

# **Introduzione allo sviluppo Web**

Studium Generale, a.a. 2013-2014, II semestre

# Contatto

- ▶ **Daniel Graziotin**
- ▶ **daniel.graziotin@unibz.it**
- ▶ **<http://ineed.coffee/teaching/introduzione-allo-sviluppo-web>**

# Esame

## ▸ **Esame ufficiale**

- Giovedì 26 Giugno, 17-19 E231

## ▸ **Invio progetti:**

- Venerdì 20 Giugno, mezzanotte
- [daniel.graziotin@unibz.it](mailto:daniel.graziotin@unibz.it)

# Esame

- ▶ **Formato**
- ▶ **Archivio (es: .zip) del sito**
  - Estratto nella root / del mio webserver
  - Autocontenuto (no file esterni)
  - NON date per scontato che si chiami localhost

# PHP

## I Form

**Moduli standard per passare informazioni dall'utente al sito sul server**

**Classici Form:**

- ▶ **Login e password**
- ▶ **Campo "Cerca" su un sito o su Google**
- ▶ **"A cosa stai pensando?" di Facebook**
- ▶ **Un sondaggio online**

# PHP

## I Form

**In HTML, un form si crea con**

```
<form method="metodo" action="pagina-destinazione">
```

**Serie di campi di testo, aree di testo, radio, checkbox**

```
<input type="submit" value="Invia" />
```

```
</form>
```

# PHP

## I Form

Guardiamo il file andata.php

```
<form method="GET" action="/ritorno.php">  
    Nome <input type="text" name="nome" /> <br/>  
    Cognome <input type="text" name="cognome" /> <br/>  
    E-mail <input type="text" name="email" /> <br/>  
    Frase segreta <input type="password" name="password" /> <br/>  
    Commenti <textarea name="commento"></textarea> <br/>  
    <input type="submit" value="Invia" />  
</form>
```

# PHP

## I Form

**andata.php**

**Aggiungiamo il seguente campo legato all'età**

**Età `<input type="text" name= "$eta" />` `<br/>`**



# PHP

## I Form

Guardiamo il file andata.php

```
<form method="GET" action="/ritorno.php">
```

```
    Nome <input type="text" name="nome" /> <br/>
```

```
    Cognome <input type="text" name="cognome" /> <br/>
```

```
    Età <input type="text" name="$eta" /> <br/>
```

```
    E-mail <input type="text" name="email" /> <br/>
```

```
    Frase segreta <input type="password" name="password" /> <br/>
```

```
    Commenti <textarea name="commento"></textarea> <br/>
```

```
    <input type="submit" value="Invia" />
```

```
</form>
```

# PHP

## I Form

**Andiamo su <http://localhost/andata.php>**

**Compiliamo il form con dei dati a caso e premiamo Invia**

# PHP

I Form

**Cosa succede?**

# PHP

## I Form

**Guardiamo la barra degli indirizzi del browser**

# PHP

## I Form

<http://localhost/ritorno.php?nome=Daniel&cognome=Graziotin&eta=27&email=daniel%40ined.coffee&password=LaMiaPassword&commento=Che+bel+sito>

# PHP

## I Form - GET

### GET

**/ritorno.php?nome=Daniel&cognome=Graziotin&eta=27&email=daniel%40ineed.coffee&password=LaMiaPassword&commento=Che+bel+sito HTTP/1.1**

**Accept-Language: en-GB,en;q=0.8,en-US;q=0.6,it;q=0.4,de;q=0.2**

**User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64)**

**AppleWebKit/537.36 (KHTML, like Gecko)**

**Chrome/36.0.1978.0 Safari/537.36**

# PHP

I Form - GET

<http://localhost/ritorno.php?nome=Daniel&cognome=Graziotin&eta=27&email=daniel%40ined.coffee&password=LaMiaPassword&commento=Che+bel+sito>

**Sezioniamo la richiesta HTTP GET**

# PHP

## I Form - GET

[http://localhost/ritorno.php?nome=Daniel&cognome=Graziotin  
&eta=27&email=daniel%40ineed.coffee&password=LaMiaPass  
word&commento=Che+bel+sito](http://localhost/ritorno.php?nome=Daniel&cognome=Graziotin&eta=27&email=daniel%40ineed.coffee&password=LaMiaPassword&commento=Che+bel+sito)

**http://localhost/ritorno.php**

? **nome** = Daniel

& **cognome** = Graziotin

& **\$eta** = 27

& **email** = daniel%40ineed.coffee

& **password** = LaMiaPassword

& **commento** = Che + bel + sito



# PHP

## I Form - GET

**method = "GET"**

**action="/ritorno.php"**

**name = "nome"**

**name = "cognome"**

**name = "\$eta"**

**name = "email"**

**name = "password"**

**name = "commento"**

**http://localhost/ritorno.php**

**? nome = Daniel**

**& cognome = Graziotin**

**& \$eta = 27**

**& email = daniel%40ineed.coffee**

**& password = LaMiaPassword**

**& commento = Che + bel + sito**

# PHP

## I Form - GET

? Indica l'inizio dei parametri

I parametri sono coppia  
nome = valore

& separa i parametri e  
loro valori (dal 2° in poi)

%40 rappresenta @

+ simboleggia lo spazio

**http://localhost/ritorno.php**

? **nome** = Daniel

& **cognome** = Graziotin

& **\$eta** = 27

& **email** = daniel%40ineed.coffee

& **password** = LaMiaPassword

& **commento** = Che + bel + sito

# PHP

## I Form - GET

**%40** rappresenta @

**+** simboleggia lo spazio  
(una volta era %20)

<http://en.wikipedia.org/wiki/Percent-encoding>

**http://localhost/ritorno.php**

? **nome** = Daniel

& **cognome** = Graziotin

& **email** = daniel%40ined.coffee

& **password** = LaMiaPassword

& **commento** = Che + bel + sito

# PHP

## I Form - GET

### **PHP analizza la richiesta HTTP GET**

**"http://localhost/ritorno.php?nome=Daniel&cognome=Graziotin&eta=27&email=daniel%40ineed.coffee&password=LaMiaPassword&commento=Che+bel+sito"**

**Scompatta la richiesta (è una stringa alla fine)**

**Crea un'array associativo (\$array["chiave"] = valore) chiamato \$\_GET (importante il trattino basso)**

# PHP

## I Form - GET

**Guardiamo ritorno.php**

```
$dati = $_GET;
```

```
print_r($dati);    // equivalente a print_r($_GET);
```

# PHP

## I Form - GET

### ritorno.php

```
$dati = $_GET;
```

```
//print_r($dati);           // equivalente a print_r($_GET);
```

```
$nome = $dati["nome"];
```

```
echo "Il tuo nome è ".$nome; // o $dati["nome"] o $_GET["nome"]
```

# PHP

I Form - GET

## ritorno.php

```
$dati = $_GET;
```

```
$nome = $dati["nome"];
```

```
$cognome = $dati["cognome"];
```

```
$eta = $dati["$eta"];
```

```
$email = $dati["email"];
```

```
$password = $dati["password"];
```

```
$commento = $dati["commento"];
```

# PHP

## I Form - GET

### ritorno.php

```
echo "<p>Il tuo nome è " . $nome. ", il tuo  
cognome è " . $cognome. "</p>";
```

```
echo "<p>La tua e-mail è " . $email. " e la tua  
password è " . $password. "</p>";
```

```
echo "<p>Hai lasciato il seguente commento:  
<br/> " . $commento. "</p>";
```



# PHP

## I Form - GET

### GET

- È comodo e veloce
- Lo usa anche Google!

<https://www.google.com/search?q=Studium+Generale>

**/search**

**? q = Studium + Generale**

La "pagina PHP" di Google (ma non è PHP)

La variabile *q* ha come valore "Studium Generale"

# PHP

## I Form - GET

### GET

- È comodo e veloce
- Lo usa anche Google!
- Ma trasmette tutto sulla barra degli indirizzi del browser

`http://localhost/ritorno.php?nome=Daniel&cognome=Graziotin  
&email=daniel%40ineed.coffee&password=LaMiaPassword&co  
mmento=Che+bel+sito`

- Non è il massimo per la sicurezza (non solo su monitor)
  - HTTPS

# PHP

## I Form - POST

### POST

- ▶ **Un metodo HTTP come GET (GET, POST, PUT, DELETE)**
- ▶ **Trasmette le informazioni nel corpo della richiesta HTTP**
  - GET la trasmette nell'indirizzo della richiesta
- ▶ **Chi guarda il browser non vede i valori**
- ▶ **Via HTTPS vengono criptati**

# PHP

## I Form - POST

### POST

- **Come si fa col codice? Velocissimo**
- **andata.php**

```
<form method= "POST" action="/ritorno.php">
```

- **(GET diventa POST, il resto uguale)**

# PHP

## I Form - POST

### POST

- **Come si fa col codice? Velocissimo**
- **andata.php**

```
<form method= "POST" action="/ritorno.php">
```

- **ritorno.php**

```
$dati = $_POST;
```

- **\$\_GET diventa \$\_POST, il resto uguale**
  - Ma solo perchè usiamo la variabile \$dati

# PHP

I Form - GET

**Funziona?**

**Cosa dice la barra degli indirizzi?**

# PHP

I Form - GET

**Funziona?**

**Cosa dice la barra degli indirizzi?**

<http://localhost/ritorno.php>

# PHP

I Form - GET

**POST /ritorno.php HTTP/1.1**

**Accept-Language: en-GB,en;q=0.8,en-US;q=0.6,it;q=0.4,de;q=0.2**

**User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64)**

**AppleWebKit/537.36 (KHTML, like Gecko) Chrome/36.0.1978.0**

**Safari/537.36**

**Content-Type: application/x-www-form-urlencoded**

**Content-Length: 102**

**nome=Daniel&cognome=Graziotin&email=daniel%40ineed.coffee&password=LaMiaPassword&commento=Che+bel+sito**



# PHP

## I Form - GET

**PHP analizza la richiesta intera HTTP POST**

**"http://localhost/ritorno.php"**

**Scompatta il corpo della richiesta (è una stringa alla fine)**

**Crea un'array associativo (\$array["chiave"] = valore) chiamato \$\_POST (importante il trattino basso)**

# PHP

## I Form

### Choose Question Type

Choose a question type by clicking on the type of response that you want for the question.

#### Available Types Sample

Label This is just a label.

Text

Text (encrypted)

Textarea

Radio  Option 1  Option 2  Option 3

Checkbox  Option 1  Option 2  Option 3

Single Select

Multiple Select   
Option 2  
Option 3  
Option 4

#### Summed Question What is your total monthly budget?

**Food:**

**Travel:**

**TOTAL:**

File Upload This will create a link to upload files.

Fonte:

<http://blog.leadmaster.com/2012/08/07/new-feature-responses-are-dependent-on-field-in-custom-forms/>

# PHP

## I Form

### Testo singola linea

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

### Password

```
<input type="password" name="nome" />
```

### Segno di spunta

```
<input type="checkbox" name="nome" />
```

# PHP

## I Form

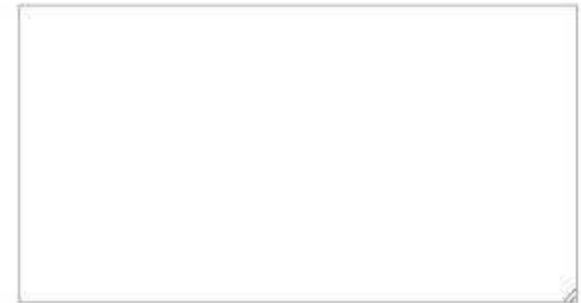
### Scelta singola tra opzioni

```
Valore 1<input type="radio"  
name="nomeGruppo" value="Valore 1" />  
Valore 2<input type="radio"  
name="nomeGruppo" value="Valore 2" />
```

Valore 1  Valore 2

### Area di testo

```
<textarea name="nome"></textarea>
```



### Inviare il form

```
<input type="submit" value="Spedisci" />
```

Spedisci

# PHP

## I Form

- ▶ **Ci sono altri moduli**
  - hidden, file, select, ...
  - Approfondimenti
- ▶ **HTML5 ne definisce a decine**

# PHP

## Strutture di controllo e cicli

- ▶ **Il bello della programmazione**
  - Prendere decisioni a seconda di certi valori
  - Automatizzare
- ▶ **PHP ha strutture di controllo e cicli**

# PHP

## Strutture di controllo – if-else

- **Prendere decisioni**

**Se l'utente ha meno di 18 anni, non gli mostriamo la frase sconcia. Altrimenti mostriamo la frase sconcia.**

**if e else**

# PHP

## Strutture di controllo – if-else

- **Prendere decisioni**

**Se l'utente ha meno di 18 anni, non gli mostriamo la frase sconcia. Altrimenti mostriamo la frase sconcia.**

**In ritorno.php**

```
if ($eta < 18) {  
    echo "Sei minorenne";  
} else {  
    echo "Sei maggiorenne: frase sconcia!";  
}
```



# PHP

## Strutture di controllo – if-else

- **If-else**
- **Forma**

```
if (condizione) {
```

```
    codice da eseguire se la condizione è VERA
```

```
} else {
```

```
    codice da eseguire se la condizione è FALSA
```

```
}
```

# PHP

## Strutture di controllo – booleani

- ▶ **Le condizioni sono espressioni booleane**
  - Tipo quelle matematiche, ma il risultato è VERO o FALSO (TRUE o FALSE)
- ▶ **Operatori più comuni**
  - UGUALE, MINORE , MAGGIORE, NON, E, O

# PHP

## Strutture di controllo – booleani

- **TRUE (vero), FALSE (falso)**

**Rappresentano gradi di verità.**

# PHP

## Strutture di controllo – booleani

- ▶ **TRUE (vero), FALSE (falso)**
- ▶ **Rappresentano gradi di verità.**
- ▶ **Se una variabile \$eta ha valore 18, vari "test" (proposizioni)**
  - \$eta è 18
  - \$eta è minore di 17
  - \$eta è maggiore di 19

**Hanno valori di verità (TRUE, FALSE, FALSE)**

# PHP

## Strutture di controllo – booleani

- ▶ **== (UGUALE)**
- ▶ **Testa se**
  - due entità hanno lo stesso valore di verità (TRUE o FALSE)
  - l'entità di sinistra ha il valore di quella di destra

# PHP

## Strutture di controllo – booleani

- ▶ **== (UGUALE)**
- ▶ **Testa se**
  - due entità hanno lo stesso valore di verità (TRUE o FALSE)
  - l'entità di sinistra ha il valore di quella di destra
- ▶ **18 == 18**  
**18 == 17**

# PHP

## Strutture di controllo – booleani

- ▶ **== (UGUALE)**
- ▶ **Testa se**
  - due entità hanno lo stesso valore o valore di verità (TRUE o FALSE)
  - l'entità di sinistra ha il valore di quella di destra
- ▶ **18 == 18 (TRUE)**  
**18 == 17 (FALSE)**
- ▶ **IMPORTANTISSIMO**
  - = assegna un valore a una variabile
  - == testa il valore di verità

# PHP

## Strutture di controllo – booleani

- ▶ **== (UGUALE)**
- ▶ **Testa se**
  - due entità hanno lo stesso valore di verità (TRUE o FALSE)
  - l'entità di sinistra ha il valore di quella di destra
- ▶ **\$eta = 18;**  
**\$eta == 19**  
**\$eta == 18**



# PHP

## Strutture di controllo – booleani

- ▶ **== (UGUALE)**
- ▶ **Testa se**
  - due entità hanno lo stesso valore di verità (TRUE o FALSE)
  - l'entità di sinistra ha il valore di quella di destra
- ▶ **\$eta = 18;**  
**\$eta == 19 (FALSE)**  
**\$eta == 18 (TRUE)**

# PHP

## Strutture di controllo – booleani

- ▶ **== (UGUALE)**
- ▶ **Testa se**
  - due entità hanno lo stesso valore di verità (TRUE o FALSE)
  - l'entità di sinistra ha il valore di quella di destra
- ▶ **\$nome = "daniel";**  
**\$nome == 19**  
**\$nome == \$eta // \$eta = 18**  
**\$nome == "daniel"**  
**\$nome == "Daniel"**

# PHP

## Strutture di controllo – booleani

- ▶ **== (UGUALE)**
- ▶ **Testa se**
  - due entità hanno lo stesso valore di verità (TRUE o FALSE)
  - l'entità di sinistra ha il valore di quella di destra
- ▶ **\$nome = "daniel";**

**\$nome == 19 (FALSE)**

**\$nome == \$eta (FALSE)**

**\$nome == "daniel" (TRUE)**

**\$nome == "Daniel" (FALSE)**

# PHP

## Strutture di controllo – booleani

- **< (MINORE)**  
**> (MAGGIORE)**
- **Assume valore TRUE se**
  - l'entità di sinistra è minore (maggiore) di quella di destra
  - FALSE altrimenti

**\$eta = 19;**

**\$eta < 18**

**\$eta > 18**

**\$eta < 19**

**\$eta > 19**

# PHP

## Strutture di controllo – booleani

- **< (MINORE)**  
**> (MAGGIORE)**
- **Assume valore TRUE se**
  - l'entità di sinistra è minore (maggiore) di quella di destra
  - FALSE altrimenti

**\$eta = 19;**

**\$eta < 18 (FALSE)**

**\$eta > 18 (TRUE)**

**\$eta < 19 (FALSE)**

**\$eta > 19 (FALSE)**

**\$eta è UGUALE a 19, né minore né maggiore**

# PHP

## Strutture di controllo – booleani

- **&& (AND)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **E** quella di destra è TRUE
  - "unisce più operazioni"

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "daniel" && $eta == 27
```

```
$nome == "Daniel" && $eta == 27
```

# PHP

## Strutture di controllo – booleani

- **&& (AND)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **E** quella di destra è TRUE
  - "unisce più operazioni"

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "daniel" && $eta == 27      (TRUE)
```

```
$nome == "Daniel" && $eta == 27     (FALSE)
```

# PHP

## Strutture di controllo – booleani

- **&& (AND)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **E** quella di destra è TRUE
  - "unisce più operazioni"

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "daniel" && $eta > 18
```

```
$nome == "daniel" && $eta < 18
```



# PHP

## Strutture di controllo – booleani

- **&& (AND)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **E** quella di destra è TRUE
  - "unisce più operazioni"

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "daniel" && $eta > 18 (TRUE)
```

```
$nome == "daniel" && $eta < 18 (FALSE)
```

# PHP

## Strutture di controllo – booleani

- **|| (OR)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **o** quella di destra è TRUE
  - Terra a terra: **almeno** una sola è TRUE

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "daniel" || $eta == 27
```

```
$nome == "Daniel" || $eta == 27
```

# PHP

## Strutture di controllo – booleani

- **|| (OR)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **o** quella di destra è TRUE
  - Terra a terra: **almeno** una sola è TRUE

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "daniel" || $eta == 27 (TRUE)
```

```
$nome == "Daniel" || $eta == 27 (TRUE)
```

# PHP

## Strutture di controllo – booleani

- **|| (OR)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **o** quella di destra è TRUE
  - Terra a terra: **almeno** una sola è TRUE

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "daniel" || $eta > 18
```

```
$nome == "daniel" || $eta < 18
```

# PHP

## Strutture di controllo – booleani

- **|| (OR)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **o** quella di destra è TRUE
  - Terra a terra: **almeno** una sola è TRUE

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "daniel" || $eta > 18           (TRUE)
```

```
$nome == "daniel" || $eta < 18           (TRUE)
```

# PHP

## Strutture di controllo – booleani

- **|| (OR)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) di sinistra è TRUE **o** quella di destra è TRUE
  - Terra a terra: **almeno** una sola è TRUE

```
$nome = "daniel"; $eta = 27;
```

```
$nome == "Daniel" || $eta < 18           (FALSE)
```

# PHP

## Strutture di controllo – booleani

- **! (NOT)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) che segue è FALSA
  - Terra a terra: "il contrario di"

```
$nome = "daniel"; $eta = 27;
```

```
!($nome == "Daniel")
```

```
!($nome == "daniel")
```

# PHP

## Strutture di controllo – booleani

- **! (NOT)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) che segue è FALSA
  - Terra a terra: "il contrario di"

```
$nome = "daniel"; $eta = 27;
```

```
!($nome == "Daniel")    (TRUE)
```

```
!($nome == "daniel")    (FALSE)
```



# PHP

## Strutture di controllo – booleani

- **! (NOT)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) che segue è FALSA
  - Terra a terra: "il contrario di"

```
$nome = "daniel"; $eta = 27;
```

```
!($eta == 27)
```

# PHP

## Strutture di controllo – booleani

- ▶ **! (NOT)**
- ▶ **Assume valore TRUE se**
  - l'entità (o operazione booleana) che segue è FALSA
  - Terra a terra: "il contrario di"

```
$nome = "daniel"; $eta = 27;
```

```
!($eta == 27)      (FALSE)
```

# PHP

## Strutture di controllo – booleani

- **! (NOT)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) che segue è FALSA
  - Terra a terra: "il contrario di"

```
$nome = "daniel"; $eta = 27;
```

```
!($nome == "daniel" && $eta > 18)
```

```
!($nome == "daniel" && $eta < 18)
```

# PHP

## Strutture di controllo – booleani

- **! (NOT)**
- **Assume valore TRUE se**
  - l'entità (o operazione booleana) che segue è FALSA
  - Terra a terra: "il contrario di"

```
$nome = "daniel"; $eta = 27;
```

```
!($nome == "daniel" && $eta > 18) (FALSE)
```

```
!($nome == "daniel" && $eta < 18) (TRUE)
```

# PHP

## Strutture di controllo – booleani

- **Esistono altri operatori**
- **!= >= <= XOR**

# PHP

## Strutture di controllo – if-else

- **Prendere decisioni**

**Se l'utente ha meno di 18 anni, non gli mostriamo la frase sconcia. Altrimenti mostriamo la frase sconcia.**

**In ritorno.php**

```
if ($eta < 18) {  
    echo "Sei minorenne";  
    // altri echo  
} else {  
    echo "Sei maggiorenne: frase sconcia!";  
    // altri echo  
}
```

# PHP

## I cookies

- ▶ **Cookie HTTP, questo mistero!**
- ▶ **I cookie sono delle informazioni scambiate tra il browser e il sito Web (il server)**
- ▶ **Stringhe che possono accompagnare le richieste (e le risposte) HTTP**
- ▶ **Generati dal server e (solitamente) memorizzati nel client**

# PHP

## I cookies

- ▶ **Un classico: quando si fa il login su un sito Web, il server genera un cookie che contiene l'informazione "Su questo browser c'è un utente chiamato Daniel. Questa informazione scadrà tra X ore"**

**HTTP/1.0 200 OK**

**Content-type: text/html**

**Set-Cookie: nome=Daniel; Expires=Mon, 30 Jun 2014 10:18:14 GMT**

**"Caro browser, salva questo cookie nome=Daniel e rimandamelo quando torni da me"**



# PHP

## I cookies

**HTTP/1.0 200 OK**

**Content-type: text/html**

**Set-Cookie: nome=Daniel; Expires=Mon, 30 Jun 2014 10:18:14**

**GMT**

**<html>.....**

**"Caro browser, salva questo cookie nome=Daniel e rimandamelo quando torni da me"**

**GET /index.php HTTP/1.1**

**Cookie: name=Daniel;**

# PHP

## I cookies

- **Scrivere un cookie in PHP**

```
setcookie("nome del cookie", "valore", time() + secondi);
```

- **time() ritorna l'ora corrente in secondi (dal 1 Gennaio 1970)**
- **Aggiungiamo + *secondi* per specificare quando far scadere il cookie.**

```
setcookie("nome", "Daniel", time() + 3600);  
// cookie nome = Daniel che scadrà tra un'ora
```

# PHP

## I cookies

- ▶ **Leggere un cookie in PHP**
- ▶ **PHP decodifica i cookie per noi e li mette nell'array associativa speciale, `$_COOKIE`; (come `$_GET`, `$_POST`)**
- ▶ **Per leggere il cookie settato prima, useremmo `$_COOKIE["nome"]`;**

# PHP

## I cookies

**Importante! Diversamente dai form, i cookie sono a 3 "caricamenti di pagina" la prima volta**

- 1. L'utente fa qualcosa (es: riempie una form) e manda le informazioni al server**
- 2. Il server riceve e chiede al browser di memorizzare un cookie**
- 3. Il browser usa il cookie **dalla prossima volta****

# PHP

`empty()`

**Per controllare se**

- **L'utente ha inserito un valore nella form con GET o POST**
- **Esiste un cookie chiamato X**

**Si usa la funzione PHP `empty($variabile)`**

**`empty($variabile)` ritorna TRUE se `$variabile` è vuota o non esiste**

**`empty( $_COOKIE["nome"] )` è TRUE se non esiste il cookie "nome"**

# PHP

`empty()`

**`empty($variabile)` ritorna TRUE se `$variabile` è vuota o non esiste**

**`empty( $_COOKIE["nome"] )` è TRUE se non esiste il cookie "nome"**

**Di conseguenza, la si usa con l'operatore ! (NOT), in modo da poter scrivere**

```
if (! empty($_COOKIE["nome"] ) {  
    // se il cookie "nome" non è vuoto  
    // fai qualcosa, es: usa il valore  
}
```

# PHP

## Strutture di controllo – if-else

**ritorno.php**

```
$nome = $dati["nome"];
```

```
setcookie("nomeVisitatore", $nome, time() + (2 * 60));
```

```
// 2 minuti
```

# PHP

## Strutture di controllo – if-else

**index.php**

```
<?php
```

```
if( !empty( $_COOKIE["nomeVisitatore"] ) ){
```

```
    echo "Ciao, " . $_COOKIE["nomeVisitatore"];
```

```
}
```

```
?>
```



# PHP

## Strutture di controllo – if-else

- **Proviamo la magia**

# PHP

## Strutture di controllo – if-else

- ▶ **TRUCCO**

# PHP

## Strutture di controllo – if-else con HTML misto

```
<?php  
if ($eta < 18) {  
?>
```

**Sei minorenne**

```
<?php  
} else {  
?>
```

**Sei maggiorenne: frase sconcia!**

```
<?php  
}  
?>
```