

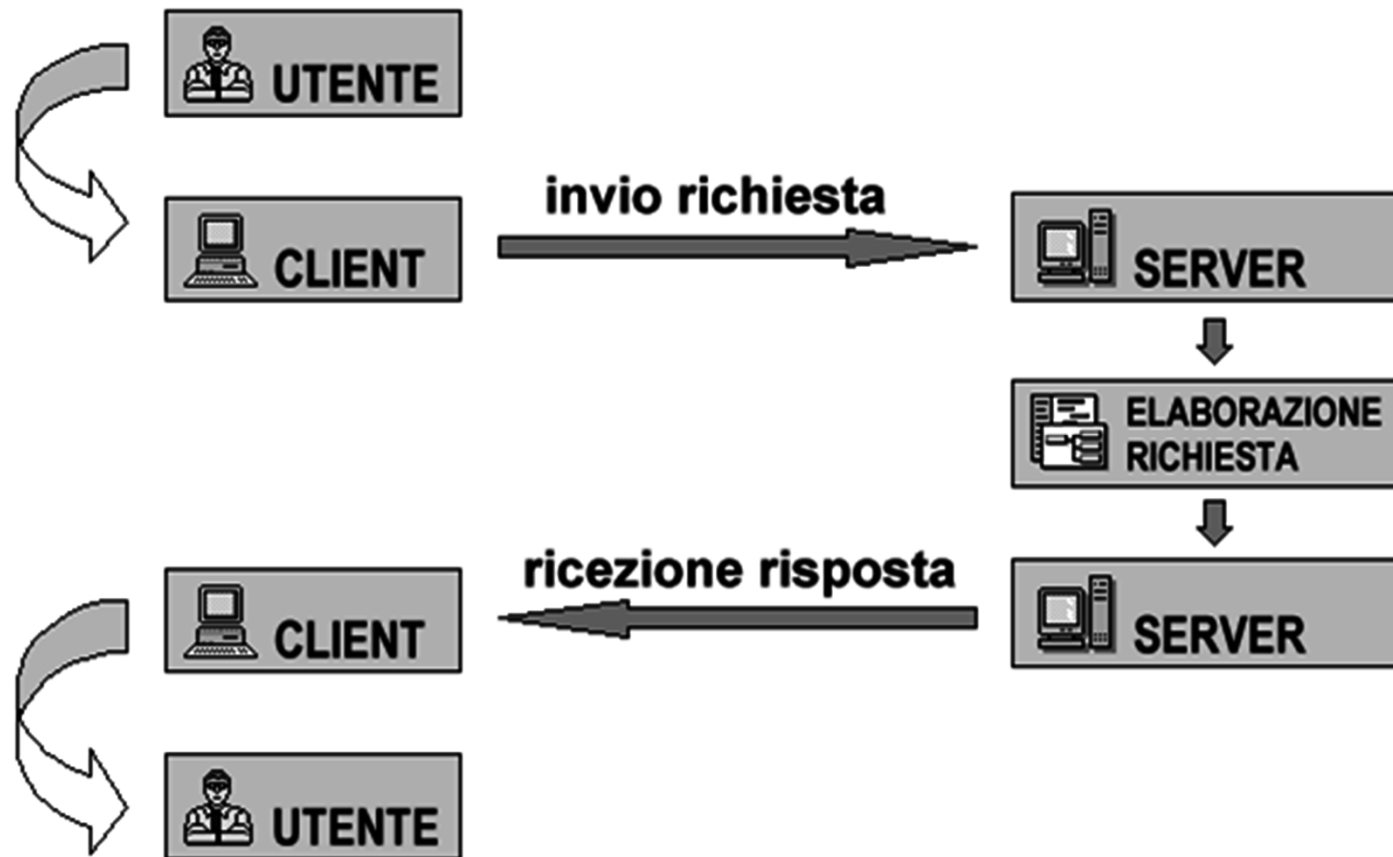
# Web Programming

Esempio di modulo per le scuole superiori

# Applicazioni Web

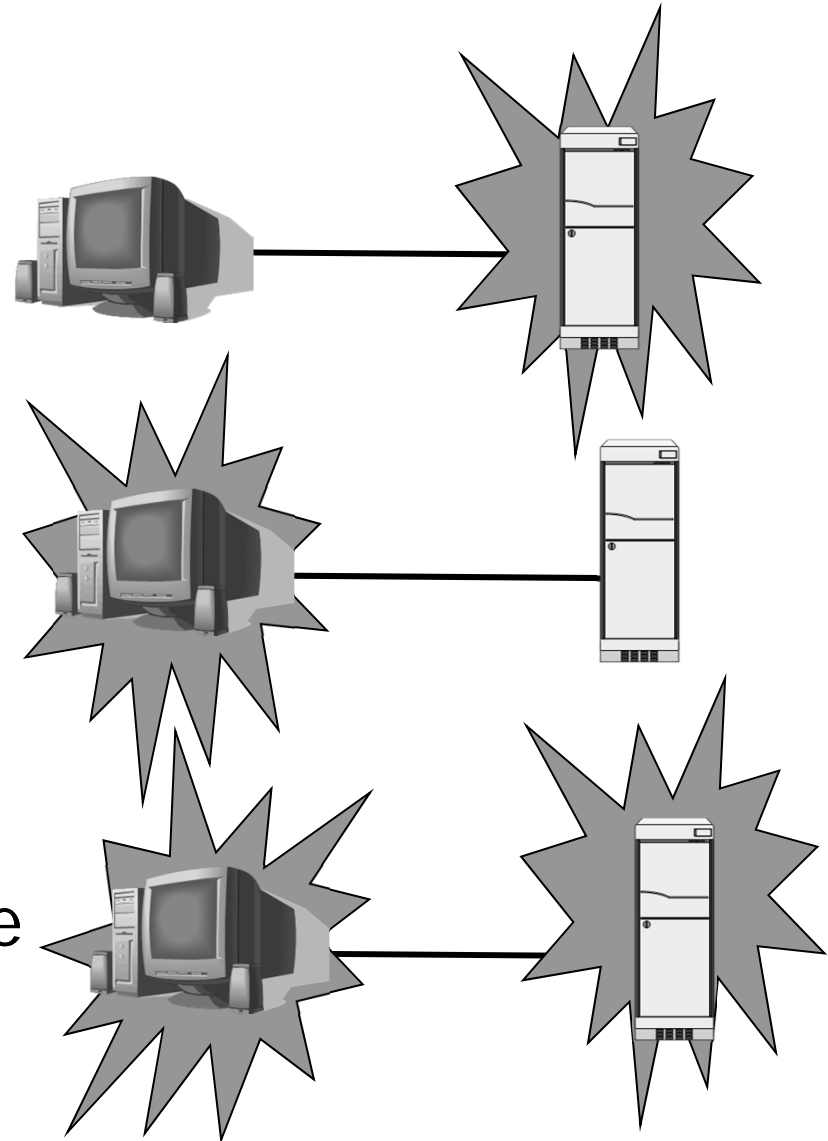
- Una applicazione web fornisce un servizio all'utente finale per mezzo dell'infrastruttura web
- In una applicazione web l'utente interagisce con il sito
- Si parla anche di siti dinamici, dove le pagine non sono statiche ma costruite dinamicamente in base all'interazione con l'utente

# Architettura Client-Server



# Tecnologie client-side e server-side

- Tecnologia server side: il peso della computazione risiede tutta sul server
- Tecnologia client-side: quando la computazione avviene principalmente sul browser
- Generalmente le applicazioni web complesse usano strategia ibrida



# Tecnologia Client Side

- E' la parte di programmazione che coinvolge il browser web ed è quindi dipendente dal tipo (e versione) del browser usato. Alcuni linguaggi client-side non sono supportati da alcuni browser. Uno standard è Javascript
- Uno script può comportarsi in modo diverso nei vari browser web
- Tipicamente le operazioni effettuate client side sono di interazione stretta con l'utente (es. rollover sui tasti o il controllo di dati inseriti tramite FORM)
- Generalmente abbiniamo alla programmazione client side compiti semplici e di interazione immediata con l'utente

# Tecnologia Server Side pt.1

- La programmazione server side si presta a maggiori possibilità e a complessità di computazione maggiore, nonché all'immagazzinamento dei dati
- L'interattività non è immediata in quanto è il server che esegue la parte di programma
- Avere la possibilità di programmare la parte server ci apre la possibilità per far fare al server qualcosa in più del semplice restituire pagine HTML che già risiedono sul disco
- La maggior parte dei siti web che consultiamo ogni giorno sono di fatto applicazioni server-side con parte meno rilevante client-side: portali, commercio elettronico, blog, ecc...

# Tecnologia Server Side pt.2

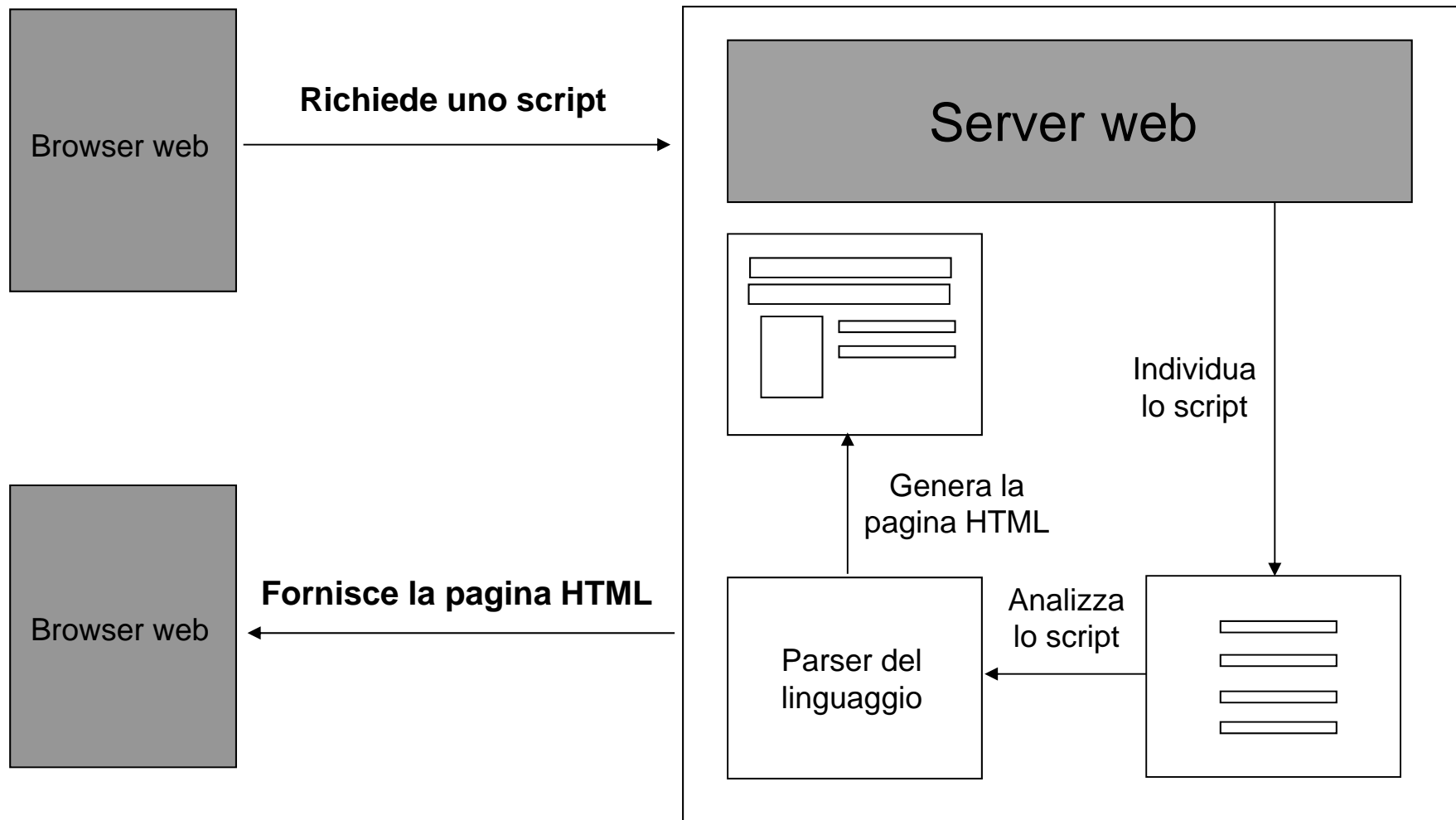
- L'utente può interagire con il server ad esempio sottomettendo dati che il server elabora per restituire poi una risposta sotto forma di pagina HTML
- Un tipico caso è l'accesso da parte dell'utente ad un database che risiede sul server. In questo caso la pagina web funziona come una interfaccia per accedere ai dati che risiedono sul server
- Ci sono varie tecnologie server side utilizzate tra cui Common Gateway Interface (CGI), PHP, ASP.NET, Servlet, JSP

# Tecnologia Server Side pt.3

- Chiaramente il server web deve essere dotato di tecnologia in grado di “costruire” su richiesta le pagine, assemblando insieme i dati e spedirle al browser
- La computazione avviene lato server e può avvenire tramite programmi compilati o script interpretati. Noi ci concentreremo su questa seconda tecnica
- Nel caso di esecuzione di script, il web server ha al suo interno un motore in grado di interpretare il linguaggio di script usato
- Si perde in velocità di esecuzione, ma si guadagna in facilità d'uso

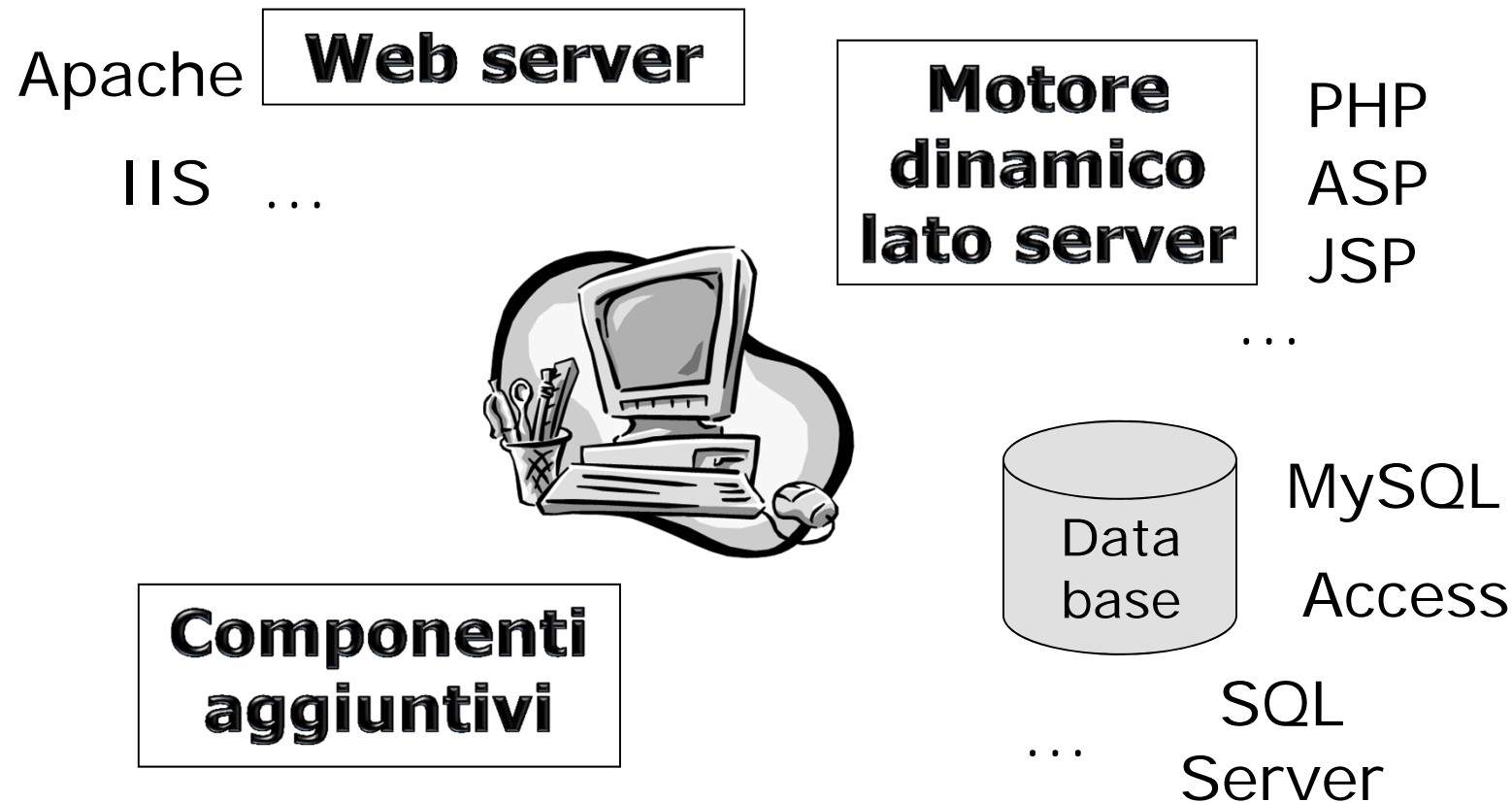


# Linguaggi di scripting



# Application server

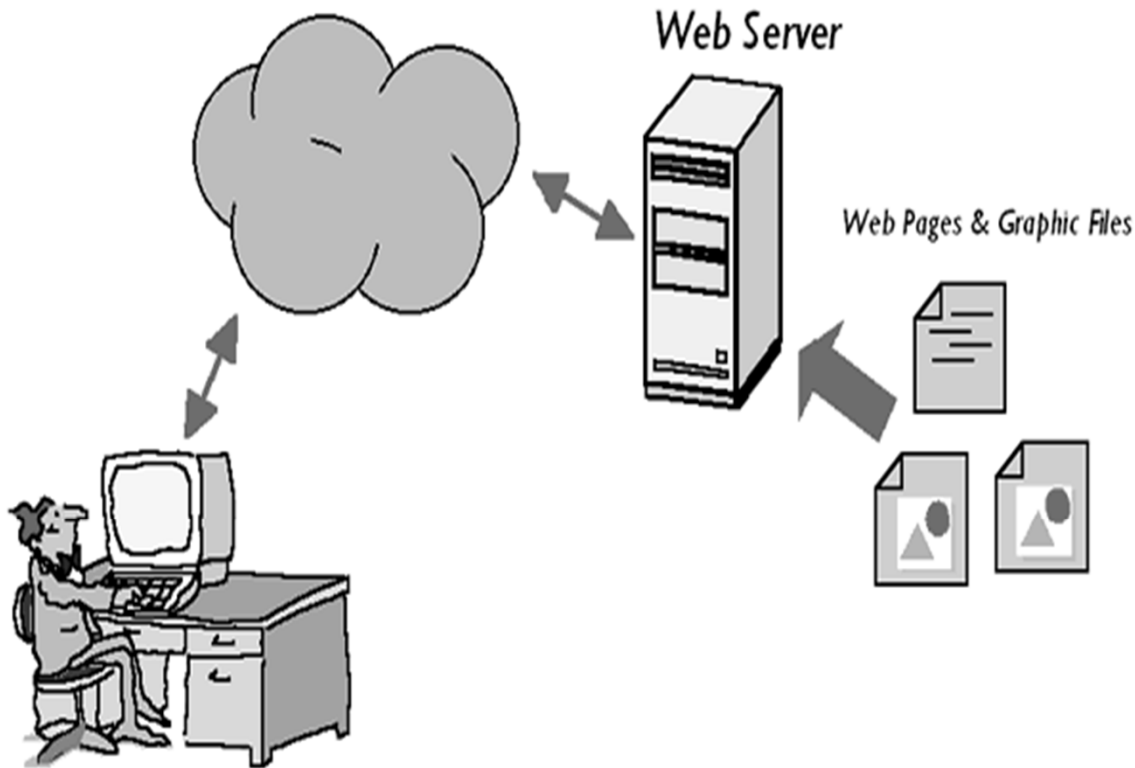
Ambiente per costruire applicazioni web



# I protagonisti del nostro progetto

- Il motore di lavoro:
  - APACHE Web Server
- La benzina:
  - PHP
- La memorizzazione dei dati:
  - MySQL database
- L'interfaccia... con stile:
  - HTML + CSS

# Il Web Server Apache



E' un programma sempre attivo che ascolta su una porta le richieste HTTP. All'arrivo di una richiesta la esegue e restituisce il risultato al browser, rimettendosi in ascolto

# Gestire un web server

## Principali parametri di configurazione:

- Avvio, spegnimento
- Porta di funzionamento
- Directory di default
- Documento di default
- Log degli accessi/errori
- Alias o cartelle virtuali

# Avvio e spegnimento del server

- Tipicamente uno server web per sua stessa definizione è un programma sempre attivo.
- Ci sono però dei casi in cui vogliamo disattivare il server (mettere fuori linea il sito per manutenzione, ad esempio). Per questa ragione ogni server web ha comandi per fermare il server e riavviarlo

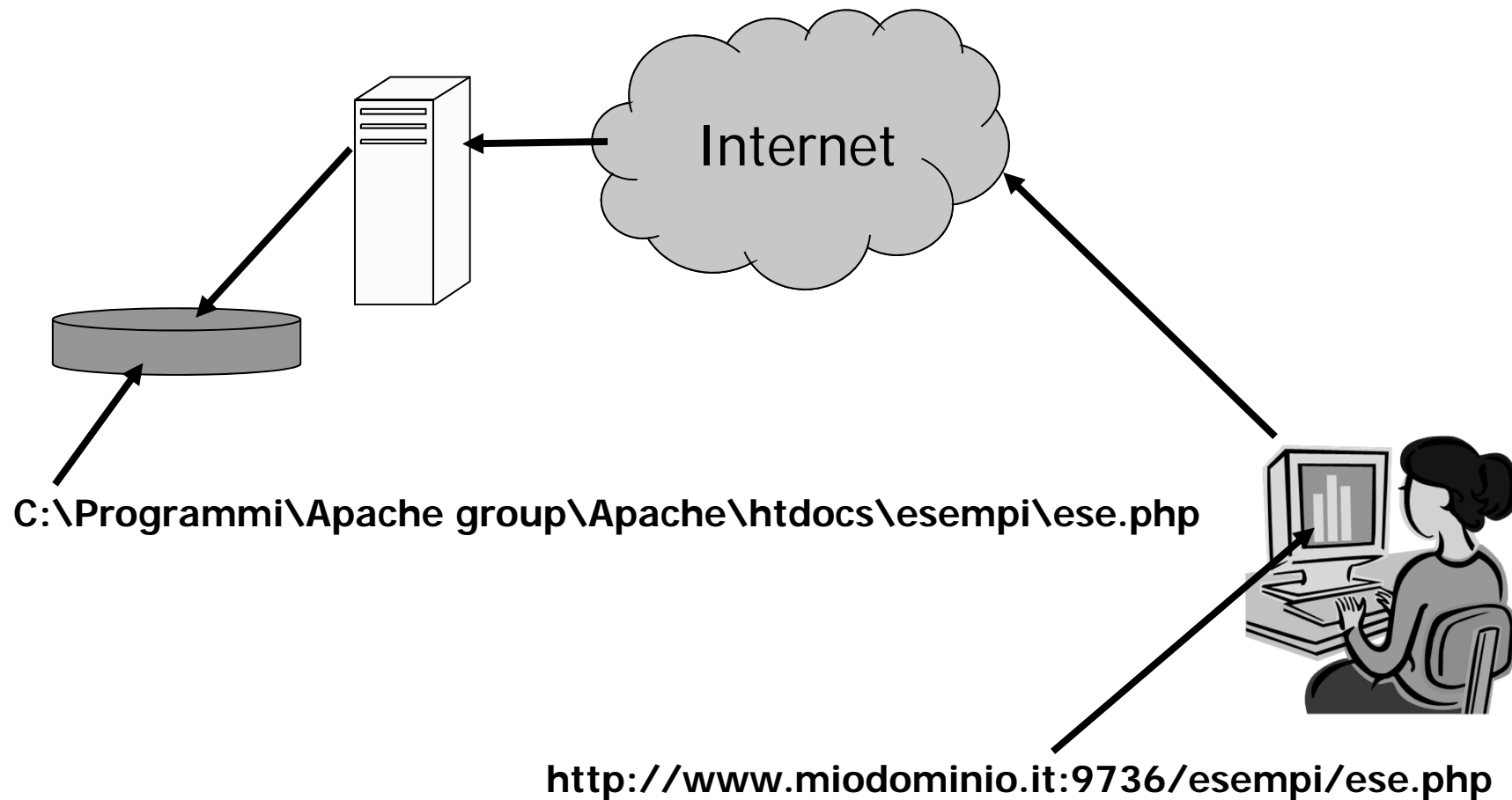
# Web Server e porte

- Il web server (o server HTTP) come tutti i servizi basati su TCP/IP, si attiva su una porta, che è il numero a cui si fa corrispondere il servizio.
- La porta di default del web server è la 80, ma può essere configurato per funzionare su un'altra porta libera. In questo caso la porta va specificata sia nella configurazione del web server che nella URL

`http://localhost:8080/`

`http://www.miodominio.it:9736/esempi/esercizio.php`

# URL e Directory





# Documenti di default

- E' uso comune configurare il server web affinché restituisca un documento predefinito qualora non ne venga specificato uno nella URL
- Questo permette ai siti web di rendere visibile una "home page" digitando il solo indirizzo del web server  
Ad es: `www.facebook.com` o `www.google.com`
- La pagina visualizzata prende il nome di homepage e tipicamente ha uno dei nomi: `index.html`, `index.htm`, `index.php`, `default.html`, `default.htm`, `default.asp` etc...
- Queste pagine di default e l'ordine in cui vengono cercate possono essere configurate sul server web

# Logs di accesso ed errori

Ogni richiesta arrivata al server web viene registrata in appositi file chiamati Logs di accesso. Possono avere formati diversi, ma essenzialmente contengono informazioni su:

- Indirizzo IP del client
- Data e ora
- Tipo di richiesta (GET o POST)
- Documento richiesto
- Risposta del server (tutto bene o codice di errore)

Analogamente vengono registrati tutti gli errori che genera il server web ad una data richiesta

Ad es. Pagina non trovata (codice 404) o errori specifici dello script se la pagina è dinamica (ad esempio PHP)

# Alias e cartelle virtuali

- Quando un server web è attivo su una macchina la sua cartella di default è automaticamente resa pubblica.
- Se vogliamo pubblicare un documento dobbiamo quindi copiarlo in tale cartella
- In alcuni casi può essere utile rendere pubblica anche un'altra porzione del file system
- Si parla in questo caso di alias o cartelle virtuali
- Il meccanismo sta nel definire una nuova URL che punta ad una specifica cartella del disco, non necessariamente dentro la cartella di default

Ad es:

`http://localhost/esalias/`

corrisponde alla cartella

`C:\esempioalias`

# II PHP: scripting embedded

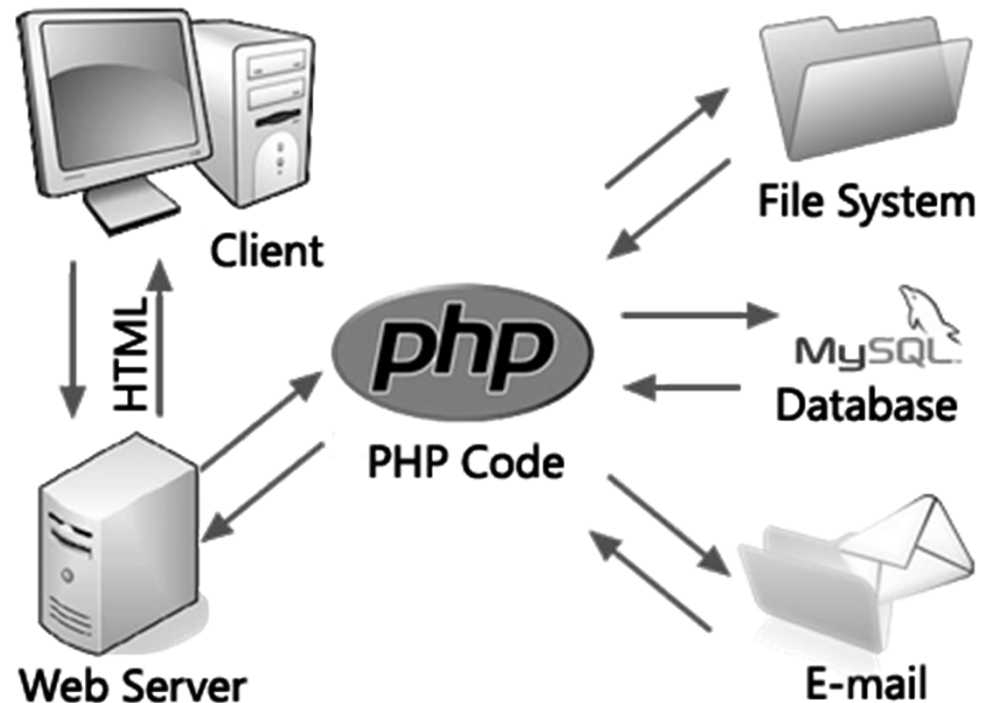
È un linguaggio di programmazione interpretato, indipendente dalla piattaforma, concepito per il Web ovvero per la realizzazione di pagine web dinamiche

L'interprete ha una licenza open source e libera

Permette di interagire con vari database

E' integrabile con numerose librerie esterne (grafica, mail, pdf, ecc...)

Le pagine PHP sono completamente integrate con l'HTML



# Installare PHP-MYSQL

- Per poter utilizzare PHP con il database MySQL (configurazione tipica per le applicazioni web) occorre installare un server web, PHP e MYSQL
- L'installazione integrata di queste tre componenti non è sempre agevole, per questo motivo sono stati resi disponibili in rete dei kit di installazione integrati (ad es. EasyPHP o XAMPP per Windows, LAMP per Linux, ecc...)



# Una pagina PHP pt.1

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

- Testo
- Marcatori HTML
- Comandi script



# Una pagina PHP<sub>pt.2</sub>

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

`<?php e ?>`

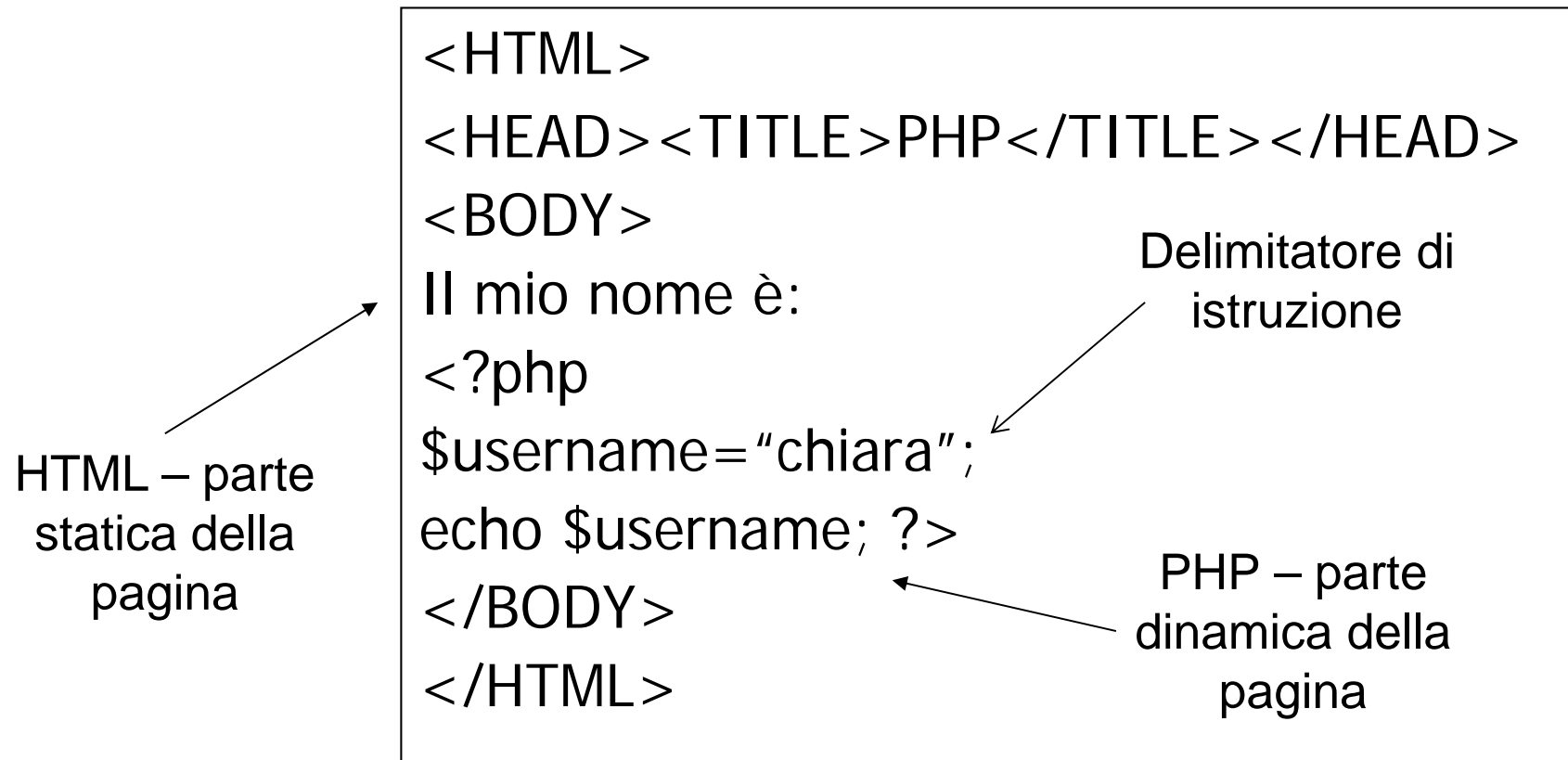
Ad esempio il comando:

`<?php $x="ciao"; ?>`

assegna alla variabile x il valore "ciao"

# Anatomia di uno script PHP

La pagina deve essere salvata con estensione .php





# MySQL database

- È un database free open source (community edition), liberamente scaricabile da [www.mysql.org](http://www.mysql.org)
- MySQL è disponibile su tutte le piattaforme ed è realizzato come un server (quindi un servizio attivo che risponde su una porta)
- Ogni server MySQL può essere configurato per gestire un certo numero di database, ognuno dei quali potrà gestire un certo numero di tabelle le quali potranno essere popolate con i dati
- L'interazione con il server MySQL può avvenire da riga di comando (shell) digitando i comandi, o tramite query SQL, per creare database, tabelle, inserire dati, fare ricerche, ecc...
- L'interazione può avvenire anche tramite interfaccia grafica, ad esempio con PhpMyAdmin (interfaccia web usata in questo progetto)

# PHP e MySQL pt.1

- PHPMyAdmin è uno strumento facile e intuitivo per effettuare alcune operazioni di gestione del server MySQL.
- Tutte le operazioni possibili da PHPMyAdmin si possono comunque fare anche da script PHP tramite opportuni comandi.
- In PHPMyAdmin è possibile visualizzare il codice PHP di ogni operazione

# PHP e MySQL pt.2

- L'accesso a MySQL server da PHP si può effettuare con semplici funzioni.
- La sequenza dei passi da effettuare è:
  - Effettuare una connessione al server
  - Selezionare il DB o crearlo se non esiste
  - Eseguire la/le query (creare tabelle, inserire o selezionare dati...)
  - Assegnare le eventuali righe risultanti ad un opportuno array e visualizzare il risultato

*Vedremo degli esempi di questi passi direttamente  
sul codice del nostro progetto*

# CSS: Cascading Style Sheets

- È un linguaggio informatico usato per definire la formattazione di documenti HTML, XHTML e XML ad esempio in siti web e relative pagine web
- Le regole per comporre il CSS sono contenute in un insieme di direttive (Recommendations) emanate a partire dal 1996 dal W3C
- L'introduzione del CSS si è resa necessaria per separare i contenuti dalla formattazione e permettere una programmazione più chiara e facile da utilizzare
- Garantisce contemporaneamente anche il riuso di codice ed una sua più facile manutenibilità

# Le regole dei CSS

Il codice CSS da inserire o referenziare è strutturato sotto forma di una o più *regole* ovvero istruzioni di tipo

*proprietà : valore*

che vengono applicate dal browser in fase di rendering, agli elementi HTML interessati, opportunamente specificati tramite un *selettore*, il tutto strutturato secondo il seguente schema sintattico:

```
selettore {  
  proprietà1 : valore1;  
  proprietà2 : valore2, valore3;  
}
```

# Le tipologie di Selettori principali

## Selettori di tipo

I selettori di tipo applicano la regola a tutti gli elementi della pagina del tipo determinato.

*Esempi:*

```
body {  
  [...]  
}
```

```
p {  
  [...]  
}
```

# Le tipologie di Selettori principali

## Classi

Le classi applicano la regola a tutti gli elementi della pagina che presentano la proprietà `class="nome_classe"`

*La sintassi CSS è la seguente:*

```
.nome_classe {  
  [...]  
}
```

# Le tipologie di Selettori principali

## Identificatori

Gli identificatori (comunemente ID) applicano la regola a quell'elemento della pagina che presenta la proprietà `id="nome_identificatore"`. Gli ID contraddistinguono elementi unici.

*La sintassi CSS è la seguente:*

```
#nome_identificatore {  
[...]  
}
```



# Le tipologie di Selettori principali

## Pseudoclassi

Le pseudoclassi identificano elementi in base alle loro proprietà.

*Ad es:*

first-child individua un elemento solo se è il primo figlio dell'elemento padre

```
p:first-child {  
  [...]  
}
```

*individuera nel codice (X)HTML  
seguente solo il primo paragrafo*

```
<body>  
  <p>Lorem ipsum</p>  
  <p>Lorem ipsum</p>  
  <p>Lorem ipsum</p>  
</body>
```

# Le tipologie di Selettori principali

## Pseudoclassi (2)

*Ad es:*

link e visited si applicano ai collegamenti. La prima identifica i collegamenti non visitati, la seconda quelli visitati

```
a:link {  
  [...]  
}
```

active, focus e hover identificano gli elementi solo in particolari condizioni, la prima se l'elemento è attivo, la seconda se è selezionato, la terza se il puntatore è sopra di lui.

```
p:hover {  
  [...]  
}
```

# Codice HTML+PHP significativo nel nostro progetto

```
35 <form name="feedback" method="post" action="PHPscript/scelta.php">
36   <fieldset>
37     <legend>Scegliere un'operazione:</legend>
38     <input type="radio" name="scelta" value="iscritti"/>Elenco
      degli iscritti a ogni corso<br/>
39     <input type="radio" name="scelta" value="liberi"/>Numero di
      posti liberi in ciascun corso<br/>
```

Passaggio di parametri al file scelta.php  
mediante l'uso del metodo post in un  
form html

# Codice HTML+PHP significativo nel nostro progetto

```
1 <?php
2     require ("funzioni.php");
3     $scelta=$_POST['scelta'];
4     //creo la connessione al server MySQL
5     $connessione=mysql_connect('localhost', 'root', 'apriti') or die
6     ('Errore di connessione al server MySQL');
7     //connessione al database
8     mysql_select_db('db_palestra',$connessione) or die('Errore di
9     connessione al database');
10    //fisso il set di caratteri da utilizzare
11    mysql_query("SET NAMES 'utf8'", $connessione);
```

- Inclusione di codice PHP da file esterno
- Lettura variabile passata mediante POST
- Creazione di una connessione ad un DB MySQL

## Codice HTML+PHP significativo nel nostro progetto

```
29      $temp_str="select clienti.cognome, clienti.nome,  
      certificati.data_certificato from (clienti inner join  
      certificati on  
      clienti.pk_cliente=certificati.fk_cliente) where  
      datediff(curdate(), certificati.data_certificato)>365  
      order by certificati.data_certificato asc,  
      clienti.cognome asc";  
30      $errore='Errore nella query di restituzione  
      dell\'elenco dei clienti con certificato medico scaduto';  
31      break;  
32  }  
33  $query=mysql_query($temp_str, $connessione) or die($errore);  
34  operazione($scelta,$query);  
35  mysql_close($connessione);
```

- Creazione ed esecuzione di una query SQL
- Richiamo di una funzione PHP
- Chiusura della connessione al DB

# CSS significativo del nostro progetto

- Richiamo di un foglio di stile da file esterno
- Dichiarazione di un Identificatore (Id) per la tabella
- Tag HTML per la definizione del piede della tabella

```
<html><head><title>Elenco dei posti liberi
</title><link rel="stylesheet"
type="text/css" href="../style.css"></head>
<body id="tabelle">
<h3>Elenco dei posti liberi ad ogni corso</h3>
<table id="background-image">
  <thead>
    <tr>
      <td>Codice corso</td>
      <td>Corso</td>
      <td>Posti liberi</td>
    </tr>
  </thead>
  <tfoot>
    <tr>
      <td><em>Grafica "backimage" dal
      foglio: style.css</em></td>
      <td>&nbsp;</td>
    </tr>
  </tfoot>
</table>
```

# CSS significativo del nostro progetto

- Richiamo di un foglio di stile da file esterno
- Dichiarazione di un Identificatore (Id) per la tabella
- Tag HTML per la definizione del piede della tabella

```
<html><head><title>Elenco dei posti liberi
</title><link rel="stylesheet"
type="text/css" href="../style.css"></head>
<body id="tabelle">
<h3>Elenco dei posti liberi ad ogni corso</h3>
<table id="background-image">
  <thead>
    <tr>
      <td>Codice corso</td>
      <td>Corso</td>
      <td>Posti liberi</td>
    </tr>
  </thead>
  <tfoot>
    <tr>
      <td><em>Grafica "backimage" dal
      foglio: style.css</em></td>
      <td>&nbsp;</td>
    </tr>
  </tfoot>
</table>
```