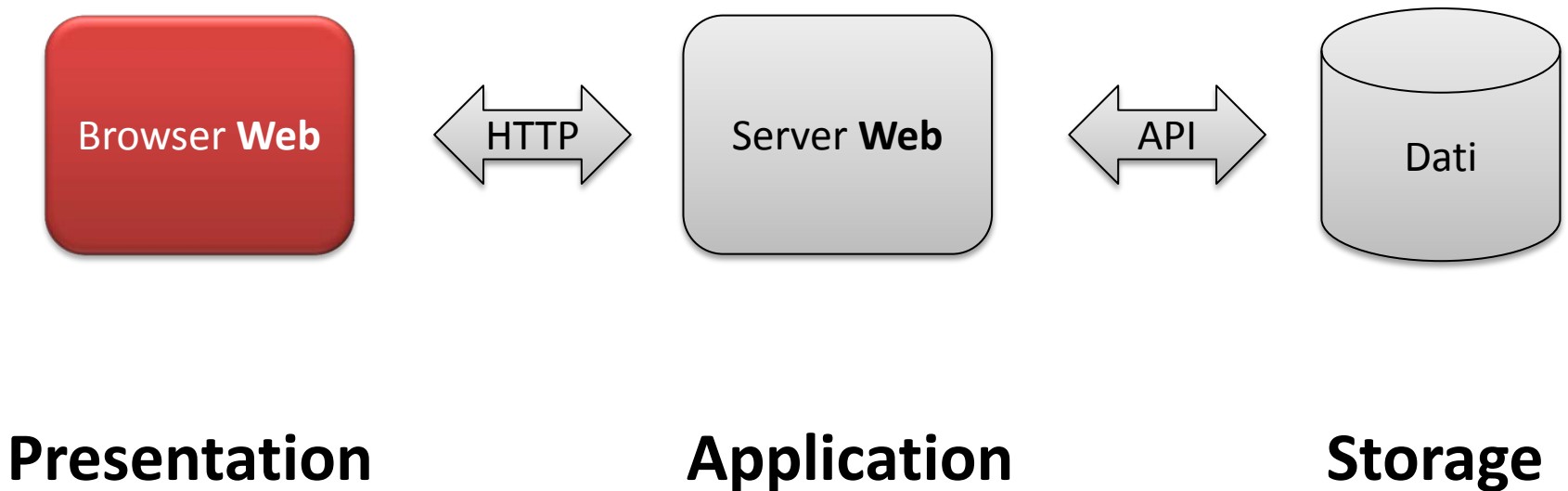


Laboratorio Progettazione Web

PHP e FORMs HTML

Andrea Marchetti – IIT-CNR
andrea.marchetti@iit.cnr.it
2013/2014

Struttura Applicazioni Web



Interazione con una applicazione Web

- Il **browser** costituisce il punto di interazione di un utente con una **applicazione web**
- Le **FORM html** sono il modo principale con cui l'utente riesce ad interagire
- Tramite le FORM html l'utente può inserire dei **parametri** che poi vengono inviati allo **script php** per essere elaborati
 - Al fine di fornire un risultato
 - Per essere inseriti in un DB
- Ciò che vedremo oggi è come far pervenire i parametri inseriti nelle form allo **script php**

Esempio

- Applicazione web che consente all'utente di
 - inserire una **parola** e di
 - selezionare un **colore**
- Il risultato è la parola nel colore selezionato.
- Pensiamo ad un FORM html che permette di inserire questi due PARAMETRI (parola, colore)

Esempio

L'attributo *action* individua lo script php che gestisce i parametri

```
<form action="esercizi01.php" method="get">  
  Parola <input type="text" name="parola" /></br>  
  <input type="radio" name="colore" value="green" />Verde  
  <input type="radio" name="colore" value="red" />Rosso</br>  
  <input type="submit" value="submit" />  
</form>
```

L'attributo *name* definisce il nome dei parametri

Parola

Verde Rosso

Parametri

Metodi GET e POST

```
<form action="esercizio1.php" method="get">
```

L'attributo *method* specifica se i parametri sono passati come *get* o *post*

- Il passaggio dei parametri della Form può avvenire secondo due modalità: **GET** e **POST**.
- Il metodo GET prevede il passaggio dei parametri in una QUERY STRING, ovvero una stringa che racchiude i parametri **accodata alla URL**
- Lunghezza massima querystring 256 caratteri.
- Nel metodo POST i parametri vengono passati direttamente tramite protocollo HTTP, non sono visibili.
- Sono metodi equivalenti, ma il GET non viene usato
 - in caso di password
 - in caso di molti parametri

Metodo GET

```
<form action="esercizio1.php" method="get">  
  Parola <input type="text" name="parola" /></br>  
  <input type="radio" name="colore" value="green" />Verde  
  <input type="radio" name="colore" value="red" />Rosso</br>  
  <input type="submit" value="submit" />  
</form>
```

- Con il metodo GET i parametri vengono passati nella URL che riferisce la pagina chiamata dalla ACTION

<http://localhost/esercizio1.php?parola=pluto&colore=rosso>

- Il carattere **?** separa il nome della pagina dai parametri che vengono passati.
- I parametri sono nella forma nomeparametro=valore separati dal simbolo **&**
- Tutta la stringa dopo il simbolo **?** prende il nome di **querystring** o stringa di interrogazione

Gestione dei parametri in PHP

- Si sfruttano degli **array associativi** definiti dal **sistema**
 - `$_POST['nomeparametro']` // per il metodo POST
 - `$_GET['nomeparametro']` // per il metodo GET
 - `$_REQUEST['nomeparametro']` // per entrambi i metodi.
-
- Con `$_REQUEST` lo script PHP è indipendente dal metodo usato dalla FORM HTML

Script PHP

```
<form action="esercizio1.php" method="get">  
    Parola <input type="text" name="parola" /></br>  
    <input type="radio" name="colore" value="green" />Verde  
    <input type="radio" name="colore" value="red" />Rosso</br>  
    <input type="submit" value="submit" />  
</form>
```

```
<?php
```

```
$parola = $_REQUEST['parola'];  
$colore = $_REQUEST['colore'];
```

```
echo "<p style='color: $colore'>$parola</p>";
```

```
?>
```

Tipi di form

- Input Text
- Text Area
- Radio Button
- Check Box
- Select

form.html

```
<form action="form.php" method="GET">  
  <!-- Inserisci qui le tue form -->  
</form>
```

form.php

```
<?PHP  
  /* Inserisci qui il tuo codice PHP */  
?>
```

Input Text

DESCRIZIONE:

Form che accetta testi alfanumerici

HTML:

```
<input type="text" name="descrizione" />
```

Parola

Altri attributi

- **size="15"** <!-- ampiezza della form sullo schermo in n. di caratteri del font corrente-->
- **maxlength="45"** <!-- massima lunghezza di input -->
- **value="value"** <!-- valore iniziale di default -->

PHP:

```
$colore=$_REQUEST["descrizione"];
```

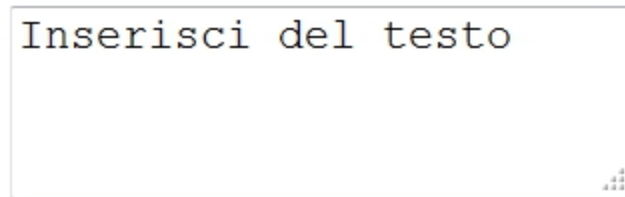
Textarea

DESCRIZIONE:

Form che accetta testi alfanumerici più estesi e organizzati in righe

HTML:

```
<textarea name="descrizione" />Inserisci del testo  
</textarea>
```



Altri attributi

- **rows="15"** <!-- n. di righe -->
- **cols = "45"** <!-- n. do colonne-->

In PHP

```
$colore=$_REQUEST["descrizione"];
```

Radio Button

DESCRIZIONE:

Usati quando si vuole selezionare un solo valore da una lista.
Importante occorre dare loro il medesimo nome

HTML:

```
<input type="radio" name="colore" value="green" />Verde  
<input type="radio" name="colore" value="red" />Rosso  
<input type="radio" name="colore" value="blue" />Blu
```

Verde Rosso Blue

In PHP

```
$colore=$_REQUEST["colore"];
```

CheckBox

DESCRIZIONE:

Usati quando si vuole selezionare uno o più valori da una lista.

Ogni checkbox ha il suo nome. Se si usa l'attributo *checked* il relativo form risulta selezionato. Per tutti i checkbox selezionati verrà inviato una stringa 'on' altrimenti non verrà inviato nulla. Con l'attributo *value* si può modificare la stringa restituita.

HTML:

```
<input type="checkbox" name="verde" checked="checked" />Verde  
<input type="checkbox" name="rosso" />Rosso  
<input type="checkbox" name="blu" />Blu
```

Verde Rosso Blu

In PHP

```
$verde=$_REQUEST["verde"]; //se selezionato contiene la stringa 'on'  
$rosso=$_REQUEST["rosso"]; //se selezionato contiene la stringa 'on'  
$blue =$_REQUEST["blue"]; //se selezionato contiene la stringa 'on'
```


Select

DESCRIZIONE:

Usato quando si vuole selezionare un solo valore da una lista. Simile ai radio button.

HTML:

```
<select name="coloreSelect" >
  <option value="red"    >rosso</option>
  <option value="green"  >verde</option>
  <option value="blue"   >blu</option>
</select>
```

In PHP

```
$colore=$_REQUEST["coloreSelect"]; //contiene il valore
                                   selezionato
```


Select Multipla

DESCRIZIONE:

Usato quando si vuole selezionare più di un valore da una lista. Simile ai check box. Anche in questo caso il parametro va dichiarato come array

HTML:

```
<select name="coloreSelect[]" multiple="multiple">
  <option value="red"    >rosso</option>
  <option value="green"  >verde</option>
  <option value="blue"   >blu</option>
</select>
```

In PHP

```
$colore=$_REQUEST["coloreSelect"]; //contiene un array di
    valori
```

Hidden Fields

DESCRIZIONE:

Usato quando si vuole salvare un dato utile per l'elaborazione successiva

HTML:

```
<input type="hidden" name="colore" value="red" />
```

In PHP

```
$colore=$_REQUEST["colore"];
```

Esercizio

- Fare una pagina HTML in cui si richiede:
 - Un testo da inserire
 - Un colore tra 5 opzioni
- Il programma PHP invocato deve produrre in risposta una pagina HTML con il testo introdotto visualizzato con il colore selezionato

Pagine per form

La gestione delle form prevede due passi

1. la visualizzazione della FORM in HTML

```
<form action="esercizi01.php" method="get">
    Parola <input type="text" name="parola" /></br>
    <input type="radio" name="colore" value="green" />Verde
    <input type="radio" name="colore" value="red" />Rosso</br>
    <input type="submit" value="submit" />
</form>
```

1. la gestione dei parametri in PHP.

```
<?php

$parola = $_REQUEST['parola'];
$colore = $_REQUEST['colore'];

echo "<p style='color: $colore'>$parola</p>";

?>
```

Pagine per form

```
<form action="esercizio1.php" method="get">
  Parola <input type="text" name="parola" /></br>
  <input type="radio" name="colore" value="verde" />Verde
  <input type="radio" name="colore" value="rosso" />Rosso</br>
  <input type="submit" value="submit" />
</form>
```

HTML

<http://localhost/esercizio1.php?parola=...&colore=...>

PHP

```
<?php
$parola = $_REQUEST['parola'];
$colore = $_REQUEST['colore'];

echo "<p style='color: $colore'>$parola</p>";

?>
```

Gestione delle form in PHP

- La prima fase del trattamento dei parametri in genere prevede un controllo sulle correttezza dei dati.
- Questo controllo può essere fatto in parte in javascript per una interazione più diretta con l'utente ma, in parte o completamente, può essere fatta anche in PHP e quindi lato server.
- Possiamo quindi aggiungere alla nostra pagina di gestione delle form una funzione di controllo campi immessi

Test di Geografia

DOMANDA
Qual è la capitale di **Italia**
 Parigi Roma Sofia

La scelta della nazione è casuale, così come le capitali proposte, una delle quali deve essere esatta

test_geo.php?domanda=Italia&risposta=Sofia

RISPOSTA
Risposta esatta
Risposta errata
La capitale di \$domanda è \$capitale

Tips
Consultare la pagina delle funzioni per gli array php

Test di Geografia - indizio 1

- Mi serve una array associativo \$geo
nazione=>capitale
- sizeof(\$geo) deve essere > 3
- Utile sia per creare la domanda che per controllare la risposta
- L'utente deve restituire due informazioni
 - la nazione oggetto della domanda
 - la capitale selezionata

Test di Geografia - indizio 2

- Tra **tutte** le coppie nazione=>capitale ne estraggo 3 che rappresentano i candidati alla domanda
- Tra questi 3 candidati ne estraggo 1 che rappresenta l'oggetto della domanda
 - \$domanda = rand(3)