



Master in Informatica del Testo ed Edizione Elettronica

Master professionale e di ricerca

Università degli Studi di Siena

Filologia e Critica delle Letterature Antiche e Moderne – Ingegneria dell'Informazione - Scienze Matematiche

ELEMENTI DI INFORMATICA DI BASE

Anno 2022

Dario Madeo

dario.madeo@unisi.it

<https://www3.diism.unisi.it/~madeo/home.html>

HTML & JavaScript

28 Gennaio 2022

Rivisitazione delle lezioni del prof. Stefano Melacci (2015)

Testi di riferimento

Tanenbaum Andrew S., **"Reti di calcolatori"**, Pearson Education Italia

Laura Lemay, Rafe Colburn, **"HTML & CSS"**, McGraw Hill

Dave Thau, **"Javascript. La grande guida"**, Mondadori Informatica



HTML



Esempio di pagina HTML

```
<html>
  <head>
    <title>Questo è un esempio</title>
  </head>
  <body>
    Ciao! Questo è
    un esempio di pagina HTML
  </body>
</html>
```

- La pagina è identificata dall'estensione **.html** o **.htm**

Sintassi dei Tag

- Quasi tutti i tag sono composti da un **tag di apertura** e da un **tag di chiusura**

`<nome_tag>testo sul quale il tag agisce</nome_tag>`

Es.: `<title>Questo è un esempio</title>`

- Alcuni tag non hanno la corrispondente chiusura:

`<nome_tag>`

Es.: `
`

- Per molti tag è possibile specificare degli attributi:

`<nome_tag nome_attributo="valore_attributo">`

Es.: `<body bgcolor="red"> ... </body>`

Esempio di pagina HTML

```
<html>
  <head>
    <title>Questo è un esempio</title>
  </head>
  <body>
    <h1>Titolo principale</h1>
    <h2>Titolo secondario</h2>
    <p>Testo all'interno di un
    <strong>paragrafo</strong></p>
    <p>Altro paragrafo</p>
    Testo al di fuori di un paragrafo
  </body>
</html>
```

Stili

`<p>Primo paragrafo</p>`

Colore del testo

`<p style="color:red;">Secondo paragrafo</p>`

`<body style="color:blue;">...</body>`

`<h1 style="color:#0022AA;">Heading 1</h1>`

Stile "locale"

Stile "globale"

Codifica RGB (in base 16)

00: quantità di rosso (Red)

22: quantità di verde (Green)

AA: quantità di blu (Blue)

FF0000: rosso
FFFF00: giallo
00FF00: verde
00FFFF: celeste (ciano)
0000FF: blu
FF00FF: magenta

000000: nero
FFFFFF: bianco
xxxxxx: grigio
(xx è un valore generico)

Minimo = 00 = (0)₁₀ - Massimo = FF = (255)₁₀

Stili

Colore dello sfondo

Stile "locale"

```
<p style="background-color:red;">Paragrafo</p>  
<body style="background-color:#FFF9D8;">Heading 1</body>
```

Stile "globale"

Si possono combinare stili diversi:

```
<p style="background-color:red; color:#00FF00;">Paragrafo</p>
```

Gli stili "locali" sostituiscono quelli "globali" (overriding)

Collegamenti Iperestuali (Link)

- Servono per creare connessioni navigabili
 - ↳ Link interni alla stessa pagina
 - ↳ Link tra pagine

- Link:

```
<a href="indirizzo">qualsiasi cosa</a>
```

Facendo click su qualsiasi cosa, il browser si sposta a
indirizzo

Indirizzo può essere una **URL**

Immagini

- E' possibile includere immagini:

```

```

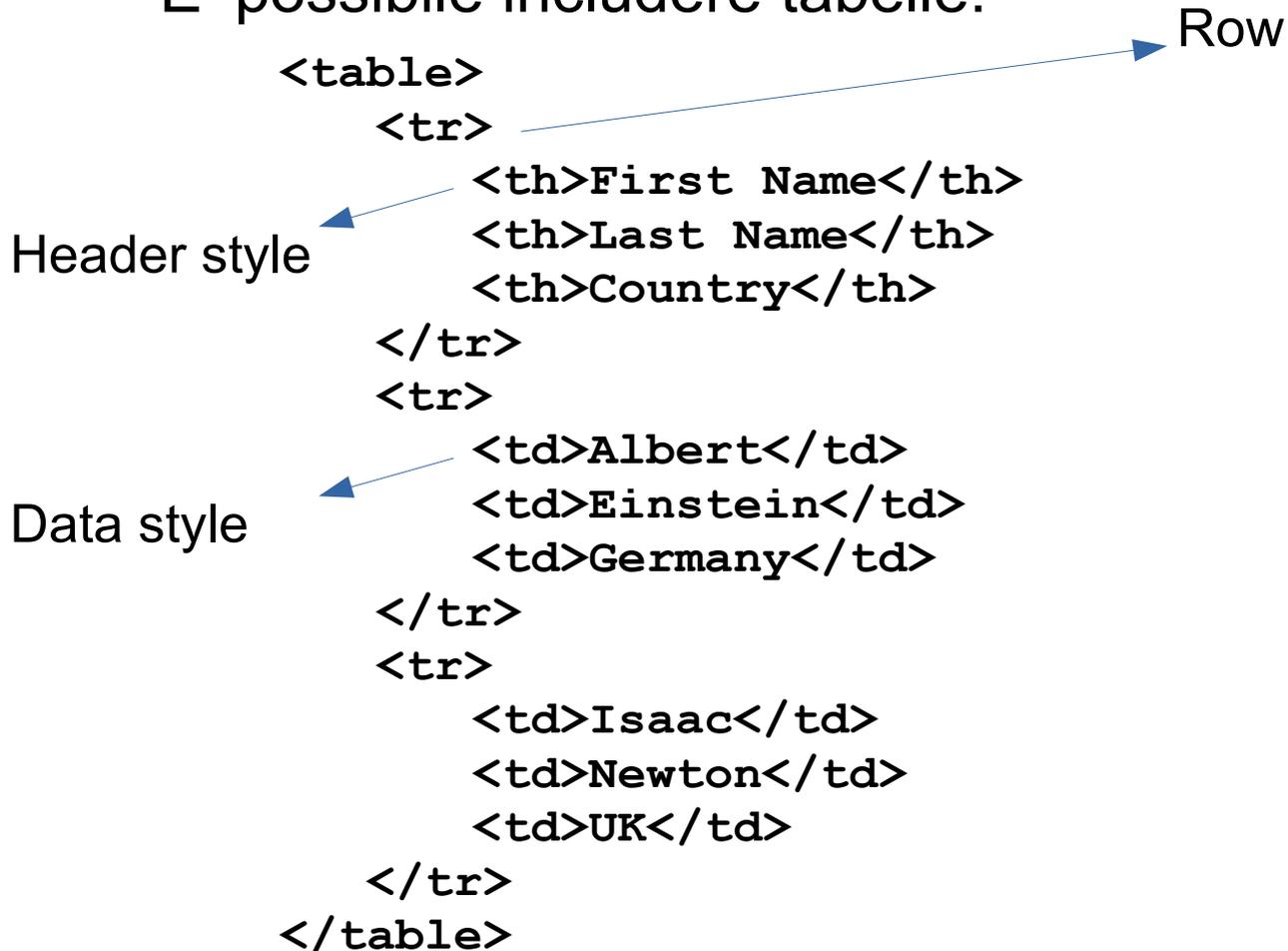
- Alcuni attributi permettono di modificare le dimensioni dell'immagine:

```
Es.: 
```

```
Es.: 
```

Tabelle

- E' possibile includere tabelle:



Tabelle

- E' possibile definire lo stile della tabella:

```
<table style="background-color:red;">
  <tr>
    <th style="border: 1px solid black;">First Name</th>
    <th style="border: 1px solid black;">Last Name</th>
    <th style="border: 1px solid black;">Country</th>
  </tr>
  <tr>
    <td style="border: 1px solid black;">Albert</td>
    <td style="border: 1px solid black;">Einstein</td>
    <td style="border: 1px solid black;">Germany</td>
  </tr>
  <tr>
    <td style="border: 1px solid black;">Isaac</td>
    <td style="border: 1px solid black;">Newton</td>
    <td style="border: 1px solid black;">UK</td>
  </tr>
</table>
```

Stili generici

- E' possibile definire uno stile principale per non ripeterlo...

```
<head>
```

```
<style>  
  table, th, td {  
    border: 4px solid black;  
  }  
  
  p {  
    background-color: #FAE6F6;  
  }  
  
  h1 {  
    color: red;  
  }  
</style>
```

```
</head>
```

CSS

(Cascading Style Sheets)

- Definisce fogli di stile per pagine HTML
- File di tipo `.css` in cui vengono dettate le regole per definire i parametri delle varie sezioni di una pagina HTML
- Creare un file separato per lo stile piuttosto che inserirlo nell'header, aumenta la "pulizia" del lavoro...

Pagine web dinamiche

- HTML non consente modifiche user-dependent dei contenuti definiti nella pagina web
 - 👉 La pagina web è "statica", nel senso che non può essere alterata in base alle operazioni effettuate dall'utente che la visita
- Le pagine web **dinamiche** consentono invece tali alterazioni
 - 👉 La loro realizzazione è possibile tramite linguaggi e standard appositamente concepiti, come:
 - Javascript
 - PHP
- Chi elabora le pagine dinamiche?
 - 👉 Dipende dal metodo di realizzazione selezionato



Java script



Javascript

- JavaScript è un linguaggio di scripting
- È particolarmente utile perché tutti i compiti di elaborazione sono Scritti nello script (incorporato nel documento HTML), e quindi L'intero processo definito dallo script è eseguito sul lato del client, Senza la necessità di fare riferimento a un server
- Invece di fare eseguire al server dei compiti e fornire i risultati al browser, quest'ultimo può gestire localmente alcuni compiti, dando così all'utente tempi di risposta più brevi e riducendo il traffico

Javascript

- Quando un documento HTML con uno script di Javascript è caricato in un browser che supporta questo linguaggio, le funzioni dello script vengono calcolate, memorizzate, ed eseguite quando si verificano determinati eventi
 - ☞ Ad esempio, quando l'utente sposta il mouse sopra un oggetto o immette un testo in una casella

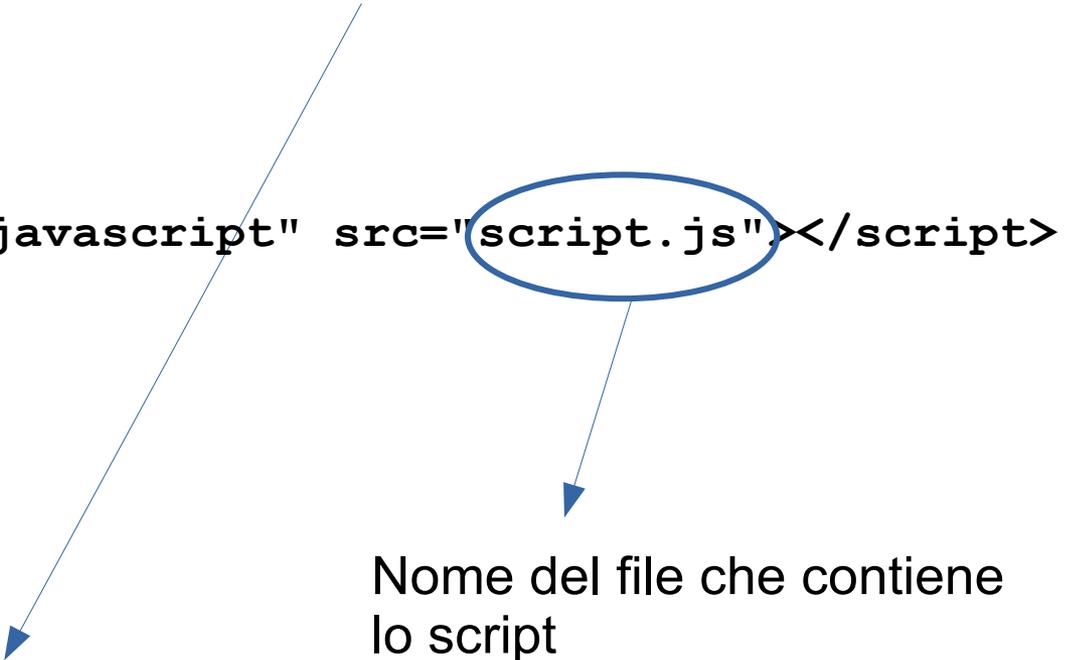
Javascript

- Per aggiungere del codice JavaScript ad un documento HTML, esso va inserito nel head (ma non è l'unico modo!)

```
<head>
```

```
<script type="text/javascript" src="script.js"></script>
```

```
</head>
```



Nome del file che contiene lo script

Potrebbe essere direttamente inserito nel file HTML...

Ma lavorare su file separati aumenta la pulizia ed aiuta l'IDE (diversa sintassi tra HTML e JavaScript)!

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

HTML

```
<a href="http://www.google.com"  
id="mylink"  
onmouseover="mouse_in()"  
onmouseout="mouse_out()">Google</a>
```

Id oggetto

Azione
evento
"mouse
over"

Azione evento "mouse out"

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

JavaScript

```
function mouse_in() {  
    var mylink = document.getElementById("mylink");  
    mylink.style.backgroundColor = "red";  
}
```

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

JavaScript

```
function mouse_in() {  
    var mylink = document.getElementById("mylink");  
    mylink.style.backgroundColor = "red";  
}
```

↓
“Firma” della funzione

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

JavaScript

```
function mouse_in() {  
    var mylink = document.getElementById("mylink");  
    mylink.style.backgroundColor = "red";  
}
```

“Scope” della funzione. Il contenuto della funzione deve stare dentro alle parentesi graffe (marcatori).

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

JavaScript

```
function mouse_in() {  
    var mylink = document.getElementById("mylink");  
    mylink.style.backgroundColor = "red";  
}
```



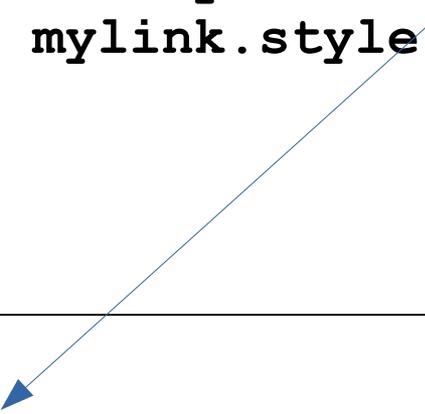
Definizione (ed assegnamento) di una variabile.
Variabile = parte di memoria (RAM) richiamabile con un nome
"human-readable"

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

JavaScript

```
function mouse_in() {  
    var mylink = document.getElementById("mylink");  
    mylink.style.backgroundColor = "red";  
}
```



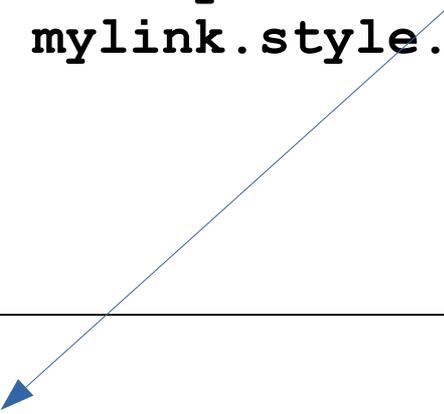
“document” è una variabile (oggetto) “già definita” che contiene la struttura della pagina HTML.

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

JavaScript

```
function mouse_in() {  
    var mylink = document.getElementById("mylink");  
    mylink.style.backgroundColor = "red";  
}
```



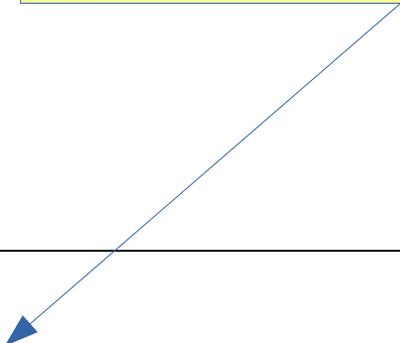
L'oggetto "document" possiede una funzione "già definita", "getElementById".
Per accedere ad una funzione appartenente ad una variabile, è necessario usare il "."
La funzione "getElementById" cerca nell'HTML gli elementi che hanno "id = mylink".

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

JavaScript

```
function mouse_in() {  
    var mylink = document.getElementById("mylink");  
    mylink.style.backgroundColor = "red";  
}
```



L'oggetto "mylink" possiede la variabile "style.backgroundColor". Possiamo settarla a nostro piacimento.

Gestione Eventi

- Javascript consente di gestire eventi "locali" associati al client, quali passaggio con il puntatore del mouse sopra ad un link o immagine o testo

JavaScript

```
function mouse_out() {  
    var mylink = document.getElementById("mylink");  
    mylink.style.backgroundColor = "";  
}
```

Moduli HTML

- Permettono di interagire con la pagina

Campo di testo

```
<input id="birthYear">
```

Pulsante

```
<button onclick="get_age()">Calcola l'età!  
</button>
```

Esistono altri “moduli” (radio button, check button...)

Esempio

HTML

```
<table>
  <tr>
    <th>Anno di nascita</th>
    <th>Età</th>
    <th></th>
  </tr>
  <tr>
    <th><input id="birthYear"></th>
    <th><input id="age" disabled></th>
    <th><button onclick="get_age()">Calcola
l'età!</button></th>
  </tr>
</table>
```

Input

Output
Notare la
keyword
"disabled".
Impedisce
all'utente di
inserire dati
in quel campo.

**Funzione JavaScript
invocata quando il
pulsante viene cliccato**

Esempio

JavaScript

```
function get_age() {  
    var birthYear =  
document.getElementById('birthYear').value;  
  
    var d = new Date();  
    var currentYear = d.getFullYear();  
  
    var ageValue = currentYear - birthYear;  
  
    var age = document.getElementById("age");  
    age.value = ageValue;  
}
```

Legge
l'input

Data
odierna

Anno
corrente

Calcolo
età

Scrive
il risultato
nel campo
output

Funzione di JavaScript per ottenere la data
corrente

Esempio

HTML

```
<table>
  <tr>
    <th>Anno di nascita</th>
    <th>Età</th>
  <tr>
    <th>Età</th>
    <th></th>
  </tr>
  <tr>
    <th><input id="birthYear"></th>
    <th><input id="age" disabled></th>
    <th><button onclick="get_age()">Calcola
l'età!</button></th>
  </tr>
</table>
```

Input

Output
Notare la
keyword
"disabled".
Impedisce
all'utente di
inserire dati
in quel campo.

**Funzione JavaScript
invocata quando il
pulsante viene cliccato**

Controllo di flusso

HTML

```
<table>
  <tr>
    <th>Anno di nascita</th>
    <th>Età</th>
    <th>Fase della vita</th>
    <th></th>
  </tr>
  <tr>
    <th><input id="birthYear"></th>
    <th><input id="age" disabled></th>
    <th><input id="phase" disabled></th>
    <th><button onclick="get_age()">Calcola
l'età!</button></th>
  </tr>
</table>
```

Controllo di flusso

JavaScript

```
function get_age() {  
    ...  
  
    age.value = ageValue;  
  
    var phaseValue = "";  
  
    if (ageValue >= 0 && ageValue <= 2) {  
        phaseValue = "Prima infanzia";  
    } else if (ageValue >= 3 && ageValue <= 6) {  
        phaseValue = "Seconda infanzia";  
    } else if (ageValue >= 7 && ageValue <= 12) {  
        phaseValue = "Terza infanzia";  
    } else if (ageValue >= 13 && ageValue <= 20) {  
        phaseValue = "Adolescenza";  
    }  
}
```

Keyword "if"
(decide rispetto a
dati non noti a priori)

Condizione

Effetto (azione)

Controllo di flusso

JavaScript

```
function get_age() {  
    ...  
  
    age.value = ageValue;  
  
    var phaseValue = "";  
  
    if (ageValue <= 2) {  
        phaseValue = "Prima infanzia";  
    } else if (ageValue >= 3 && ageValue <=6) {  
        phaseValue = "Seconda infanzia";  
    } else if (ageValue >= 7 && ageValue <= 12) {  
        phaseValue = "Terza infanzia";  
    } else if (ageValue >= 13 && ageValue <= 20) {  
        phaseValue = "Adolescenza";  
    }  
}
```

Keyword "else if"
(decide rispetto a
dati non noti a priori)

Condizione

|| = OR logico (algebra di Boole)
E' sufficiente che una solo condizione
sia soddisfatta

&& = AND logico (algebra di Boole)
Le due condizioni deve essere soddisfatte
contemporaneamente.

Controllo di flusso

JavaScript

```
function get_age() {  
    ...  
    phaseValue = "Adolescenza";  
} else if (ageValue >= 21 && ageValue <= 40) {  
    phaseValue = "Giovinezza";  
} else if (ageValue >= 41 && ageValue <= 65) {  
    phaseValue = "Età adulta";  
} else if (ageValue >= 66) {  
    phaseValue = "Età senile";  
} else {  
    phaseValue = "non determinata";  
}  
  
var phase = document.getElementById("phase");  
  
phase.value = phaseValue;  
}
```

Cosa faccio nei casi non previsti?

Scrive il risultato nel campo output

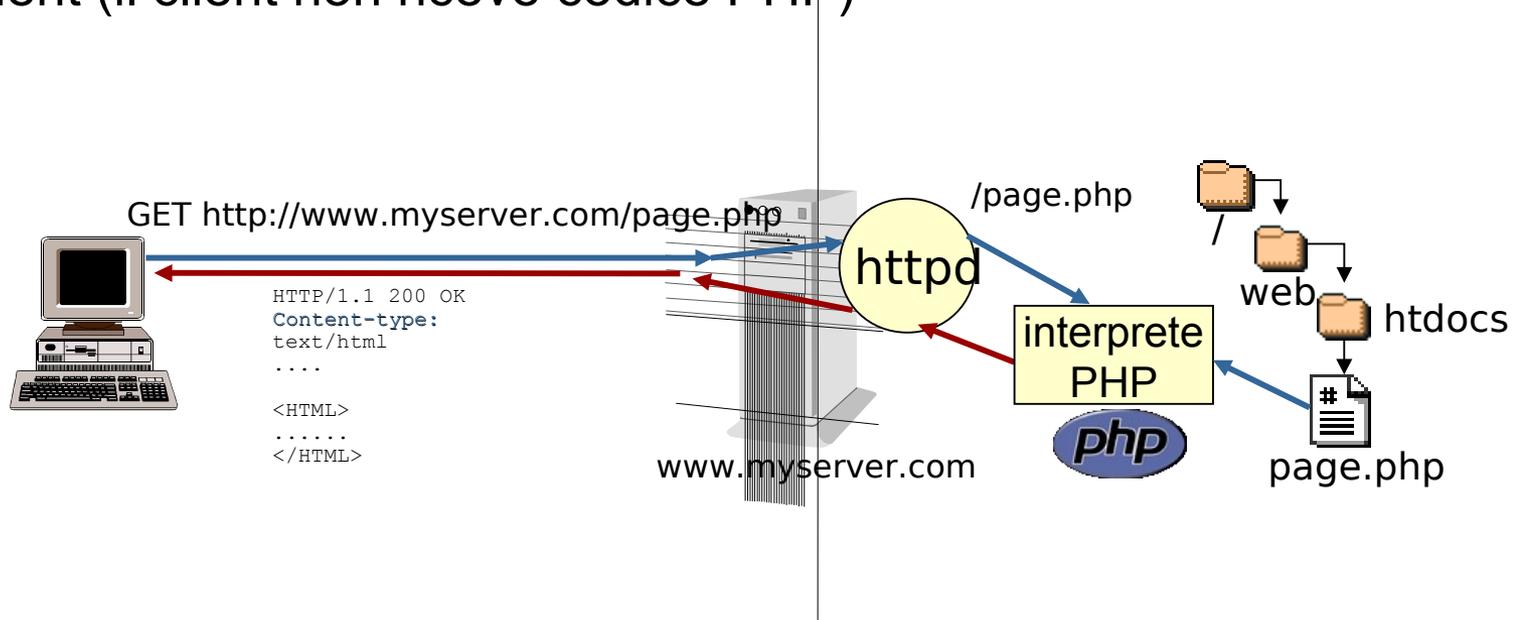


PHP

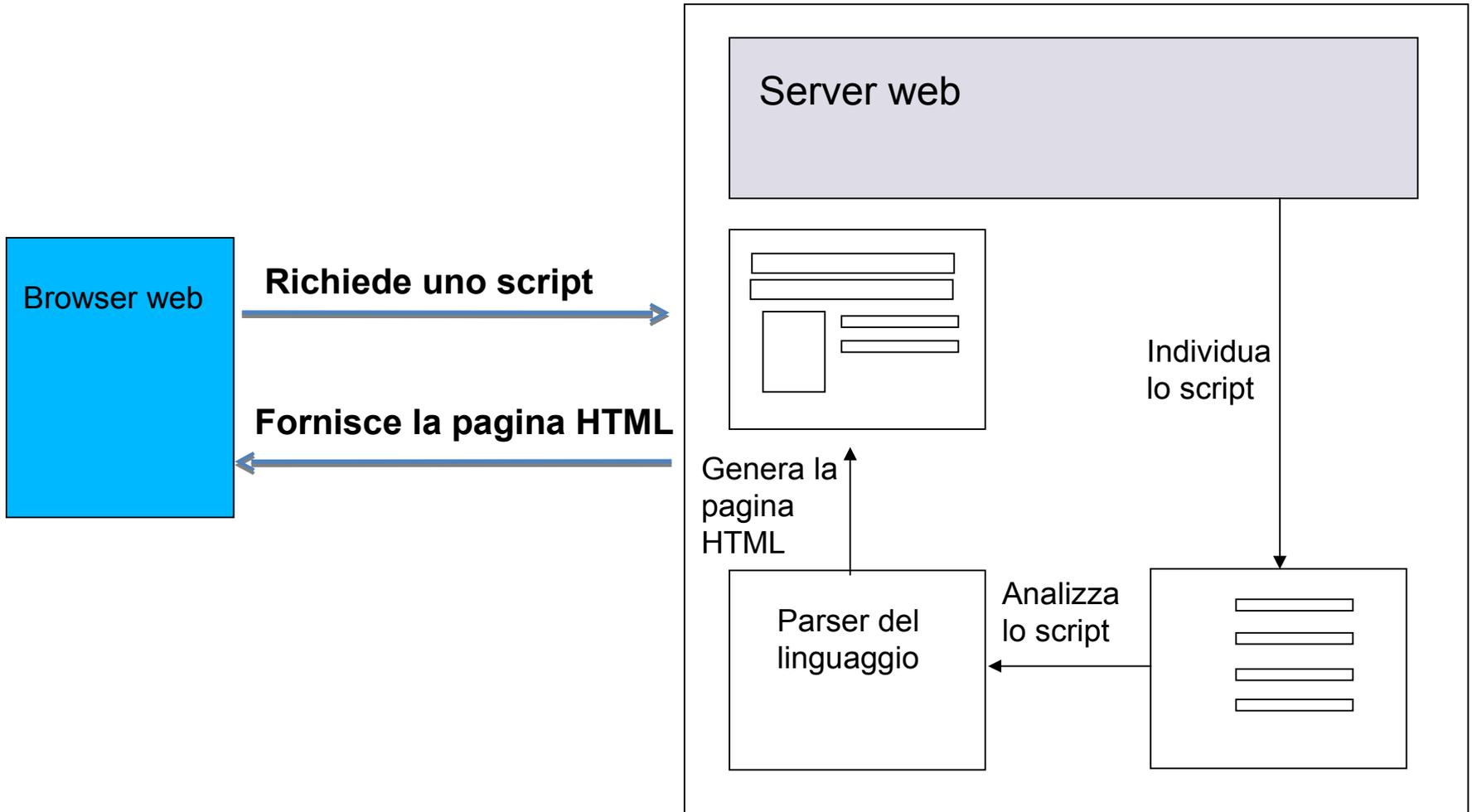


PHP

- ▲ Il **PHP (PHP: Hypertext Preprocessor)** è un linguaggio di programmazione utilizzato per lo sviluppo di **pagine Web dinamiche**
- ▲ Il linguaggio è di tipo **HTML-embedded** ovvero le istruzioni sono inserite in un file HTML e vengono interpretate prima di essere trasmesse al client
- ▲ Gli script vengono eseguiti sul server
- ▲ Il server genera dinamicamente codice html che viene inviato al client (il client non riceve codice PHP)



Client e Server



Caratteristiche

- Sintassi simile ai linguaggio C e Java
- Fornisce connettività a **database**
 - ↳ MySQL, PostgreSQL ...
- Pagine facili da creare
- Non necessitano di compilazione (Linguaggio Interpretato)
- Tecnologie simili sono **ASP** (Active Server Pages) e **JSP** (Java Server Pages)

Pagina PHP

- Una pagina PHP è un file con estensione .php e contiene:

- ✎ Testo

- ✎ Marcatori HTML

- ✎ Comandi script

- In PHP la parte di script può apparire in qualsiasi punto della pagina html e viene delimitato dai delimitatori di script:

- `<? e ?>`

- oppure

- `<?php e ?>`

- Ad esempio il comando: `<? $x=10 ?>` assegna alla variabile x il valore 10

Esecuzione Script PHP

- Quando il web server riceve la richiesta di una pagina con estensione **.php** sa che la pagina deve essere processata dal motore PHP
 - 👉 Il motore PHP è un software installato sulla macchina che ospita il server web
- Il risultato dell'esecuzione dello script viene restituito al web server che lo invia al client
- Gli elementi HTML sono passati direttamente al server web...
-mentre le parti di script vengono interpretate dal motore di scripting PHP

Dove si inserisce PHP?

```
<html>
  <head>
    <title>Greetings!</title>
  </head>

  <body>
    <H1>Pagina di prova</H1>
    <?
      // Questo è uno script PHP
      echo "TESTO PRODOTTO DA PHP";
    ?>
  </body>
</html>
```