



# Introduzione a XHTML

Prof. Ing. Andrea Omicini

II Facoltà di Ingegneria, Cesena

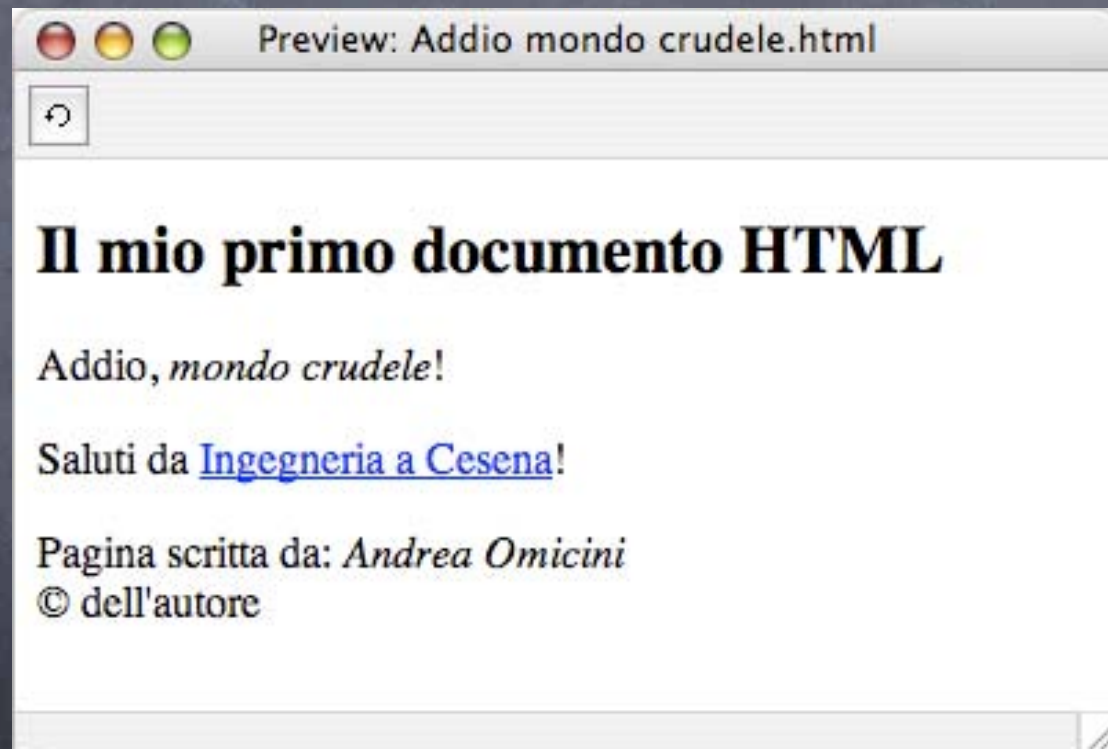
Alma Mater Studiorum, Università di Bologna

[andrea.omicini@unibo.it](mailto:andrea.omicini@unibo.it)

# Addio mondo crudele!

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>Il mio primo documento HTML</title>
</head>
<body>
<h2>Il mio primo documento HTML</h2>
<p>
    Addio, <i>mondo crudele</i>!
</p>
<!-- "Ciao mondo" ci pareva poco -->
<p>
    Saluti da <a href="http://www.ing2.unibo.it/">Ingegneria a Cesena</
a>!
</p>
<p>
    Pagina scritta da: <cite>Andrea Omicini</cite>
    <br />
    &#169; dell'autore
</p>
</body>
</html>
```

# Risultato...



# Cos'è HTML?

- È un linguaggio di markup
  - cioè consente di incorporare nel documento note che fanno riferimento a proprietà del testo
  - soluzione naturale per separare contenuto da proprietà di stampa
  - è un sottoinsieme di SGML
    - Standard Generalized Markup Language
- È una famiglia di standard
  - W3C: consorzio in carico degli standard Web
    - <http://w3c.org/Markup>
  - Si sviluppa nel tempo
    - estensioni ufficiali e proprietarie
    - proposte, raccomandazioni e ratifiche ufficiali

# Versioni

- Da 1.0, 2.0, 3.2, 4.0 a 4.01
  - che è il riferimento ultimo
- XHTML 1.0 è lo standard corrente
  - definito alle differenze con HTML 4.01
  - è XML...
  - XHTML 2.0 è ancora in fase di definizione avanzata
    - ultima a luglio 2004

# Da SGML a HTML

- È un linguaggio di markup molto complicato
  - per il Web ne basta un sottoinsieme
- È anche un meta-linguaggio per markup
  - può essere usato per definire altri linguaggi di markup
  - definendo l'opportuno DTD (Document Type Definition)
- HTML definito con un DTD SGML
  - <http://www.w3.org/TR/html4/sgml/dtd.html>

# Documento

## HTML / XHTML

- Suffisso .html (o .htm)
- File di testo
  - lo posso creare con un qualunque editor di testo
  - con un word processor che consenta salvataggio in formato testo
  - con tool per la creazione di pagine HTML come Composer, Dreamweaver, ecc.
- Come lo vede il computer?
  - Come lo vediamo noi?

# Elementi e Tag

- Un documento HTML contiene
  - elementi e sezioni delimitate da "tag"
- Generalmente, i tag delimitano inizio e fine di una sezione o elemento
  - `<tag>sezione o elemento</tag>`
  - Ovviamente, è utile imparare i principali tag HTML
- I tag possono contenere specifiche dette "attributi"
  - alcuni dei quali sono richiesti, altri opzionali
  - ``
    - src è obbligatorio
    - il tag di chiusura non c'è, allora si usa `/>`
- `<p>` è un tag, `<p>Paragrafo</p>` è un elemento



# Dettagli

- Gli spazi bianchi non hanno significato
  - se non dentro le stringhe
- HTML non è case sensitive
  - `<p>` o `<P>` sono la stessa cosa
- ma XHTML no!
  - `<p>` è giusto, `<P>` sbagliato

# Tipi di tag

- Section tag
- Header tag
- Content tag
- Styling tag
- Empty elements
- Anchor / Hyperlink tags

# Section tag

- dividono il documento in sezioni
- tag radice
  - `<html>` inizia il documento HTML
    - non è detto che il file cominci da lì...
  - `</html>` lo conclude
    - mentre il file può continuare...
- due sezioni: Header e Body

```
<html>  
<head> ... </head>  
<body> ... </body>  
</html>
```

# Header tag

- dentro lo header tra `<head>` e `</head>`
  - non compaiono direttamente nella pagina visualizzata dal browser
- principali
  - `<title>` definisce il titolo della pagina
    - riportato dalla barra del titolo della finestra del browser
  - `<meta />` reca meta-informazione sul contenuto del documento
    - es: `<meta name="author" content="Andrea Omicini" />`
    - è come un commento, ma può essere referenziato e usato

# Content tag

- Stanno nel body tra `<body>` e `</body>`
  - considerati dal browser per la visualizzazione
- Sono la maggioranza degli "utili"
  - `<p>` è un paragrafo
  - `<table>` una tabella
  - `<h1>` uno header di primo livello
    - `<h2>`, `<h3>`, ... di livello successivo
  - commenti
    - `<!-- questo è un commento -->`

# Styling tag

- Due tipi
  - basati sulla natura del contenuto
  - basati sulla formattazione richiesta
- Esempi contenuto
  - `<blockquote>` contiene un blocco di una citazione
  - `<cite>` contiene un riferimento a una citazione
- Esempi formattazione
  - `<b>` grassetto, `<i>` corsivo
- A volte appaiono uguali: non si distingue da browser tra `<em>`, `<cite>`, `<i>`, `<dfn>`
  - ma se leggo il sorgente capisco il ruolo diverso del testo "marcato"

# Empty tag

- In XHTML
  - `<br />` è un'interruzione di linea
  - `<hr />` è una riga orizzontale
  - `<img />` è un'immagine inserita nel testo
- In HTML, `<br>` e `<hr>` vanno bene
  - in XHTML vanno "terminati"
- Attenzione agli attributi
  - obbligatori e opzionali
    - p.e., l'attributo `src` in `img` è obbligatorio

# Anchor / Hypertext tag

- tag `<a>` per entrambe
  - l'ancora serve per dare un nome a porzioni di documento
    - che possono poi essere indirizzati direttamente con `#`
  - il link ipertestuale serve a riferire altri documenti o porzioni di documenti
    - contiene ovviamente un URL
- `<a name="NOME">...</a>`
- `<a href="URL">...</a>`
  - URL relativo o assoluto
  - `<a href="URL#NOME">` fa riferimento a un'ancora



# Limiti di HTML (1)

- Non ben separato il contenuto dalla presentazione
  - l'evoluzione da 1.0 a 4.01 ha migliorato le cose
  - ma troppe deviazioni dettate dai browser
    - che bisogna EVITARE
    - in generale, ma sicuramente in questo corso :)
- Non "ben formato"
  - come è invece XML
  - perdona troppi errori
    - si possono intrecciare elementi, mettere tag sbagliati, dimenticare chiusure, attributi, ecc...

# Limiti di HTML (2)

- È più un linguaggio di markup “strutturale”
  - descrive la struttura del testo
    - structural markup
  - più che la natura del contenuto
    - descriptive / semantic markup
  - quindi fa fatica ad adattarsi alla natura del medium
- Per questo si va verso XML
  - tramite XHTML
- Obiettivo: linguaggio
  - disciplinato e controllabile
  - potente ma semplice
  - descrittivo

# XML in breve

- Linguaggio di markup estensibile per definire linguaggi di markup
- Un linguaggio di markup definito con XML è un'applicazione di XML
  - XHTML è un'applicazione di XML
- Fondamentale: non ha tag o elementi predefiniti
  - si possono creare i tag e le strutture che meglio si adattano ai contenuti desiderati

# Parti fondamentali di un documento XML

1. Documento XML vero e proprio
  - contenuto costruito con le regole XML
2. Document Type Definition (DTD)
  - quali sono i tag e il loro significato
3. Foglio di stile (Style Sheet)
  - per la presentazione

# Benefici di XML

- Portabile
  - formato testo, qualunque piattaforma e tanti applicativi per scriverlo e leggerlo
- Configurabile / Estensibile
  - ciascuno può definire il linguaggio di markup che serve
- Auto-descrittivo
  - un documento XML è auto-contenuto: presentazione, significato, dati e loro struttura

# XHTML = HTML + XML

- XHTML scritto usando XML come meta-linguaggio
  - HTML definito in SGML
- vocabolario HTML ma sintassi XML
  - compatibilità all'indietro
  - proprietà XML
    - ben formato, non pronò a errori, estensibile via XML

# XHTML - HTML = ?

- Prologo XML
  - primo elemento del documento
  - `<?xml version="1.0" encoding="UTF-8"?>`
    - come l'informazione `<meta />` per i vecchi browser
- Dichiarazione di tipo del documento
  - dice qual è il DTD del documento
  - `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">`
  - indirizzo o DTD embedded
- Entrambe prima di `<html>`

# Altre differenze

- Raccomandazione: definire il namespace
  - per dare significato ai tag
  - potrei scrivere

`<http://www.w3.org/1999/xhtml:p>` per il tag paragrafo

- e poi per tutti gli altri...

- ma è più comodo scrivere

```
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
```

- e posso anche definire i miei namespace, e darli nomi
- e in più dichiarare il linguaggio

- Altre

- case sensitivity
- innestamento compiuto
- elementi obbligatori: `<head>`, `<body>`, `<html>`, `<title>`



# Cosa vogliamo imparare in laboratorio?

- Struttura del documento XHTML
  - header, body, e loro contenuto
- Elementi inline, commenti, liste, caratteri speciali
- Attributi
  - comuni alla maggior parte degli elementi
  - attributi per cambiare la presentazione
- Riferimenti ipertestuali
- Immagini
- Tabelle
- Moduli (Form)
- Frame