

Informatica per le discipline umanistiche

Lezione 13 – Hyper Text Markup Language

`cristiano.longo@unict.it`



HTML

Hyper Text Markup Language (HTML) è il formato per le pagine web:

- è un formato di tipo testo;
- il media type corrispondente è **text/html**;
- È una raccomandazione del w3c (attualmente alla versione 5.2)
<https://html.spec.whatwg.org>

Vedi anche

HTML Tutorial (<https://www.w3schools.com/html/>)

Struttura di un documento HTML

Un documento HTML è costituito da

- l'intestazione, che indica che si tratta di un documento HTML 5, e

```
<!DOCTYPE html>  
<html lang="en">  
  <head>  
    <title>Page Title</title>  
  </head>  
  <body>  
    <h1>My First Heading</h1>  
    <p>My first paragraph.</p>  
  </body>  
</html>
```

Struttura di un documento HTML

Un documento HTML è costituito da

- l'intestazione, che indica che si tratta di un documento HTML 5, e
- l'elemento ***radice***.

```
<!DOCTYPE html>  
<html lang="en">  
  <head>  
    <title>Page Title</title>  
  </head>  
  <body>  
    <h1>My First Heading</h1>  
    <p>My first paragraph.</p>  
  </body>  
</html>
```

Elementi

Ogni elemento è caratterizzato da

- Un **tipo**, detto *tagname*, che indica il significato dell'elemento,

```
<!DOCTYPE html>  
<html lang="it">  
  ...  
</html>
```

Elementi

Ogni elemento è caratterizzato da

- Un tipo, detto *tagname*, che indica il significato dell'elemento,
- zero o più **attributi** (coppie *nome="valore"*),

```
<!DOCTYPE html>  
<html lang="it">  
  ...  
</html>
```

Elementi

Ogni elemento è caratterizzato da

- Un tipo, detto *tagname*, che indica il significato dell'elemento,
- zero o più attributi (coppie *nome="valore"*),
- Zero o più **figli**.

```
<!DOCTYPE html>
<html lang="it">
  <head>
    ...
  </head>
  <body>
    ...
  </body>
</html>
```

Tipo di Elemento

Ogni elemento è delimitato da un *tag* di apertura

<tagname [attributi]> ... </tagname>

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Tipo di Elemento

Ogni elemento è delimitato da un *tag* di apertura e un corrispondente tag di chiusura.

`<tagname [attributi]> ... </tagname>`

```
<!DOCTYPE html>  
<html lang="en">  
  <head>  
    <title>Page Title</title>  
  </head>  
  <body>  
    <h1>My First Heading</h1>  
    <p>My first paragraph.</p>  
  </body>  
</html>
```

Tipo di Elemento

Ogni elemento è delimitato da un *tag* di apertura e un corrispondente tag di chiusura.

`<tagname [attributi]> ... </tagname>`

L'elemento radice di un documento HTML è sempre di tipo **html**.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Elementi *void*

I tipi di elementi *void* indicano elementi che non possono avere figli. In questo caso viene indicato solo il tag di apertura.

```
<voidtagname [attributi]>
```

```
...  
: <input type="text" name="username"> :  
: <input type="text" name="password"> :  
: ...
```

Attributi

Nel tag di apertura possono essere specificati degli attributi.

`<tagname [attributi]> ... </tagname>`

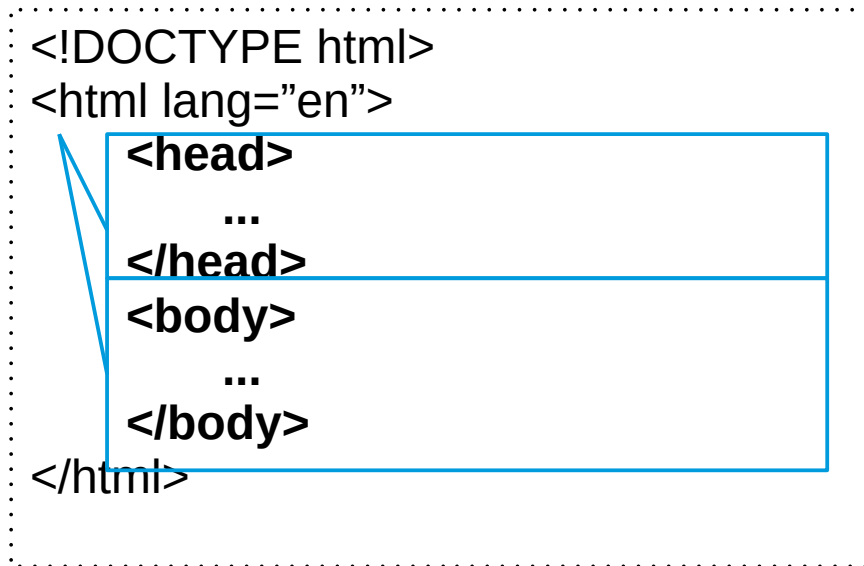
Sono coppie *nome="valore"* che forniscono informazioni aggiuntive sull'elemento.

```
<!DOCTYPE html>  
<html lang="en">  
  <head>  
    <title>Page Title</title>  
  </head>  
  <body>  
    <h1>My First Heading</h1>  
    <p>My first paragraph.</p>  
  </body>  
</html>
```

Contenuto degli elementi

Ogni elemento può avere dei **figli**, che possono essere

- ulteriori elementi,



Contenuto degli elementi

Ogni elemento può avere dei **figli**, che possono essere

- ulteriori elementi,
- nodi di testo.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Struttura di un documento HTML

L'elemento html ha due figli

- **head** che contiene i *metadati* relativi a tutto il documento HTML,

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Struttura di un documento HTML

L'elemento html ha due figli

- head che contiene i *metadati* relativi a tutto il documento HTML,
- **body** per il contenuto vero e proprio, che poi sarà visualizzato dal browser.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

HEAD

L'elemento **head** contiene I metadati relativi a tutto il documento HTML, ad esempio:

- title fornisce un titolo per indicare il documento in contesti in cui il contenuto non sia disponibile, ad esempio nei risultati di una ricerca su internet.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <base href="https://www.example.com/" />
    <meta charset="UTF-8" />
  </head>
  <body>
    ...
  </body>
</html>
```

HEAD

L'elemento **head** contiene I metadati relativi a tutto il documento HTML, ad esempio:

- title fornisce un titolo per indicare il documento in contesti in cui il contenuto non sia disponibile, ad esempio nei risultati di una ricerca su internet.
- base indica una **base URL** da usare per risolvere le URL *relative* (solo path);

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <base href="https://www.example.com/">
    <meta charset="UTF-8">
  </head>
  <body>
    ...
  </body>
</html>
```

HEAD

L'elemento **head** contiene I metadati relativi a tutto il documento HTML, ad esempio:

- title fornisce un titolo per indicare il documento in contesti in cui il contenuto non sia disponibile, ad esempio nei risultati di una ricerca su internet.
- base indica una **base URL** da usare per risolvere le URL *relative* (solo path);
- meta usualmente per metadati che vanno anche inviati come header HTTP.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <base href="https://www.example.com/">
    <meta charset="UTF-8">
  </head>
  <body>
    ...
  </body>
</html>
```

Struttura del body

All'interno del body, gli elementi di tipo h1, h2, h3, h4, h5, h6 vengono usati h1 per il titolo principale, h2 per i titoli delle sezioni, h3 delle sottosezioni, e così via.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    ...
  </head>
  <body>
    <h1>Main title</h1>
    <h2>Section 1</h2>
    ...
    <h2>Section 2</h2>
    ...
    <h3>Section 2.1</h3>
    ...
    <h3>Section 2.2</h3>
  </body>
</html>
```

Struttura del body

All'interno del body, gli elementi di tipo h1, h2, h3, h4, h5, h6 vengono usati h1 per il titolo principale, h2 per i titoli delle sezioni, h3 delle sottosezioni, e così via.

Il tipo di elemento **p** viene usato per rappresentare il contenuto dei *paragrafi*.

```
<!DOCTYPE html>
<html lang="en">
  <head> ... </head>
  <body>
    <h1>Main title</h1>
    <h2>Section 1</h2>
    <p>This is a paragraph.</p>
    <h2>Section 2</h2>
    <p>Another paragraph.</p>
    <h3>Section 2.1</h3>
    <p>Still a paragraph.</p>
    <h3>Section 2.2</h3>
    <p>And so on...</p>
  </body>
</html>
```

Liste

Per le **liste** di elementi si usano

- ul per le liste non ordinate,

```
<!DOCTYPE html>
<html lang="en">
  <head> ... </head>
  <body>
    <ul>
      <li>an item</li>
      <li>another item</li>
    </ul>
    <ol>
      <li>first item</li>
      <li>next item</li>
    </ol>
  </body>
</html>
```

- 
- an item
 - another item

Liste

Per le **liste** di elementi si usano

- ul per le liste non ordinate,
- ol per le liste ordinate,

```
<!DOCTYPE html>
<html lang="en">
  <head> ... </head>
  <body>
    <ul>
      <li>an item</li>
      <li>another item</li>
    </ul>
    <ol>
      <li>first item</li>
      <li>next item</li>
    </ol>
  </body>
</html>
```

-
- an item
 - another item

- 1) first item
- 2) next item

Liste

Per le **liste** di elementi si usano

- ul per le liste non ordinate,
- ol per le liste ordinate,
- li per indicare gli elementi.

```
<!DOCTYPE html>
<html lang="en">
  <head> ... </head>
  <body>
    <ul>
      <li>an item</li>
      <li>another item</li>
    </ul>
    <ol>
      <li>first item</li>
      <li>next item</li>
    </ol>
  </body>
</html>
```

-
- an item
 - another item

- 1) first item
- 2) next item

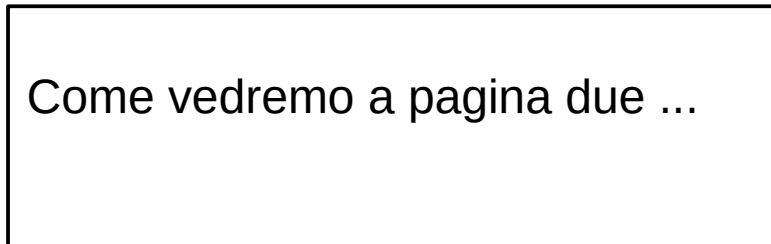
Collegamenti

È possibile associare ad una porzione di testo un collegamento ad una risorsa, esterna o interna al documento, mediante il tipo di elemento **a**. La risorsa collegata viene indicata tramite una URL specificata mediante l'attributo **href**.

Esempio di frammento di codice HTML:

```
<p>Come vedremo a <a href="http://mysite/page2.html">pagina due</a> ...</p>
```

Visualizzazione su browser

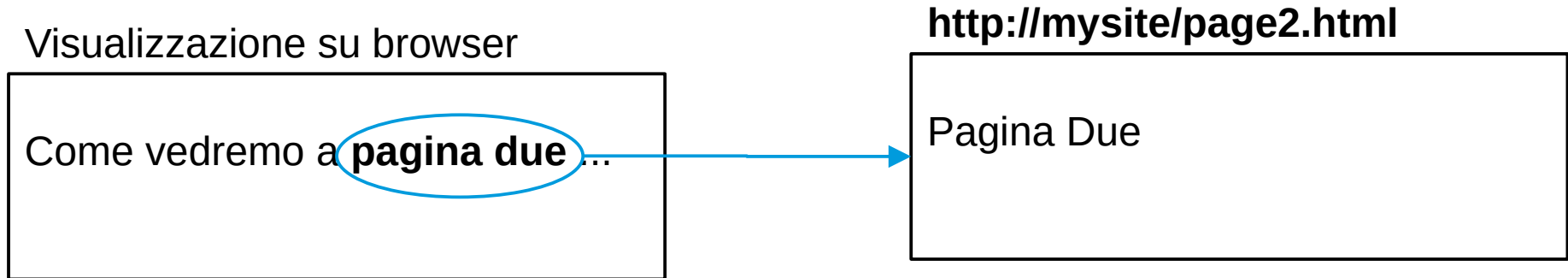


Collegamenti

È possibile associare ad una porzione di testo un collegamento ad una risorsa, esterna o interna al documento, mediante il tipo di elemento **a**. La risorsa collegata viene indicata tramite una URL specificata mediante l'attributo **href**.

Esempio di frammento di codice HTML:

```
<p>Come vedremo a <a href="http://mysite/page2.html">pagina due</a> ...</p>
```



IRI relative

Le IRI *incomplete* vengono considerate relative alla IRI utilizzata per accedere alla pagina oppure a quanto specificato come *base* nell'head.

Ad esempio:

- Sto visitando la pagina disponibile alla IRI <http://info.cern.ch/hypertext/DataSources/>

IRI relative

Le IRI *incomplete* vengono considerate relative alla IRI utilizzata per accedere alla pagina oppure a quanto specificato come *base* nell'head.

Ad esempio:

- Sto visitando la pagina disponibile alla IRI `http://info.cern.ch/hypertext/DataSources/`
- Essa contiene l'elemento `WWW servers`

IRI relative

Le IRI *incomplete* vengono considerate relative alla IRI utilizzata per accedere alla pagina oppure a quanto specificato come *base* nell'head.

Ad esempio:

- Sto visitando la pagina disponibile alla IRI **`http://info.cern.ch/hypertext/DataSources/`**
- Essa contiene l'elemento `WWW servers`

Questo elemento genera un collegamento verso

`http://info.cern.ch/hypertext/DataSources/`

IRI relative

Le IRI *incomplete* vengono considerate relative alla IRI utilizzata per accedere alla pagina oppure a quanto specificato come *base* nell'head.

Ad esempio:

- Sto visitando la pagina disponibile alla IRI `http://info.cern.ch/hypertext/DataSources/`
- Essa contiene l'elemento `WWW servers`

Questo elemento genera un collegamento verso

`http://info.cern.ch/hypertext/DataSources/Available.html`

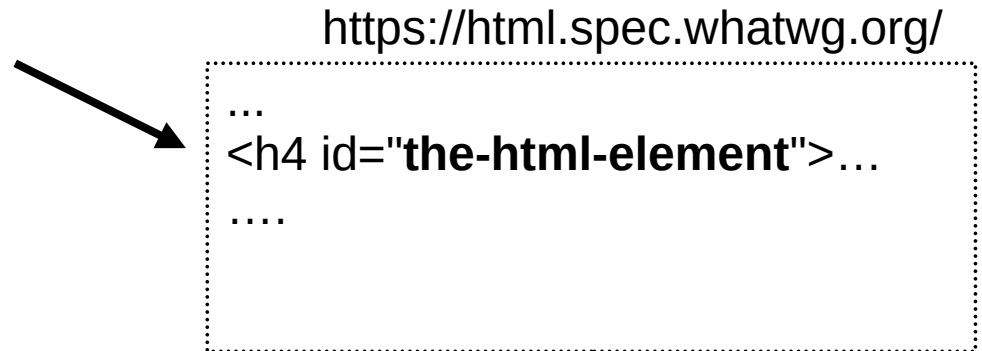
Fragment Identifier

Una IRI può contenere un *fragment identifier* per indicare lo specifico elemento con attributo **id**="fragment-id" all'interno della pagina.

```
<scheme>://[<user>[:<password>]@]<host>[:<port>][/<url-path>][#<b>fragment-id</b>][?<query-part>]
```

Esempio

<https://html.spec.whatwg.org/#the-html-element>



Immagini

Il tipo di elemento **img** permette di inserire immagini (nei formati immagine riconosciuti dal browser).

```

```

Immagini

Il tipo di elemento **img** permette di inserire immagini (nei formati immagine riconosciuti dal browser). La URL dalla quale andare a scaricare l'immagine è specificata mediante l'attributo **src**.

```

```

Immagini

Il tipo di elemento **img** permette di inserire immagini (nei formati immagine riconosciuti dal browser). La URL dalla quale andare a scaricare l'immagine è specificata mediante l'attributo **src**. L'attributo **alt** permette di specificare un testo da mostrare nel caso in cui sia impossibile scaricare o visualizzare l'immagine.

```

```

Enfasi

È possibile enfatizzare alcune porzioni di testo usando i tipi di elemento `em` (enfasi) e `strong` (enfasi *forte*).

```
<p><em>Devi</em> sbrigarti. <strong>Non</strong> possiamo vivere in questo modo.</p>
```

Visualizzazione su browser

Devi sbrigarti. **Non** possiamo vivere in questo modo.

Tabelle

Il tipo di elemento **table** permette di presentare dati in forma tabellare.

```
<table>
```

```
</table>
```

Tabelle

Il tipo di elemento **table** permette di presentare dati in forma tabellare.
Gli elementi interni **tr** rappresentano le righe.

```
<table>  
  <tr>.....</tr>  
  <tr>.....</tr>  
  <tr>.....</tr>  
</table>
```

Tabelle

Il tipo di elemento **table** permette di presentare dati in forma tabellare. Gli elementi interni **tr** rappresentano le righe. Per i contenuti delle *celle* all'interno di ogni riga si specificano elementi di tipo **th**, se la cella è una intestazione,

```
<table>
  <tr>
    <th>Nome</th>
    <th>Cognome</th>
  </tr>
  <tr>.....</tr>
  <tr>.....</tr>
</table>
```

Tabelle

Il tipo di elemento **table** permette di presentare dati in forma tabellare.

Gli elementi interni **tr** rappresentano le righe. Per i contenuti delle *celle* all'interno di ogni riga si specificano elementi di tipo **th**, se la cella è una intestazione, e **td**, per le celle contenenti i dati.

```
<table>
  <tr>
    <th>Nome</th>
    <th>Cognome</th>
  </tr>
  <tr>
    <td>Cristiano</td>
    <td>Longo</td>
  </tr>
</table>
```

Form

Il tipo di elemento **form** permette di definire dei campi di inserimento da compilare a cura dell'utente ed il cui contenuto verrà poi inviato ad un server.

```
<form action="http://mysite.com">  
  <p>Username <input type="text" name="username"></p>  
  <p>Password <input type="password" name="password"></p>  
  <p><input type="submit" name="invia" value="invia"></p>  
</form>
```

Form

Il tipo di elemento **form** permette di definire dei campi di inserimento da compilare a cura dell'utente ed il cui contenuto verrà poi inviato ad un server.

L'attributo `action` indica la URL alla quale saranno inviati i dati.

```
<form action="http://mysite.com">  
  <p>Username <input type="text" name="username"></p>  
  <p>Password <input type="password" name="password"></p>  
  <p><input type="submit" name="invia" value="invia"></p>  
</form>
```

Cascading Style Sheet

Cascading Style Sheet (CSS, <https://www.w3.org/TR/CSS2/>) è un formato che contiene le direttive per il *rendering* degli elementi HTML.

Vedi anche <https://www.w3schools.com/css/>

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

My First Heading

My first paragraph.

Associare un CSS

Le regole CSS possono essere associate ad un documento HTML

- indicando un CSS **esterno**

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <link rel="stylesheet" href="mystyle.css">
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Associare un CSS

Le regole CSS possono essere associate ad un documento HTML

- indicando un CSS **esterno**,
- con l'elemento **style**,

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <link rel="stylesheet" href="mystyle.css">
    <style>
      h1 { color: rgb(255,0,0)}
    </style>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Associare un CSS

Le regole CSS possono essere associate ad un documento HTML

- indicando un CSS **esterno**,
- con l'elemento `style`,
- ad un singolo elemento con l'**attributo style**.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <link rel="stylesheet" href="mystyle.css">
    <style>
      h1 { color: rgb(255,0,0)}
    </style>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p style="color: rgb(0,255,0)">My first
paragraph.</p>
```

....

Associare un CSS

Le regole CSS possono essere associate ad un documento HTML

- indicando un CSS **esterno**,
- con l'elemento `style`,
- ad un singolo elemento con l'attributo `style`.

Fogli di stile

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <link rel="stylesheet" href="mystyle.css">
    <style>
      h1 { color: rgb(255,0,0)}
    </style>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p style="color: rgb(0,255,0)">My first
paragraph.</p>
```

....

Fogli di stile

Un foglio di stile è una sequenza di **regole** CSS

```
h1 {  
    color: rgb(255,0,0)  
}  
  
p {  
    color: rgb(0,255,0);  
    text-align: center;  
}
```

Fogli di stile

Un foglio di stile è una sequenza di **regole** CSS, ognuna costituita da

- un **selettore** che specifica gli elementi ai quali la regola si applica

```
h1 {  
    color: rgb(255,0,0)  
}  
  
p {  
    color: rgb(0,255,0);  
    text-align: center  
}
```

Fogli di stile

Un foglio di stile è una sequenza di **regole** CSS, ognuna costituita da

- un **selettore** che specifica gli elementi ai quali la regola si applica
- Un blocco di **dichiarazioni**

nomeattributo:

```
h1 {  
    color: rgb(255,0,0)  
}  
  
p {  
    color: rgb(0,255,0);  
    text-align: center;  
}
```

Fogli di stile

Un foglio di stile è una sequenza di **regole** CSS, ognuna costituita da

- un **selettore** che specifica gli elementi ai quali la regola si applica
- Un blocco di **dichiarazioni**

nomeattributo: valore

```
h1 {  
    color: rgb(255,0,0)  
}  
  
p {  
    color: rgb(0,255,0);  
    text-align: center  
}
```

Java Script

Una pagina web può contenere dei programmi in linguaggio Java Script, interpretato dal browser.

Vedi anche <https://www.w3schools.com/js/>.

Java Script

Una pagina web può contenere del *codice sorgente* in linguaggio Java Script, interpretato dal browser.

Vedi anche <https://www.w3schools.com/js/>.

Il codice Java Script può essere specificato con elementi di tipo **script**

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <script>
      alert("Javascript-generated");
    </script>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Java Script

Una pagina web può contenere del *codice sorgente* in linguaggio Java Script, interpretato dal browser.

Vedi anche <https://www.w3schools.com/js/>.

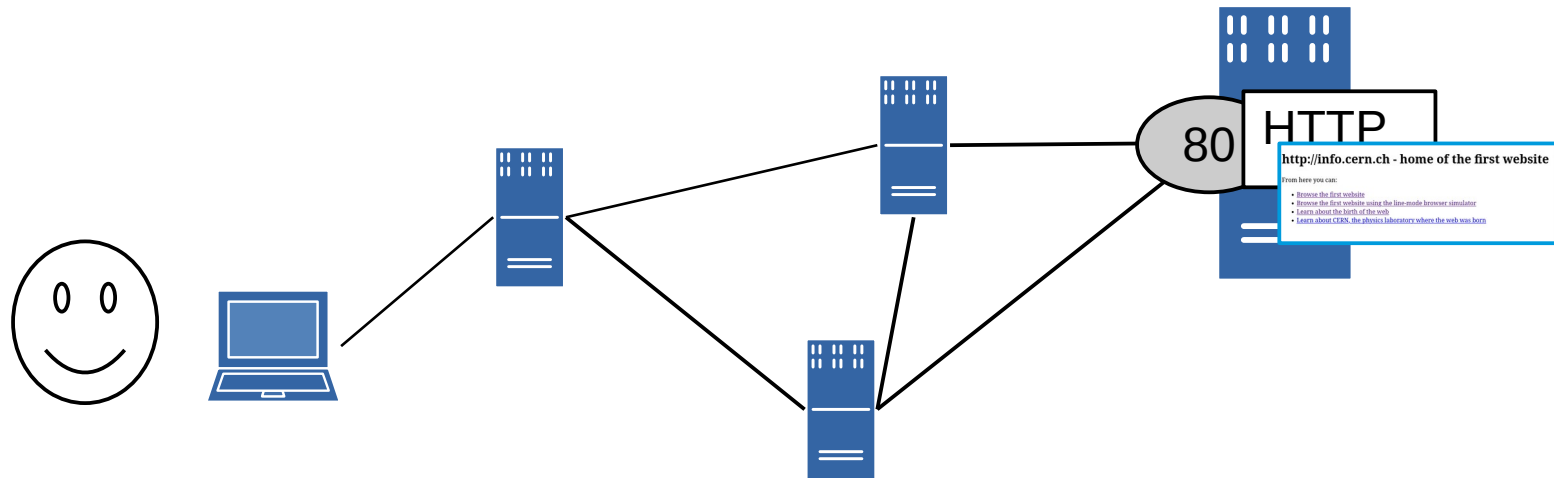
Il codice Java Script può essere specificato con elementi di tipo **script**.

L'attributo **src** di script permette di caricare il java script da una fonte esterna.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Page Title</title>
    <script src="myprog.js"></script>
  </head>
  <body>
    <h1>My First Heading</h1>
    <p>My first paragraph.</p>
  </body>
</html>
```

Riepilogo

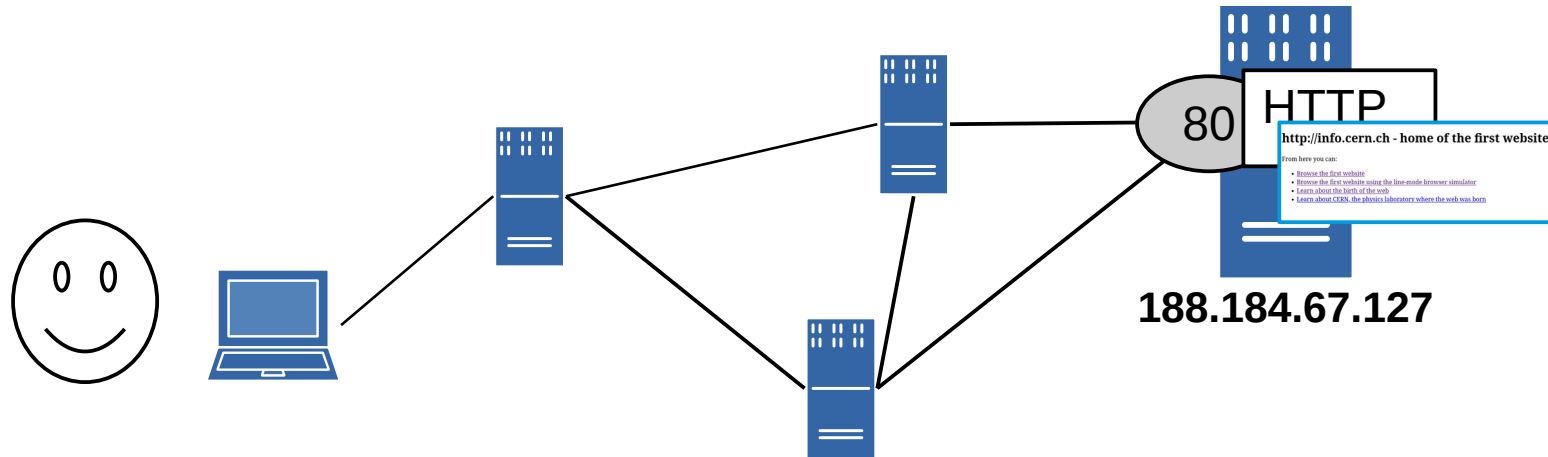
Quali operazioni effettua il browser per visualizzare la pagina <http://info.cern.ch>?



Riepilogo

Quali operazioni effettua il browser per visualizzare la pagina `http://info.cern.ch`?

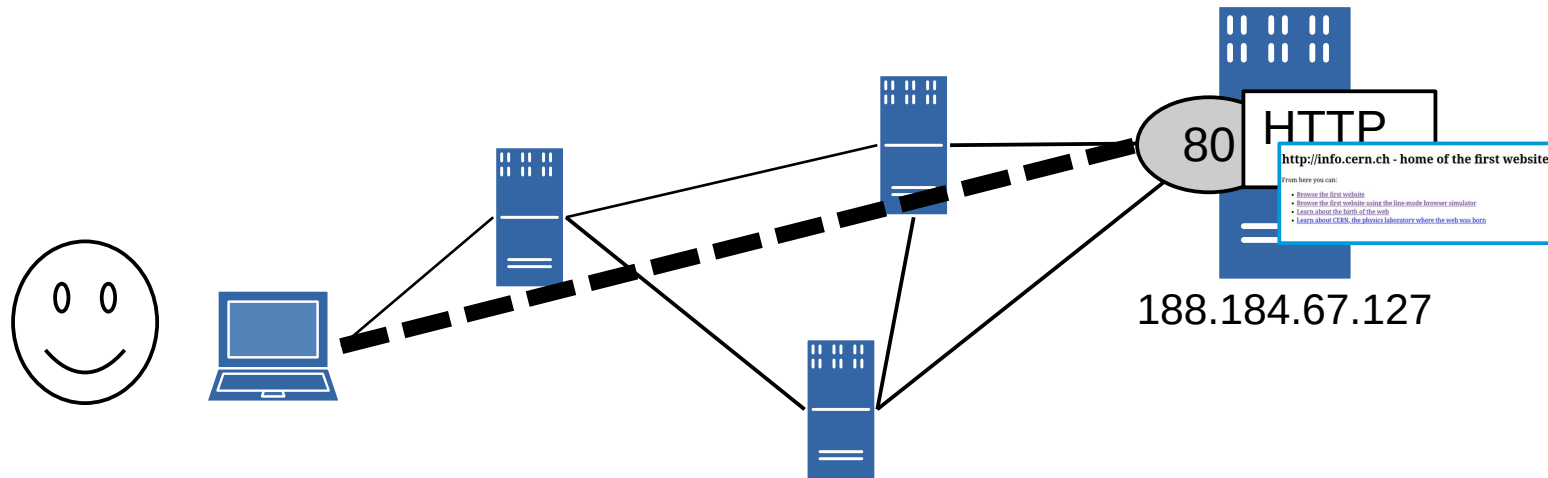
1) Risolvere l'indirizzo IP corrispondente all'hostname;



Riepilogo

Quali operazioni effettua il browser per visualizzare la pagina `http://info.cern.ch`?

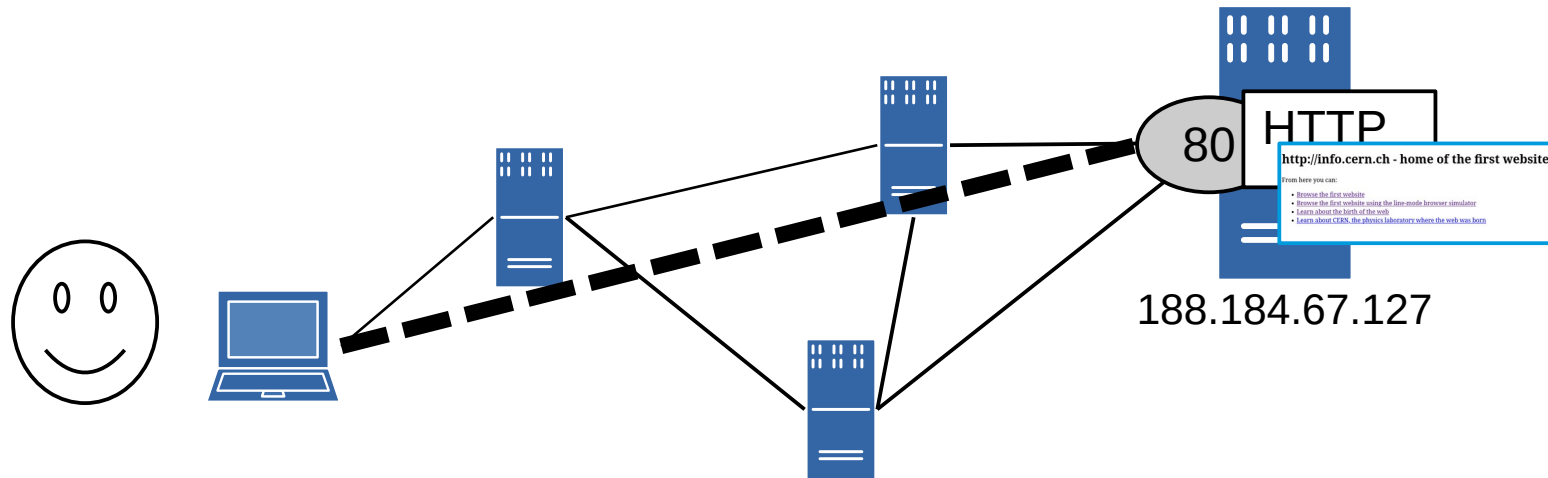
- 1) Risolvere l'indirizzo IP corrispondente all'hostname;
- 2) iniziare una connessione TCP/IP sulla porta associata al protocollo;
- 3) Richiedere la pagina.



Riepilogo

Quali operazioni effettua il browser per visualizzare la pagina `http://info.cern.ch`?

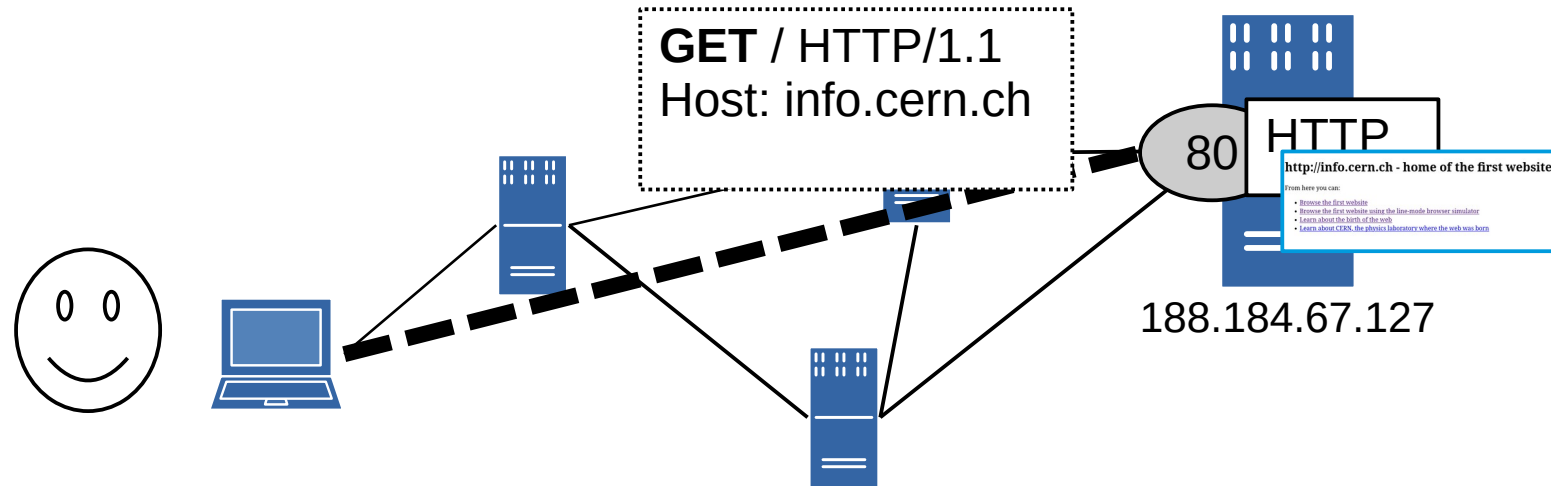
- 1) Risolvere l'indirizzo IP corrispondente all'hostname;
- 2) iniziare una connessione TCP/IP sulla porta associata al protocollo;



Riepilogo

Quali operazioni effettua il browser per visualizzare la pagina `http://info.cern.ch`?

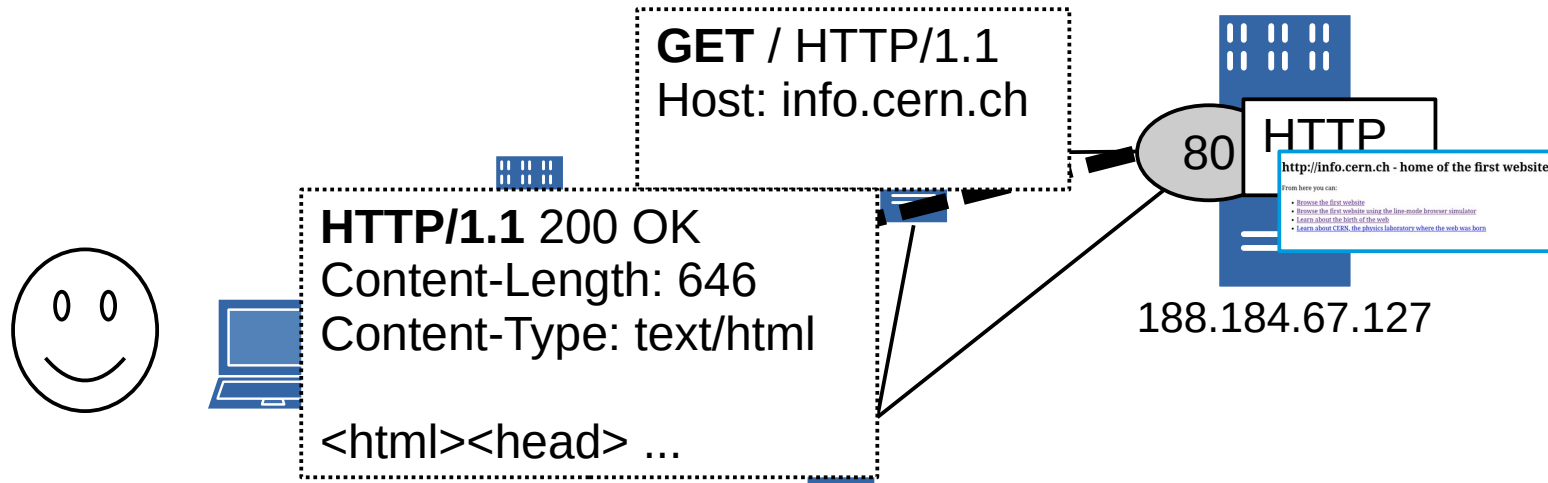
- 1) Risolvere l'indirizzo IP corrispondente all'hostname;
- 2) iniziare una connessione TCP/IP sulla porta associata al protocollo;
- 3) inviare una richiesta HTTP GET;



Riepilogo

Quali operazioni effettua il browser per visualizzare la pagina `http://info.cern.ch`?

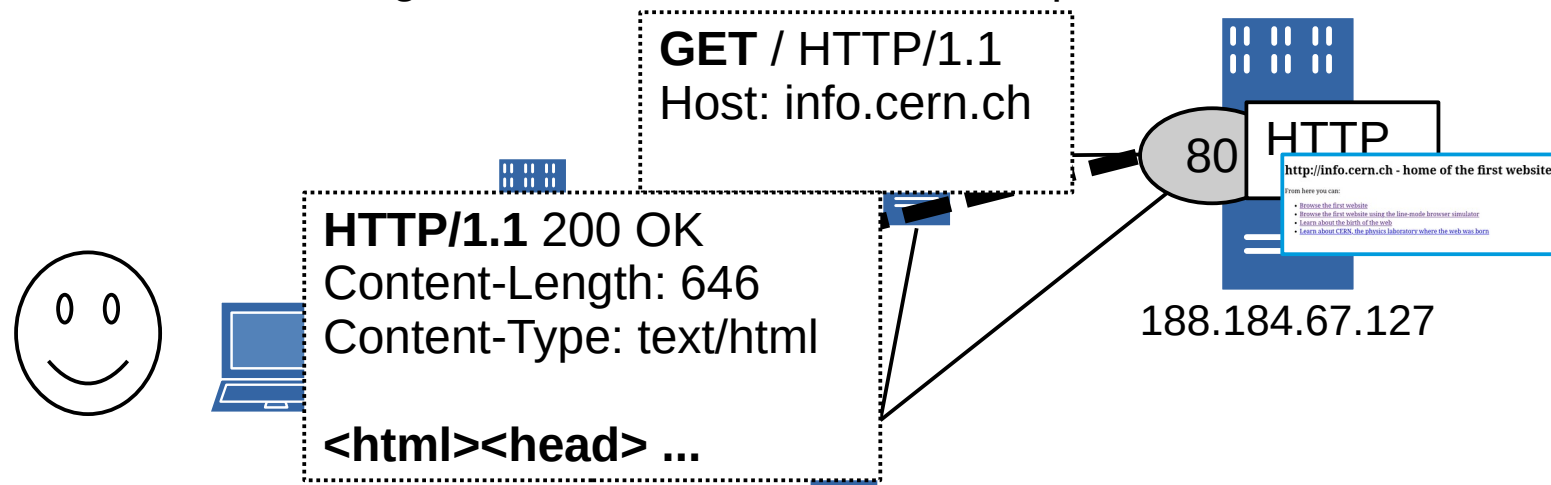
- 1) Risolvere l'indirizzo IP corrispondente all'hostname;
- 2) iniziare una connessione TCP/IP sulla porta associata al protocollo;
- 3) inviare una richiesta HTTP GET;
- 4) analizzare la risposta;



Riepilogo

Quali operazioni effettua il browser per visualizzare la pagina `http://info.cern.ch`?

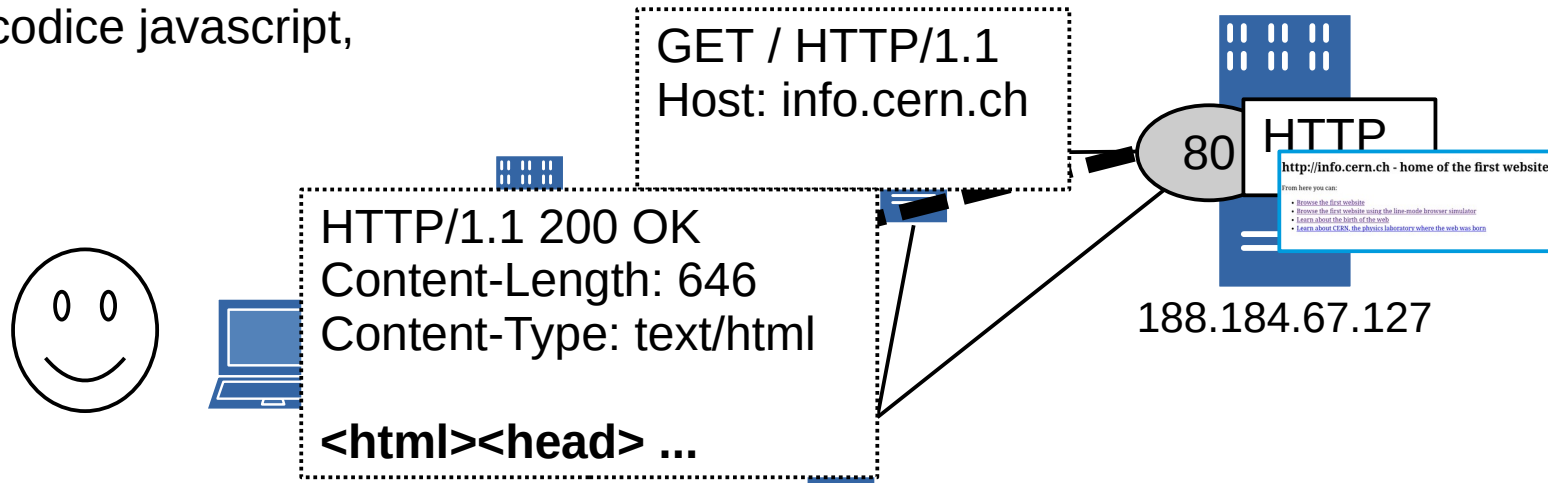
- 1) Risolvere l'indirizzo IP corrispondente all'hostname;
- 2) iniziare una connessione TCP/IP sulla porta associata al protocollo;
- 3) inviare una richiesta HTTP GET;
- 4) analizzare la risposta;
- 5) se lo status code è 2xx, iniziare il *rendering* del documento HTML nel corpo



Riepilogo

Quali operazioni effettua il browser per visualizzare la pagina `http://info.cern.ch`?

- 1) Risolvere l'indirizzo IP corrispondente all'hostname;
- 2) iniziare una connessione TCP/IP sulla porta associata al protocollo;
- 3) inviare una richiesta HTTP GET;
- 4) analizzare la risposta;
- 5) se lo status code è 2xx, iniziare il *rendering* del documento HTML nel corpo;
 - a) eseguire l'eventuale codice javascript,



Riepilogo

Quali operazioni effettua il browser per visualizzare la pagina `http://info.cern.ch`?

- 1) Risolvere l'indirizzo IP corrispondente all'hostname;
- 2) iniziare una connessione TCP/IP sulla porta associata al protocollo;
- 3) inviare una richiesta HTTP GET;
- 4) analizzare la risposta;
- 5) se lo status code è 2xx, iniziare il *rendering* del documento HTML nel corpo;
 - a) eseguire l'eventuale codice javascript,
 - b) scaricare eventuali CSS esterni,

