13-14
HTML 1	HTML 2	HTML 3	HTML 4	XHTML 1					HTML 5		
		CSS 1	CSS 2			Web 2.0			CSS3		
		JS	XML 1.0, DOM	DOM 2		XML 1.1	Ajax		DOM, APIs		

HTML

1991	HTML 1 (HTML Tag)
1994	HTML 2 Draft
1995	HTML 3 Internet Draft
1997	HTML 3.2 W3C Rec
1997-98	HTML 4.0 W3C Proposed Rec
2008	HTML 5 W3C Working Draft
2012	HTML 5 W3C Candidate Rec
2014	HTML 5 W3C Rec

Storia degli standard web

- Ultima parte del 1991: prima descrizione di HTML (HTML tag)
- 1993-94: HTML 2.0 pubblicato
- Marzo 1995: HTML 3.0 pubblicato come internet draft
- Dicembre 1997: HTML 4.0 is published by the W3C
- Febbraio-Marzo 1998: XML 1.0 is published

Storia degli standard web

- Dicembre 1999-Gennaio 2000: ECMAScript 3rd Edition, XHTML 1.0 (HTML tag riformulati in XML)e HTML 4.01 recommendation pubblicate
- Maggio 2001: XHTML 1.1 recommendation pubblicata
- Agosto 2002: XHTML 2.0 first working draft rilasciato
- Dicembre 2002: XHTML 2.0 second working draft pubblicato
- Gennaio 2008: First W3C working draft di HTML5 pubblicato

Struttura, sintassi e contenuto di una pagina HTML

HTML - Tag

- Tag sono marcatori che identificano porzioni di testo
- Permettono la personalizzazione della pagina, e di identificare e processare le porzioni di testo contrassegnate da una certa marcatura
- Nome tag = nome funzione

Struttura tag

- Tag contenuto tra parentesi triangolari
- Tag di inizio, tag di fine
- Contenuto del tag
- <tag attributi>contenuto</tag>

Esempio tag

- Tag con contenuto
  - testo
- Tag senza contenuto
  - <img src="mialmmagine.jpg">

Struttura della pagina web

- Semantic markup
  - Definizione e scelta dell'elemento migliore per l'oggetto che si vuole rappresentare
  - Sintassi che da l'idea di cosa si sta facendo sia a un uomo che a una macchina (browser)
  - Associa un significato al contenuto
  - Pagine facili da capire e modificare per sviluppatore, adatte per mostrare il contenuto in maniera comprensibile a utenti con disabilità

Struttura della pagina web

- Riga di intestazione
  - HTML 4.0.1
    - <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN“ "http://www.w3.org/TR/html4/loose.dtd">
  - Più semplice per HTML5
    - <!DOCTYPE html>
- Indichiamo le specifiche W3C adottate
- Dice al browser quale linguaggio è stato usato per il rendering

Struttura della pagina web

```
<!DOCTYPE html>
<html>
<head>
<title>Title of the document</title>
</head>

<body>
        That’s all I need to create my first HTML5 page
</body>

</html>
```

Struttura della pagina web

- Il tag <html> indica l'inizio di una pagina HTML

- Tutto ciò compreso all'interno del tag <html> è il codice HTML

```
<html>
                … codice HTML …
</html>
```

Struttura della pagina web

- Documento HTML diviso in due
  - Testa
  - Corpo

Testa

- Contiene informazioni non immediate
- Descrive come il documento deve essere letto e interpretato
- Contiene meta-tag, script, stili
- Racchiuso tra il tag <head>

Corpo

- Contiene il documento vero e proprio
- Contiene tutti i tag per la realizzazione del sito Web (ad esempio <font …>)
- Racchiuso tra il tag <body>

Esempio

- Scrivere una pagina HTML con i tag essenziali (html, head, body) inserendo una stringa all'interno del corpo

- primo.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//IT">
<html>
<head>
</head>
<body>
Hello World!
</body>
</html>
```

Indentazione

- Buona norma utilizzare caratteri di tabulazione
- Aumenta leggibilità
- Diminuisce tempi di modifica

Esempio - Indentazione

```
<html>
        <head>
        </head>
        <body>
                Hello World!
        </body>
</html>

<html><head></head>    <body>
        Hello World!</body></html>
```

Esempio - Indentazione

- Scrittura equivalente per il browser

- Non per l'uomo
  - Un'unica pagina HTML su una stessa riga risulterebbe illeggibile

Annidamento

- Caratteristica HTML: tag annidati

```
<tag1 attributi>
        contenuto1
        <tag2 attributi>
                        contenuto2
        </tag2>
</tag1>
```

- Formattazioni successive

Annidamento

- Esempio

```
<font color="blue">
        Hello
        <font color="red">
                World!
        </font>
</font>
```

- Hello World!

Commenti

- Aumenta la leggibilità del codice
- Buona norma commentare SOLO le parti significative
- Permetteno insieme all'indentazione di orientarsi all'interno di un grosso documento

Esempio - Commenti

- <!-- = inizio commento
- --> = fine commento
- <!-- questo è un commento -->

Maiuscolo o minuscolo

- Case unsensitive
  - testo
  - <FONT COLOR="RED">testo</FONT>
- XHTML case sensitive
- Consigliabile usare carattere minuscolo

Altre caratteristiche

- Non è sensibile agli spazi

Hello World !

- Non è sensibile alle linee vuote

Hello

World !

- Stringa visualizzata: Hello World !
- Esempio1.html

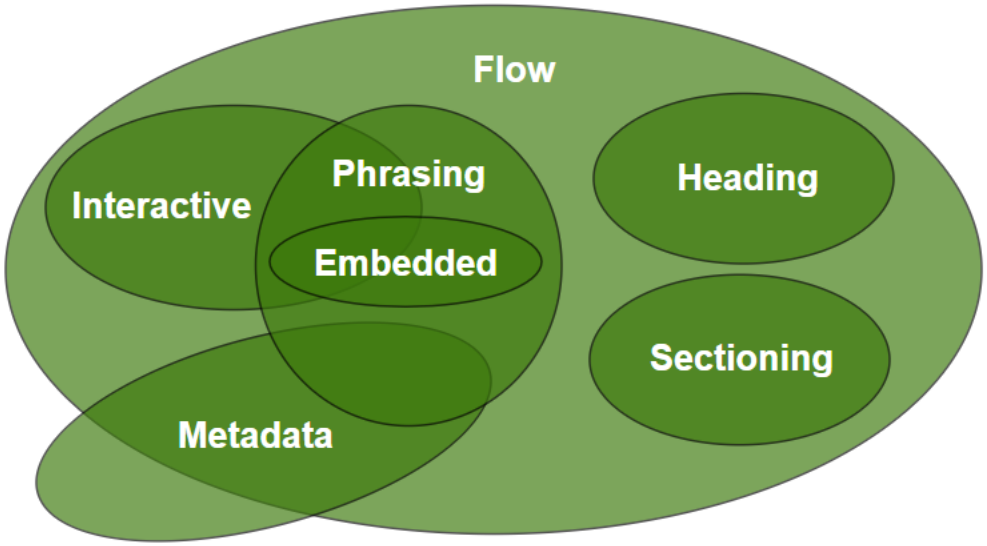
Sintassi (Summary)

- Permissiva per garantire compatibilità
  - Massimizzazione del rendering corretto
- Case insensitive: maiuscole e minuscole non interferiscono con la validazione della pagina
  - <H1> Text </h1>
- I tag di chiusura non sono richiesti
  - <p> first paragraph
  - <p> second paragraph
- Le virgolette e i valori sono opzionali per gli attributi
  - <img src=trafficjam.jpg alt=traffic jam>

Categorie di contenuti HTML5

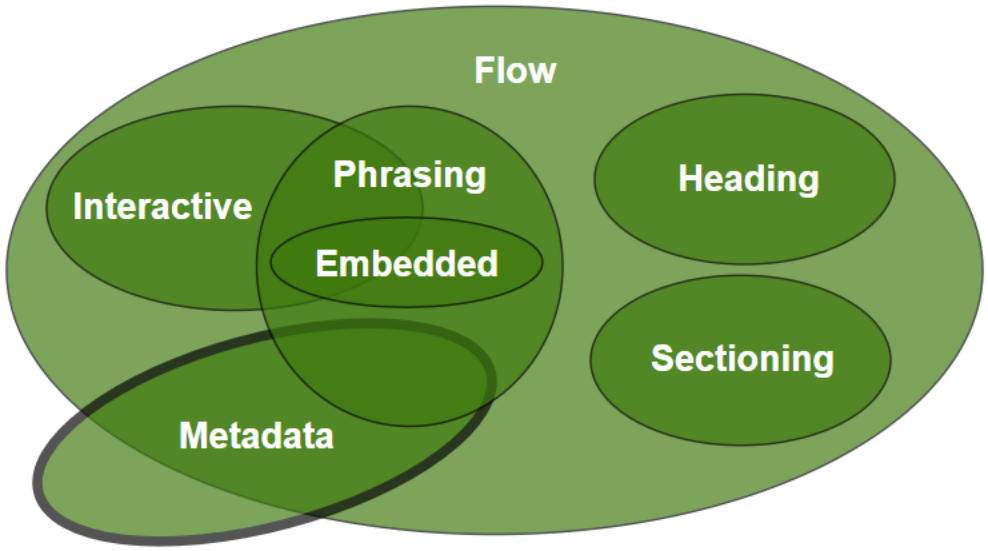
- Metadata
  - Configura la modalità di presentazione e il comportamento del contenuto della pagina
- Flow
  - Il vero contenuto della pagina
- Sectioning
  - Definisce una struttura naturale del documento. W3C: “defin[ing] the scope of headings and footers”
  - Sottoinsieme di flow
- Heading
  - Contiene tutti gli elementi standard HTML 4.0 per heading
  - Sottoinsieme di flow
- Phrasing
  - È il testo del documento
  - Includono elementi per marcare testo interno a paragrafi
- Embedded
  - Importa un’altra risorsa nella pagina (es. audio, video)
- Interactive
  - Elementi con il compito di gestire l’interazione con l’utente

Categorie di contenuti HTML5



https://html.spec.whatwg.org/multipage/dom.html#kinds-of-content

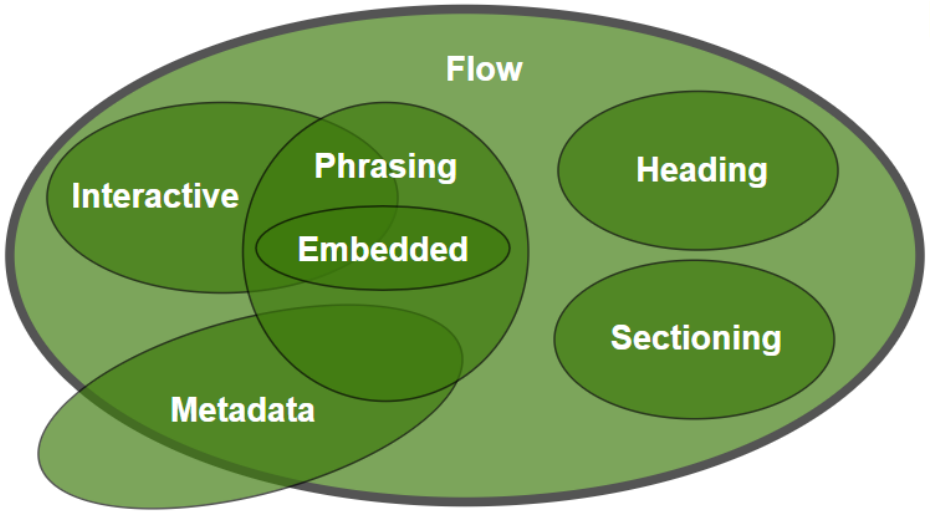
Categorie di contenuti HTML5: Metadata



Metadata content

`base`, `link`, `meta`, `noscript`, `script`, `style`, `template`, `title`

Categorie di contenuti HTML5: Flow

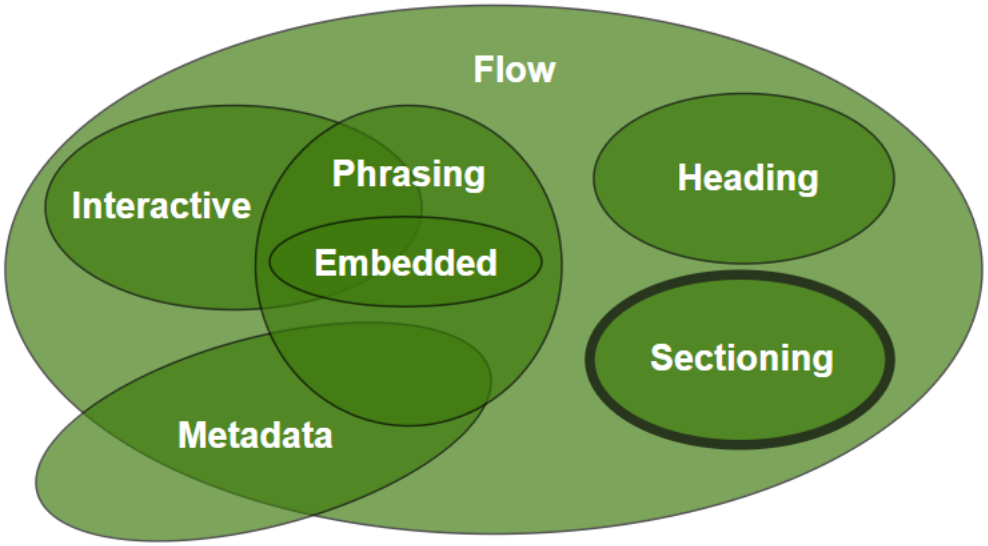


Flow content

a, abbr, address, area*, article, aside, audio, b, bdi, bdo, blockquote, br, button, canvas, cite, code, data, date, datalist, del, details, dfn, dialog, div, dl, em, embed, fieldset, figure, footer, form, h1, h2, h3, h4, h5, h6, header, hgroup, hr, i, iframe, img, input, ins, kbd, keygen, label, link*, main*, map, mark, math, menu, meta*, meter, nav, noscript, object, ol, output, p, picture, pre, progress, q, ruby, s, samp, script, search, section, select, slot, small, span, strong, sub, sup, svg, table, template, textarea, time, u, ul, var, video, wbr, *autonomous custom elements*, *Text**

* *Under certain circumstances (see prose).*

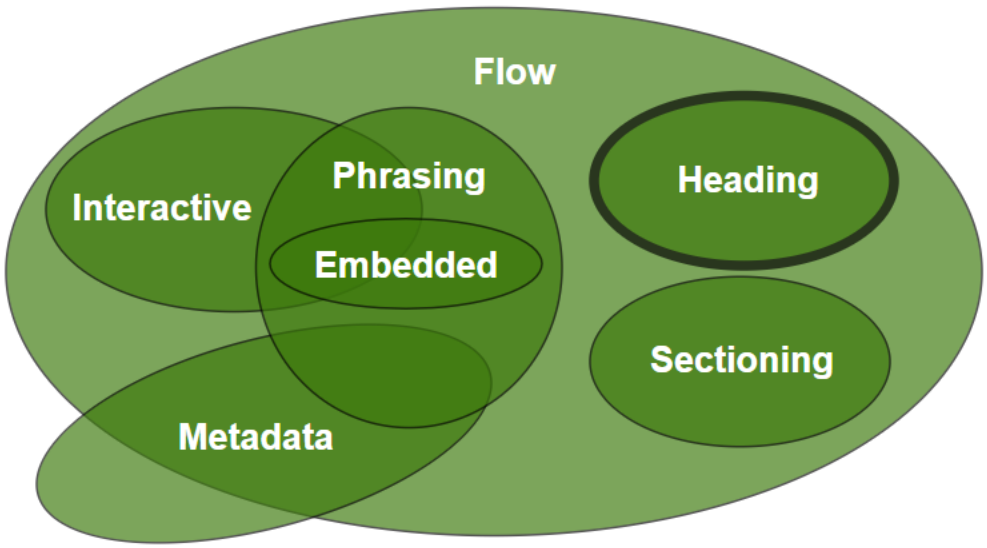
Categorie di contenuti HTML5: Sectioning



Sectioning content

`article`, `aside`, `nav`, `section`

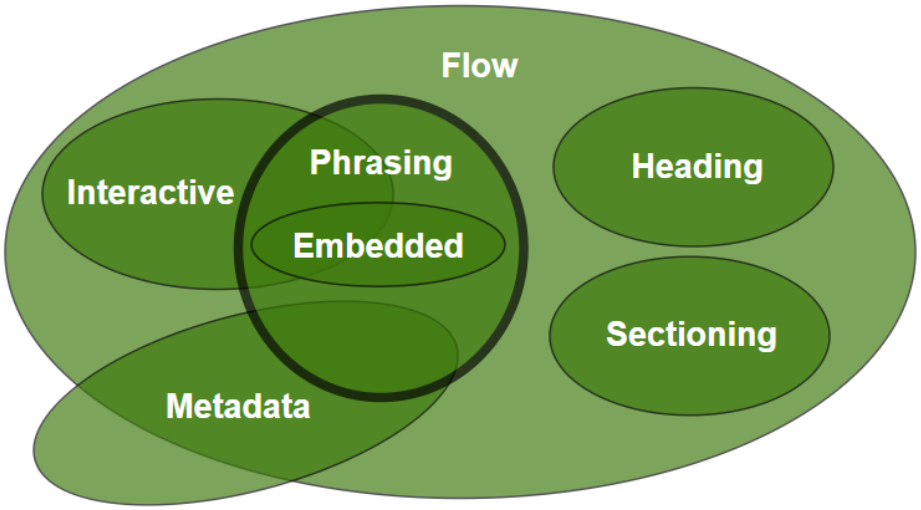
Categorie di contenuti HTML5: Heading



Heading content

h1, h2, h3, h4, h5, h6, hgroup

Categorie di contenuti HTML5: Phrasing

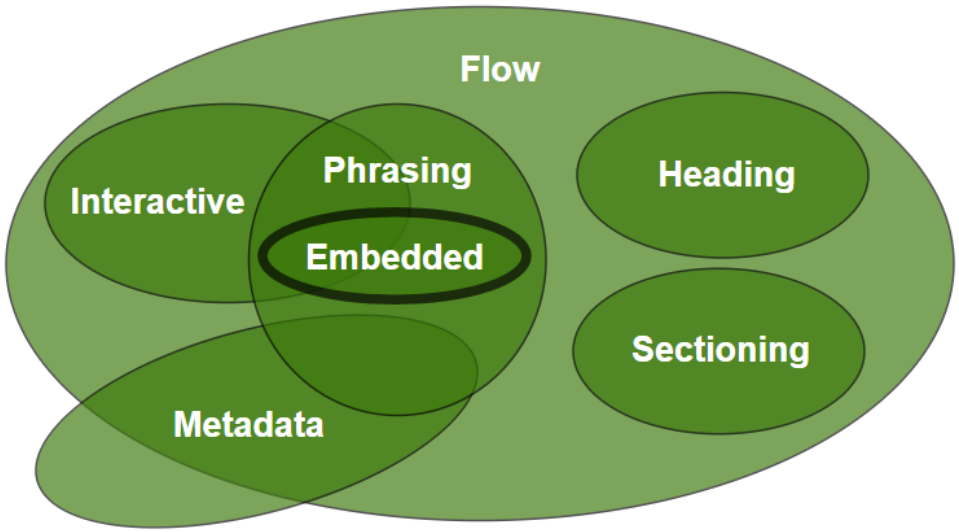


Phrasing content

a*, abbr, area*, audio, b, bdi, bdo, br, button, canvas, cite, code, data, date, datalist, del*, dfn, em, embed, i, iframe, img, input, ins*, kbd, keygen, label, link*, map*, mark, math, meta*, meter, noscript, object, output, picture, progress, q, ruby, s, samp, script, select, slot, small, span, strong, sub, sup, svg, template, textarea, time, u, var, video, wbr, *autonomous custom elements*, *Text**

** Under certain circumstances; see prose.*

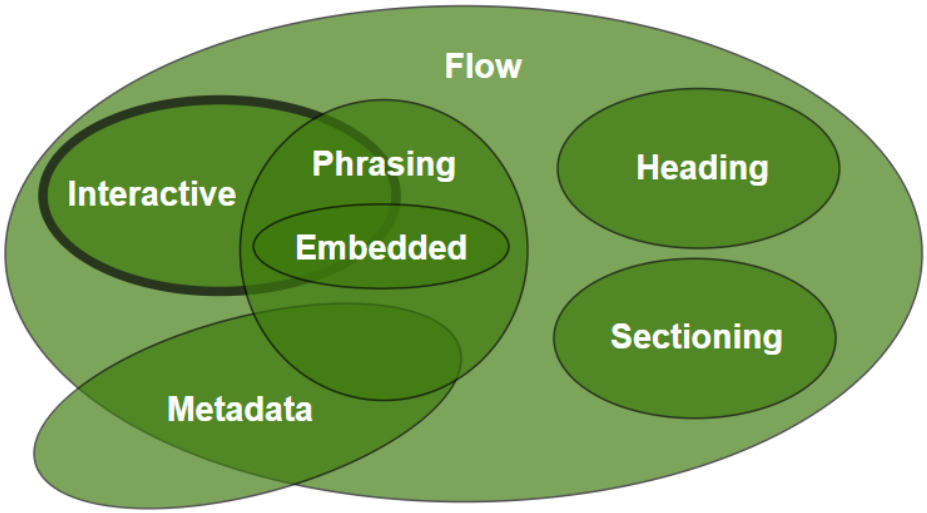
Categorie di contenuti HTML5: Embedding



Embedded content

audio, canvas, embed, iframe, img, math, object, picture, svg, video

Categorie di contenuti HTML5: Interactive



Interactive content

a*, audio*, button, details, embed, iframe, img*, input*, keygen, label, object*, select, textarea, video*

** Under certain circumstances.*

Testa (<head>)

Testa

- Contiene informazioni/metadati relativi al documento
  - Ad esempio, titolo, keyword, altri dati non considerati contenuto del documento
  - Browser di solito non presentano elementi che compaiono nella testa (HEAD)
  - Rendono disponibili queste informazioni attraverso altri meccanismi

Testa: elemento title

- Elemento <title>
- Indica il titolo della pagina
- Viene visualizzato in alto a sinistra nei browser più datati o nella scheda nei browser più moderni

Metadata

- Configura la modalità di presentazione e il comportamento del contenuto della pagina
- Definisce relazioni con altri documenti
- Incluso nell'elemento <head>
- Include elementi come <meta>, <link>, <script>

Metadata

- Elemento <meta>
  - Definisce informazioni relative alla pagina piuttosto che contenuto vero e proprio
  - Include parole chiave o descrizione della pagina
  - Usato dal motore di ricerca per classificare la pagina

- Contiene tre attributi principali
  - Name, content, http-equiv
  - Name contiene il nome della proprietà
    - Author, Description, Copyright, Generator, Language, Keywords
  - Content contiene il valore della proprietà
  - http-equiv specifica caratteristiche dell'HTTP response header

Esempi

<META NAME="author" CONTENT="Pluto">

<META NAME="description" CONTENT="…">

<META NAME="copyright" CONTENT="…">

<META NAME="keyword" CONTENT="…">

<META CHARSET="utf-8">

Meta Tag - Keywords

- Indica alcuni informazioni sul contenuto del sito
- Separate da virgola, punto e virgola o spazio
- Sono chiavi di ricerca

Meta Tag - Keywords

- Motore di ricerca ricava chiavi di indicizzazione
- Attenzione nella scrittura delle keywords
  - Evitare termini generici e particolari
  - Riportare termine italiano e inglese
  - Singolare e plurale
- Evitare di ripetere le stesse parole e usare Keyword astute
- <meta name="keywords" content="calcio soccer tennis basket pallacanestro pallavolo volley inter milan juventus roma lazio…">

Attributo HTTP-EQUIV

- Informazioni sulla comunicazione server-browser
  - Può essere usato al posto dell'attributo name
- Attributo http-equiv fornisce un header HTTP per informazioni/valore dell'attributo content
  - Può essere usato per simulare un HTTP response header
  - HTTP server usano questo attributo per ottenere informazioni circa l'HTTP response header
- Alcuni valori
  - content-type, default-style, refresh
- Esempio
  - <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">

Esempio

- Esempio2.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01
   Transitional//IT">
<html>
        <head>
                <meta http-equiv="Content-Type"
  content="text/html; charset=iso-8859-1">
                <title>Pagina di prova</title>
        </head>
        <body>
                <!-- Scriveremo qui -->
                Hello World!
        </body>
</html>
```

<link>

- HTML
  - <link rel="stylesheet" type="text/css" href="stylefile.css">

- HTML5
  - Rimuove contenuto inutile
  - <link rel="stylesheet" href="stylefile.css">

<script>

- Contribuisce a stabilire il layout della pagina e il suo comportamento

- HTML
  - <script type="text/JavaScript" src="scriptfile.js"></script>

- HTML5
  - Rimuove contenuto inutile
  - <script src="scriptfile.js"></script>

Metadata

```
<head>
  <title> titolo </title>
  <meta charset=utf-8>
  <link rel="stylesheet" href="styles.css" media="all">
  <script src= "jquery-1.4.2.min.js"></script>
</head>
```

Corpo (<body>)

Corpo

- Il corpo di un documento contiene il contenuto vero e proprio
- Il browser presenta il contenuto in diversi modi
  - Browser visuali: è una tela su cui disegnare testi immagini colori…
  - Agenti audio: lo stesso contenuto è parlato
- Contiene tutte le informazioni per la visualizzazione della pagina
  - Siccome gli style sheet sono il modo suggerito per specificare come verrà presentato un documento gli attributi del BODY sono stati deprecati

Tag interni al body

Elementi inline vs block

- Alcuni elementi HTML nel body sono "block-level" altri "inline" ("text level")

- Content model
  - Elementi block-level possono contenere elementi inline e altri elementi block-level
  - Elementi inline contengono solo dati e altri elementi inline

- Formatting
  - Elementi block-level sono formattati in maniera differente dagli elementi inline
  - Elementi block-level iniziano su una linea nuova, elementi inline no

Testo – Tag predefiniti

- Tag-contenitori di testo (elementi block-level, unica eccezione)
  - <h1>, <h2> … (h = heading)
  - <p>
  -
  -

Titoli - Heading

- Sono previste sei grandezze predefinite per i titoli
- Si va da <h1> che è la grandezza maggiore fino ad <h6> che è la grandezza minore
- Sintassi: <h1>Titolo</h1>

Esempio

- <h1>Titolo1</h1>
- <h2>Titolo2</h2>
- <h3>Titolo3</h3>
- <h4>Titolo4</h4>
- <h5>Titolo5</h5>
- <h6>Titolo6</h6>

- Esempio6.html

Paragrafo

-
- Unità base di suddivisione del testo
- Sintassi: paragrafo
- lascia una riga vuota prima e dopo il testo

Blocco testo

- è un blocco contenitore
- Blocco testo va a capo ma non lascia righe di spazio
- Sintassi: blocco 1
- Non impone rappresentazioni di default (a parte il fatto di essere un elemento block-level)

Contenitore

-
- Contenitore generico
- Può essere annidato
- Elemento inline (continua sulla stessa riga del tag che lo contiene (ad es.))
  - Non va a capo
- Non impone rappresentazioni di default (a parte il fatto di essere un elemento inline)

Esempio

- Pagina contenente un titolo, un paragrafo, un blocco di testo e un contenitore

```
…
<h2>Titolo della pagina</h2>
<p>paragrafo 1</p>
<p>paragrafo 2</p>
<div>blocco 1</div>
<div>blocco 2
   <span>contenitore 1</span>
   <span>contenitore 2</span>
</div>
```

- Esempio7.html

Testo - Stili

- Due tipi di stili
  - Fisici: definiscono lo stile grafico
  - Logici: forniscono informazioni sul ruolo svolto dal testo
    - Opzionale: impongono uno stile grafico

Stili fisici - 1

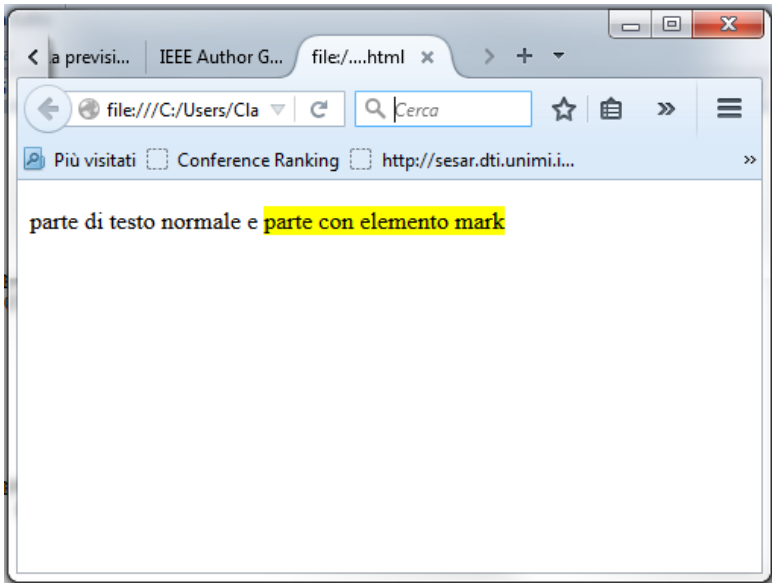
- (<b> prima di HTML5) bold: formatta testo in **grassetto**
- (<i> prima di HTML5) italic: formatta il testo in *corsivo*
- <u> underline: <u>sottolinea</u> il testo
- <strike>: testo barrato (usato per correzioni)
- Tutti elementi inline

Stili fisici - 2

- <sup> apice: $E=mc^2$
- <sub> pedice: H_20
- Utili per la scrittura di formule matematiche

Stili fisici – 3

- <mark>: rappresenta una parte di testo in un documento marcato o sottolineato per riferimento
- Identifica parti rilevanti di testo (parte di categoria Phrasing)
- Usato per portare l'attenzione di un utente su quella parte di testo



Stili logici - esempi

- <abbr> abbreviazione: non comporta nessun cambiamento grafico
- <address> fornisce informazioni di contatto per l'intera pagina (o parti di essa, es. article), <cite> citazioni: testo in corsivo
- <samp> esempio: testo a spaziatura fissa
- <small> per testi piccoli e inoltre note legali, diritti d'autore e piccole note a margine
- Esempio8.1.html

Testo - Font

- Font = colore, dimensione e tipo di carattere

- Carattere predefinito Times new roman
  - Poco leggibile
  - Meglio Verdana, Arial o Helvetica

Esempio

- testo

- E' buona norma:
  - Utilizzare caratteri sicuri (sicuramente visualizzabili)
  - Non indicare un solo carattere
  - Solo come ultima spiaggia sarà usato il Times

Testo - Font

- Scegliere il colore del testo
- testo blu
- La scelta del colore e del carattere può essere fatta nello stesso tag
- testo blu

Font annidati

```
<font color=“blue” face=“verdana”>
        testo blu
        <font color=“red” >
                testo rosso
        </font>
</font>
```

Dimensione del testo

- Attributo size del tag font

- Due modi per impostare la dimensione
  - Valore tra 1 e 7
    - Dimensione 3
  - Valori relativi alla dimensione base del font
    - Dimensione +2
    - Dimensione del font di base
    - Tag <basefont> permette di cambiare il font di base

- Esempio8.2.html

Elenchi

- Elenchi ordinati
- Elenchi non ordinati
- Elenchi di definizioni

Sintassi

<elenco>

<elemento> primo elemento

<elemento> secondo elemento

</elenco>

- Chiusura elemento opzionale

- Sintassi degli elenchi di definizione leggermente diversa

Elenchi ordinati

- Numerazione degli elementi
- Numerazione progressiva
- Tag elenco ordinato <ol> (ordered list)
- Tag elemento <li> (list item)

Esempio elenchi ordinati

```
<div>Titolo</div>
<ol>
        <li>Primo elem prima tabella
        <li>Secondo elem prima tabella
        <ol>
            <li> Primo elem seconda tabella
            <li> Secondo elem seconda tabella
        </ol>
</ol>
```

- Esempio9.html

Elenchi non ordinati

- Nessuna numerazione
- Elenchi puntati
- Tabella tag <ul> (unordered list)

Esempio elenchi non ordinati

```
<ul>
        <li> Primo elemento
        <li> Secondo elemento
        <ul>
                <li> Primo elemento
                <li> Secondo elemento
        </ul>
</ul>
```

- Esempio11.html

Elenchi di definizioni

- Elenco di definizione <dl> (definition list)
- Termine da definire <dt> (definition term)
- Definizione del termine <dd> (definition description)

Elenchi di definizioni - Esempio

```
<p> Titolo
<dl>
        <dt>&ltp&gt
                <dd>apertura paragrafo
        <dt>&ltdiv&gt
                <dd>apertura blocco di testo
        <dt>&ltspan&gt
                <dd>apertura elemento inline
        Altri tag…
<dl>
```

- Esempio13.html

Struttura della pagina web

Struttura della pagina

- Creazione della struttura della pagina ha subito diverse evoluzioni negli ultimi anni
  - Fondamentale nella creazione di layout
  - Struttura basata su tag
  - Struttura basata su tabelle – DEPRECATO
  - Struttura basata su sectioning
  - Struttura semplificata con framework (es. bootstrap) – IN LABORATORIO

Struttura della pagina con tag

- È uno tra i tag più utilizzati nella creazione di pagine Web, soprattutto nella creazione di layout
- Fornisce un vero e proprio elemento strutturale della pagina
- Suddivide gli spazi in zone per progettare il sito in modo semplice e dettagliato
- Molto utile quando usato insieme a fogli di stile

Struttura della pagina con tag

- Facciamo subito un esempio: immaginando di dover costruire un sito pensiamo a come esso debba essere strutturato, tipicamente abbiamo
  - Un contenitore (container)
  - Una parte alta (header)
  - Un corpo centrale (middle)
  - Un menù (navigation)
  - Una parte bassa (footer)
  - Spesso anche un pannello laterale (sidebar)

Struttura della pagina con tag

```
<div id="container">
   <div id="header">
      <div id="navigation"></div><!--#navigation-->
   </div><!--#header-->
   <div id="main"></div><!--#main-->
   <div id="sidebar"></div><!--#sidebar-->
   <div id="footer"></div><!--#footer-->
</div><!--#container-->
```

Struttura della pagina con tag

div-senza-id.html

Sectioning

- Nuova categoria di HTML5
- Include quattro elementi
  - <article>, <aside>, <nav>, e <section>
  - https://www.w3.org/TR/html5/dom.html#sectioning-content-0
- Definizione W3C: “defin[ing] the scope of headings and footers”
  - È un sottoinsieme di flow
- Definisce una struttura naturale del documento

Sectioning

- <article>: reppresenta una parte di contenuto indipendente (ad es., blog entry, articolo di giornale)
- <aside>: una parte di contenuto che non è in stretta relazione con il resto della pagina
- <nav>: una sezione del documento intesa per la navigazione
- <section>: reppresenta un documento o una sezione dell'applicazione

<article>

- <article>: il web contiene un'infinità di articoli di news e entry di blog
- W3C definisce un element article invece di
  - <article>
    ...
    </article>
- Article pensato per contenuto che può essere distribuito
  - News o blog entry che possiamo condividere in feed RSS
- Elemento article può essere innestato in un altro elemento article
- Elemento article non significa solo contenuto di un articolo
  - Può contenere elementi header e footer
  - Ad esempio, header rappresenta il titolo dell'articolo
- Si possono aggiungere elementi article dentro a elementi section

<nav>

- <nav>: rappresenta un blocco di navigazione
- Raggruppa i collegamenti ad altre pagine o parti della pagina corrente
- <nav> non deve essere utilizzato per ogni link
  - Ad esempio, footer spesso contiene link a “terms of service”, “copyright page” e molto altro
- <nav>

  home
  Contact us

  </nav>

<aside>

- <aside>: sezione in relazione con il contenuto principale, ma separato dal contenuto stesso

- <article>
   main content
  <aside>
  contenuto in relazione
  </aside>
  </article>

<section>

- Raggruppamento tematico
- Di solito contiene un heading
- <section>
  <h1>Title</h1>
  One article can have multiple sections
</section>

<header>

- <Header>: specifica un'intestazione per il documento o una sezione
  - <header>
    <h1>Title</h1>
    <nav>

        home
        Contact us

    </nav>
    <header>
- Elemento header può essere anche usato come intestazione di una entry di blog o un articolo di news
  - Elemento article ha un titolo e una data e ora di pubblicazione

<footer>

- <footer>: simile a header
- Elemento footer è spesso riferito come footer di una pagina web
- Nonostante questo possiamo avere un footer per ogni documento, sezione o articolo
- <article>
   main content
  <aside>
  contenuto in relazione
  </aside>
  <footer>
  …
  </footer>
  </article>

Struttura della pagina con tag

Struttura-html5.html

Pagina web complete

- Tre esempi equivalenti (ma non troppo)
  - Primo fatto con solo div e con id che hanno nomi poco significativi (id che puntano a CSS) - div-senza-id.html
  - Secondo uguale al primo ma con id che hanno nomi significativi uguali agli elementi semantici di HTML5 (ad es., id="section") - div-con-id.html
  - Terzo fatto con elementi HTML5 (section, article…)
    - Elementi HTML5 sopra sono solo semantici possono essere sostituiti da elementi vecchi tipo
    - Struttura-html5.html

Ipertestualità (Link)

Introduzione

- Testi vs ipertesti
- Ipertesti caratteristica che ha reso grande il web
- Link = ponte tra un testo e un altro

Link

- I link sono formati da due componenti
  - Il contenuto (testo o immagine) che nasconde il collegamento
  - La risorsa puntata

Link

- Sintassi
  - Clicca qui per visualizzare il collegamento.
  - La testa del link è *qui*
  - La coda è *indirizzo*
  - Contiene nuovi attributi come download, media
- Coda
  - Pagina HTML, immagine, documenti, altri file

Destinazioni

- Pagina HTML: aperta nel browser
- Immagine (.gif, .jpeg): aperta nel browser
- Altri documenti (.doc, .pdf):
  - Visualizzata nel browser
  - Richiesta di salvataggio
- Altri file (.zip, .exe) richiesta di salvataggio

Esempio

- Link a pagina HTML
  - Clicca qui
- Link a documenti
  - Clicca qui
- Link ad immagini
  - Clicca qui
- Link ad altri file
  - Clicca qui

Un tipo particolare di link

- Link ad un indirizzo email
- Sintassi
  - Mandami un'email
- Esempio15.html

Caratteristiche link

- Diversi stati
  - Link a riposo
    - Colore blu #0000FF
  - Link visitato
    - Colore violetto
  - Link attivo
    - Passaggio da una pagina all'altra
    - Utile nel passato
  - Link al passaggio del mouse
    - Solo con fogli di stile

Manipolazione dei link

- Attributi link, alink, vlink del tag body
- <body link="red"> cambia il colore dei link
- <body vlink="green"> cambia il colore dei link visited
- <body alink="yellow"> cambia il colore dei link attivi
- Esempio link classici: Esempio15.html
- Esempio link personalizzati: Esempio16.html

Percorsi dei link

- Percorsi assoluti
  - Viene indicato il percorso per esteso
- Percorsi relativi
  - Fanno riferimento alla posizione del file a cui si riferisce al link a partire dalla posizione della pagina che stiamo sviluppando

Percorsi assoluti

- Leggi l'esempio qui
- Nel caso di file sull'hard disk
- Leggi l'esempio qui

Percorsi relativi

- Clicca qui
  - Indica al browser di cercare il file nella stessa directory
- Clicca qui
  - Indica al browser di cercare il file nella sottodirectory sottodir
- Clicca qui
  - .. Indica al browser di cercare il file nella directory padre
- Esempio18.html

Considerazioni

- Nomi dei file
  - Evitare gli spazi (mio_file.html)
  - Ricordarsi che maiuscole e minuscole fanno la differenza
- non funziona quando caricate la pagina sul vostro sito
- Allora diventerà
  -

Link interni o ancore

- Permette di creare un indice
- Ancora
  - àncora
- Riferimento all'àncora
  - vai all'àncora
- Riferimento ad inizio pagina
  - torna su
- ancora.html

Colorare i link

- Colorare solo un link
- Annidare il font all'interno del tag del link
-

  link verde

- AttributiLink.html

Immagini

- Il web non è solo ipertesto ma bensì ipermedia

- Sintassi: <img src="pic.jpg" alt="description" width="128" height="128">
  - Attributo src: sorgente del file (percorso di memorizzazione file)
  - Attributo alt: descrizione mostrata al passaggio sull'immagine
  - Attributi width e height: dimensioni immagine

- Immagine come link
  -
    <img src="pic.jpg">

Tabelle (table)

Tabelle

- Definizione tramite l'elemento <table>
- Ogni riga definita con il tag <tr> contenuto in table
- Ogni cella definita con il tag <td> contenuto in <tr>
- Una riga può essere anche divisa in table heading usando l'elemento <th>
  - Di solito mostrati centrati e in grassetto
- Elemento <caption> per definire la caption della tabella

Esempio

```
<table style="width:100%">
  <tr>
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
  <tr>
    <td>Eve</td>
    <td>Jackson</td>
    <td>94</td>
  </tr>
  <tr>
    <td>John</td>
    <td>Doe</td>
    <td>80</td>
  </tr>
</table>
```

- Esempiot1.html

Bordi

- Attributo border
  - Se non specificato nessun bordo
  - <table border="1" style="width:100%">

- Esempiot2.html

Celle collassate in una riga/colonna

- Attributo rowspan
  - <th rowspan="2">Telephone</th>
  - <td rowspan="2">Telephone</th>
- Attributo colspan
  - <th colspan="2">Telephone</th>
  - <td colspan="2">Telephone</th>
- Esempiot3.html

Form

Introduzione

- Form usate per collezionare input utente
- Basato sull'elemento <form>
- Specifica diversi elementi
  - input, checkbox, radio button, submit button...

Elemento input

- Elemento più importante
- Diverse varianti che dipendono dall'attributo type
  - Text: definisce input testo normale
  - Radio: definisce un input radio button (per selezionare una tra diverse scelte
  - Submit: definisce un bottone submit (per sottomettere la form)

Text input

- <input type="text">
  - Definisce un campo testo di una linea
  - Esempiof1.html
  - <input type="password"> un tipo speciale di campo testo dove il valore inserito viene nascosto

```
<form>
  First name:<br>
  <input type="text" name="firstname">
  <br>
  Last name:<br>
  <input type="text" name="lastname">
</form>
```

Text area

- <textarea>
  - Definisce un campo testo a linee multiple
  - Esempiof1.html

```
<textarea name="message" rows="10" cols="30">
 The cat was playing in the garden.
</textarea>
```

Radio button input

- <input type=“radio">
  - Definisce un radio button che permette **una** scelta tra diverse opzioni
  - Esempiof2.html

```
<form>
  <input type="radio" name="sex" value="male"
checked>Male
  <br>
  <input type="radio" name="sex" value="female">Female
</form>
```

Checkbox input

- <input type="checkbox">
  - Definisce una serie di checkbox che permettono una scelta **multipla** tra diverse opzioni
  - Esempiof2.html

```
<form>
  <input type="checkbox" name="vehicle1" value="Bike"> I have a bike
  <br>
  <input type="checkbox" name="vehicle2" value="Car"> I have a car
</form>
```

Drop-down list

- <select>
  - Definisce un menù a tendina
  - Elemento <option> definisce le possibili opzioni
    - <option value="fiat" selected>Fiat</option>
  - Esempiof3.html

```
<select name="cars">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="fiat">Fiat</option>
  <option value="audi">Audi</option>
</select>
```

Submit button

- <input type="submit">
  - Definisce un bottone per sottomettere la form a un handler
  - Handler è una pagina lato server che processa i dati della form
  - Handler è definito con l'attributo action
  - Esempiof4.html

```
<form action="action_page.php">
  First name:<br>
  <input type="text" name="firstname" value="Mickey">
  <br>
  Last name:<br>
  <input type="text" name="lastname" value="Mouse">
  <br><br>
  <input type="submit" value="Submit">
</form>
```

- Permette anche di eseguire semplici azioni
  - <button type="button" onclick="alert('Hello World!')">Click Me!</button>

Action attribute

- Definisce le azioni che devono essere fatte quando la form è inviata
- Sottomissione fatta solitamente con il bottone submit
- La pagina viene inviata a una pagina web lato server
  - Ad esempio uno script lato serve gestisce la form inviata
  - <form action="action_page.php">
  - Se l'attributo action è omesso si considera la pagina corrente di default

Method attribute

- Specifica l'operazione HTTP (GET o POST) da usare per inviare la form
  - <form action="action_page.php" method="get">
  - <form action="action_page.php" method="post">
- GET (default) usata per invio di form passive (ad es., search engine query) e senza informazioni sensibili
- I dati inviati tramite la form sono visibili nell'indirizzo
  - action_page.php?firstname=Mickey&lastname=Mouse
- GET è ottima per piccolo quantità di dati

Method attribute

- Specifica l'operazione HTTP (GET o POST) da usare per inviare la form
  - <form action="action_page.php" method="get">
  - <form action="action_page.php" method="post">
- GET (default) usata per invio di form che aggiornano dati
- POST fornisce maggiore sicurezza perchè i dati non sono visibili nell'indirizzo (ad es., password)

Name attribute

- Ogni campo input deve avere un nome

```
<form action="action_page.php">
  First name:<br>
  <input type="text" name="firstname" value="Mickey">
  <br>
  Last name:<br>
  <input type="text" name="lastname" value="Mouse">
  <br><br>
  <input type="submit" value="Submit">
</form>
```

Fieldset

- Raggruppa campi in una form tramite elemento <fieldset>
- Si può definire una caption per la form <legend>

- <form action="action_page.php">
  <fieldset>
  <legend>Personal information:</legend>
  First name:
  <input type="text" name="firstname" value="Mickey">

  Last name:
  <input type="text" name="lastname" value="Mouse">

  <input type="submit" value="Submit">
  </fieldset>
  </form>

- Esempiof5.html

Altri attributi

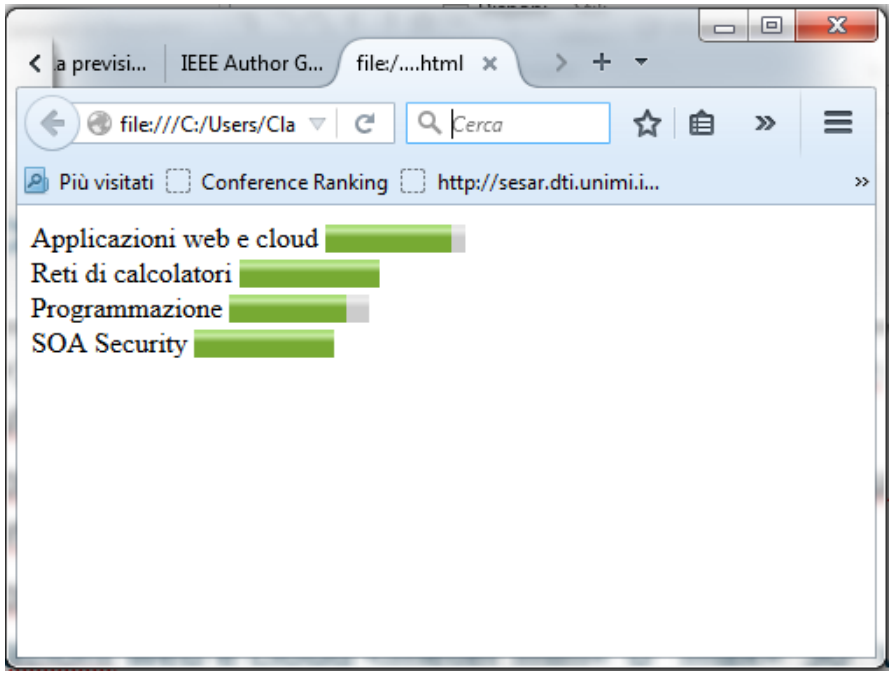
- Attributo value specifica il valore iniziale
  - <input type="text" name="firstname" value="John">
- Attributo readonly specifica che il valore non può essere cambiato
  - <input type="text" name="firstname" value="John" readonly>
- Attributo disabled specifica un campo disabilitato e non utilizzabile (non viene inviato)
  - <input type="text" name="firstname" value="John" disabled>

Altri attributi

- Attributo size specifica la dimensione in caratteri del campo
  - <input type="text" name="firstname" value="John" size="40">
- Attributo maxlength specifica la dimensione massima del contenuto del campo
  - <input type="text" name="firstname" maxlength="10">

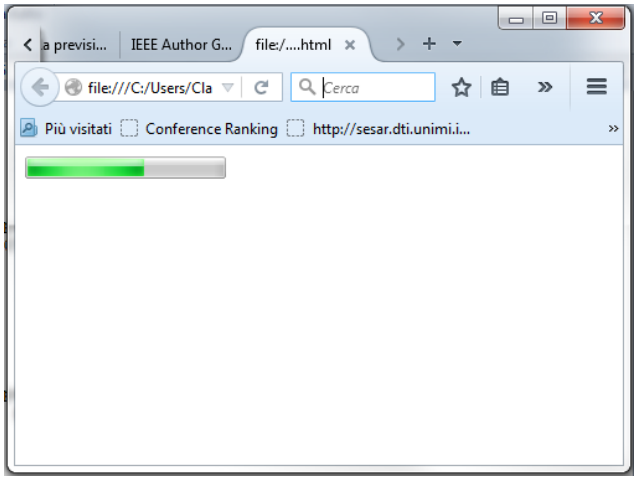
Altri elementi: <meter>

- <meter>: nuovo elemento che rappresenta il valore di un range conosciuto come un misura
- Range conosciuto significa che si può usare solo quando si conoscono i valori minimo e massimo
- Valutazione a esami da 0 a
  - Applicazioni web e cloud <meter min="0" max="30" value="27"> </meter>



Altre elementi: <progress>

- <Progress>: simile all'elemento meter
- Creato per indicare il progresso di un determinato task
- Il progresso è determinato (conosco quanto rimanane del lavoro) o non determinato (non conosco quanto rimane del lavoro)
- Task A
  - <progress value="60" max="100"> 60%</progress>



Conclusioni

HTML

- Linguaggio di markup (tag)
- Nessun meccanismo di decisione e iterazione
- Poche regole sintattiche

HTML

- HTML5 è lo standard attuale per strutturare e presentare il contenuto del World Wide Web

- Mira a risolvere le **limitazioni** di HTML4.01 e precedenti che hanno limitato l'evoluzione del web
  - Bisogno di flash/quicktime per eseguire audio/video

  - Impossibilità di memorizzazione dati lato client se non con linguaggi di scripting o altre tecnologie

  - Mancanza di un formato nativo per il disegno
    - Grafica o animazioni fornite come immagini o con Flash, Java, Silverlight

Semantica

- HTML5 aggiunge semantica alle pagine web
- Il browser conosce quale area di un sito web è l'header, il footer, mentre i div possono avere un id con un valore qualsiasi
- Utile anche per engine di ricerca come Google
  - Peso maggiore al contenuto in header e minore a quello in footer
- Sito web navigabile per persone con disabilità
  - Persone con problemi di apprendimento possono dire al loro browser di mettere sempre prima gli article dei navigation

Il problema del rendering

- HTML5 è una tecnologia in evoluzione e in base al browser che si usa si potrebbero avere differenze di visualizzazione sostanziali
- Per assicurare il miglior risultato si ha bisogno delle versioni più recenti dei browser
- Anche browser moderni hanno problemi di rendering eterogeneo
  - Esistono tabelle di compatibilità dei browser (https://html5test.com/, https://www.w3schools.com/tags/ref_html_browsersupport.asp)
  - Ad esempio, browser diversi supportano tag HTML5 diversi
  - Ad esempio, stesso elemento visualizzato in maniera diversa da browser diversi
- Per designer e sviluppatori è importante testare le pagine HTML5 sui diversi browser

Supporto per elementi HTML5

- Fornito dai browser principali
- Nessun supporto causa errori di visualizzazione
  - Block element trattati come inline element
  - Nuovi elementi non visualizzati correttamente
- Possibili contromisure
  - Tabelle di compatibilità dei browser (identificazione browser)
  - CSS (prossime lezioni)
    - header, section, aside, nav, footer, figure, figcaption { display:block;}
  - JS
    - Modernizr JS library (www.modernizr.com)
    - Aggiunge supporto per mostrare elementi HTML5 e identificarne il supporto

Supporto per elementi HTML5: Identificazione

- Identificazione del browser
  - Utilizzo della stringa User Agent di HTTP
  - Dice quale è il browser
  - Ci si riferisce alle tabelle di compatibilità
  - Non è la strategia migliore per identificare il supporto delle funzionalità
- Utilizzo di Modernizr
  - Permette di fare test diretti alla verifica delle funzionalità di un browser attraverso un oggetto Modernizr e un insieme di proprietà booleane (ad esempio, video per la verifica dell'elemento <video>)
  - True elemento supportato, false elemento non supportato

Conclusioni

- Storia del web
- Standard HTML e sua evoluzione
- HTML5: tag e componenti
- Creazione pagine web