

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

Lezione da recuperare

- 17 Giugno
- 17-19
- E231

▸ **Markup**

- HTML

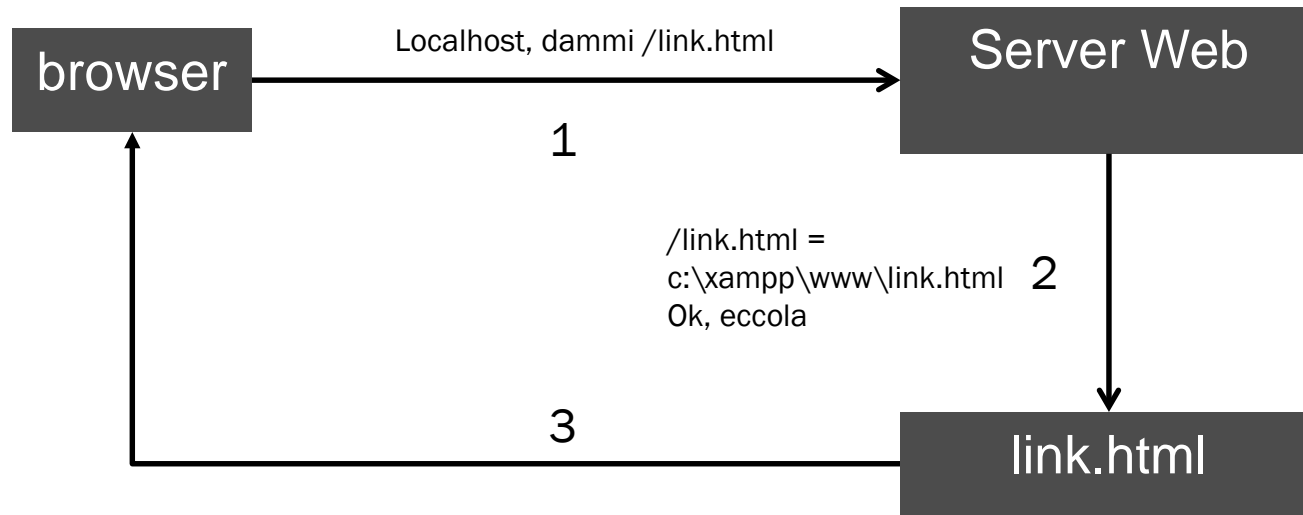
▸ **Stile**

- CSS

▸ **Dinamicità**

- **Server: PHP**

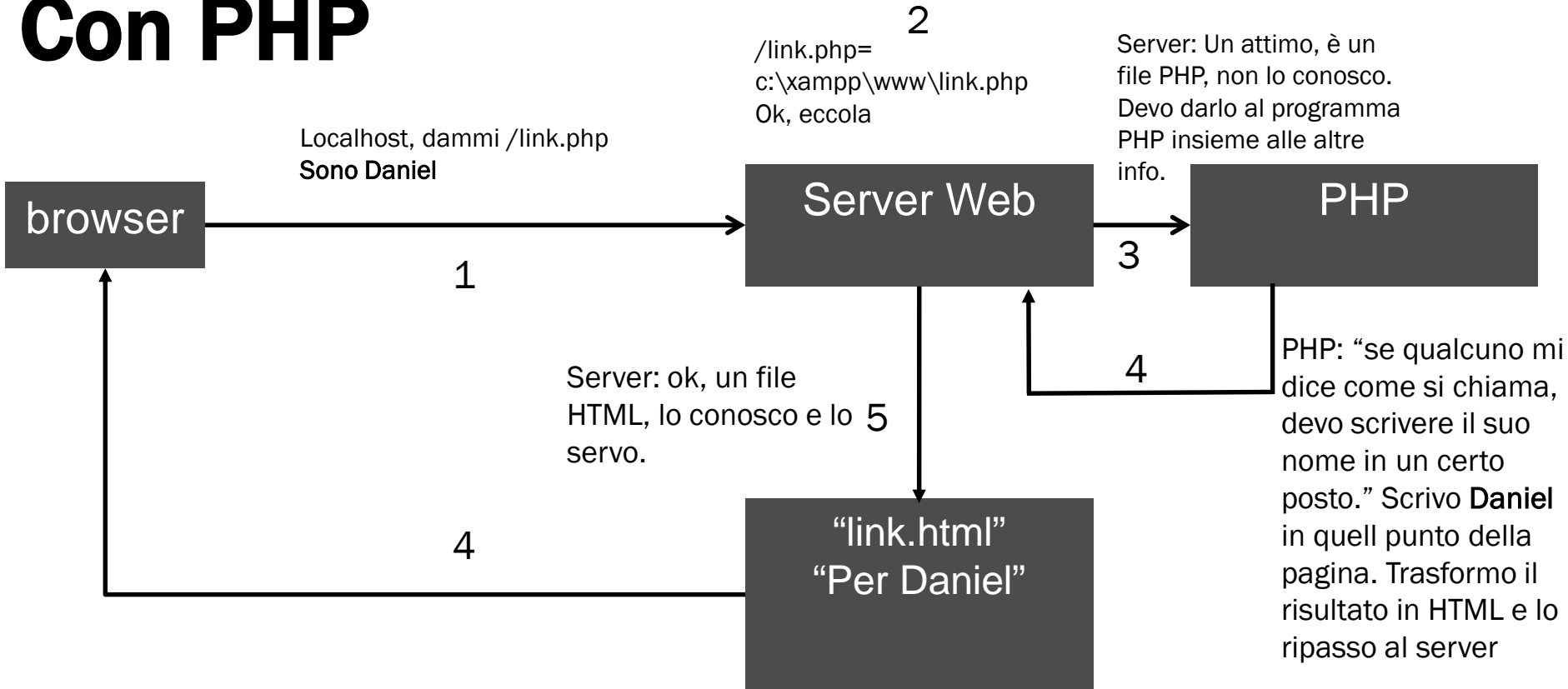
Con HTML



Risultato

Ecco [qui](#) la foto di un gattino

Con PHP



Risultato

Ecco [qui](#) la foto di un gattino, Daniel

PHP

PHP: Hypertext Preprocessor

- ▶ **Acronimo (ricorsivo) PHP: Hypertext Preprocessor**
- ▶ **Linguaggio di programmazione orientato al Web**
- ▶ **Nella top 10 dei linguaggi più popolari al mondo di sempre (Fonte: [Tiobe Index](#))**
- ▶ **Gratuito e open-source**

PHP

PHP: Hypertext Preprocessor

- ▶ **Interpretato nel server**
 - Mai nel client
- ▶ **Genera pagine HTML dinamicamente**
 - L'essenza è questa
 - Genera varianti di una data pagina HTML a seconda di certe condizioni

PHP

PHP: Hypertext Preprocessor

- ▶ **Dove si scrive il codice PHP?**
- ▶ **Per i principianti, direttamente nelle pagine HTML**
- ▶ **..con estensione .php**

PHP

Facciamo spazio

- ▶ **Il corso qui dá per scontato che la radice del vostro server Web sia pulita**
- ▶ **Nessun file, nessuna cartella dentro**
- ▶ **Fate pulizia!**

PHP

Hello World

- ▶ **Scaricate il materiale della [Lezione 12](#)**
- ▶ **Scompattatelo nella radice del server Web**
- ▶ **Accedete a <http://localhost/index.php>**

PHP

Hello World

- ▶ **Cosa vedete di diverso?**

PHP

Hello World

- **Cercate nel codice di index.php**

<h1>Chi Siamo</h1>

PHP

Hello World

- Cercate

```
<h1>Chi Siamo</h1>
```

- Modificatelo così

```
<h1><?php echo "Chi Siamo"; ?></h1>
```

PHP

Hello World

- **Risultato?**
- **Guardiamo nel browser**

PHP

Hello World

- Il codice PHP sta **SEMPRE** tra i suoi «tag»

<?php e ?>

- La maggior parte delle linee scritte in PHP sono istruzioni, nella forma

istruzione;

istruzione;

istruzione;

PHP

echo

- ▶ **Una specie di funzione del linguaggio**

- Serve a scrivere nell'HTML

- ▶ **Forma**

echo "frase da scrivere";

- ▶ **"frase da scrivere" è una stringa**

- "un insieme di caratteri"
- Una frase (ma non dite a nessuno che ve l'ho detto 😊)

PHP

echo scrive davvero solo testo nell'HTML

<p>

<?php

echo "Lorem";

echo "ipsum";

echo "dolor";

echo "sit";

echo "amet";

?>

</p>



<p>Loremipsumdolorsitamet</p>

PHP

echo scrive davvero solo testo nell'HTML

<p>

<?php

echo "Lorem ";

echo "ipsum ";

echo "dolor ";

echo "sit ";

echo "amet ";

?>

</p>



<p>Lorem ipsum dolor sit amet</p>

PHP

echo scrive davvero solo testo nell'HTML

```
<p>
```

```
<?php
```

```
echo "Lorem ";
```

```
echo "ipsum<br/>";
```

```
echo "dolor ";
```

```
echo "sit ";
```

```
echo "amet ";
```

```
?>
```

```
</p>
```



```
<p>Lorem ipsum<br>  
dolor sit amet</p>
```

PHP

echo

<i>Echo Statement</i>	<i>PHP Output</i>	<i>Web Page Display</i>
<code>echo "Hello World!";</code>	Hello World!	Hello World!
<code>echo "Hello";</code> <code>echo "World!";</code>	HelloWorld!	HelloWorld!
<code>echo "Hello\nWorld!";</code>	Hello World!	Hello World!
<code>echo "Hello
World!";</code>	Hello World	Hello World!
<code>echo "Hello
\nWorld!";</code>	Hello World!	Hello World!

Fonte: Janet Valade, PHP5 For Dummies

PHP

echo

- ▶ **In echo, il punto «.» concatena le cose**

```
echo "ciao " . "come " . "stai?";
```

- ▶ **È come**

```
echo "ciao come stai?";
```

PHP

commenti

- ▶ **Anche PHP ha dei commenti**
- ▶ **Su una linea sola cominciano con // oppure con #**

//questo è un commento

```
echo "ciao"."come"."stai?";
```

```
# ma anche questo
```

- ▶ **Su più linee, cominciano con /* e finiscono con */**

```
/* questo codice non viene mai eseguito
```

```
echo "ciao come stai?";
```

```
*/
```


PHP

variabili

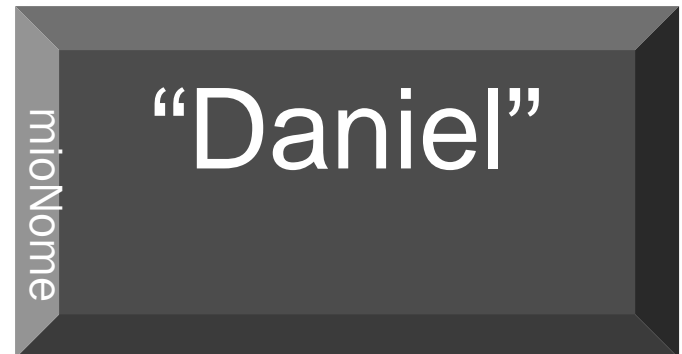
- ▶ **Le variabili sono dei contenitori per memorizzare dei dati**
 - Esempio: stringhe o numeri
- ▶ **Sono come delle scatole di memoria per metterci dentro cose**
- ▶ **Come le buone scatole in soffitta, hanno una etichetta per distinguerle tra loro, un nome**
 - `$nome`

PHP

variabili

- ▶ Per **assegnare** un valore a una variabile si usa l'istruzione
- ▶ **\$nome = valore;**

```
$mioNome = "Daniel";
```



PHP

variabili

- ▶ **Per stampare il valore di una variabile, si usa sempre echo**

<p>

<?php

```
$mioNome = "Daniel";
```

```
echo "il mio nome è ";
```

```
echo $mioNome;
```

?>

</p>

PHP

variabili

- ▶ **Per stampare il valore di una variabile, si usa sempre echo**

<p>

<?php

\$mioNome = "Daniel";

echo "il mio nome è ".\$mioNome;

?>

</p>

PHP

variabili

- ▶ **Anche i numeri sono valori validi per una variabile**

```
$numeroAcaso = 213;
```

PHP

Esercizio

- ▶ **Scrivete in PHP la seguente frase**
- ▶ **Le parole in arancione devono essere variabili con valore appropriato**

Il mio nome è VOSTRONOME, abito a VOSTROPAESE, ho VOSTRAETÀ anni e in tasca ho NUMERO euro

PHP

variabili

```
$mioNome = "Daniel";  
  
$eta = 27; // non si usano accenti  
  
$abito_a = "Bolzano"; // camelCase o _?  
  
$euroInTasca = 4.45; // separatore decimale  
  
echo "Il mio nome è ".$mioNome.", abito a  
".$abito_a.", ho ".$eta."anni e in tasca ho  
".$euroInTasca." euro";
```

PHP

Esercizio

- ▶ **Si può assegnare a una variabile il valore di un'altra variabile**

```
$verde= "#00FF00";
```

```
$coloreAlbero = $verde;
```

```
$coloreBanconoteDollari = $verde;
```


PHP

I tipi di dato

- ▶ **Stringhe**
 - "Hello world"
- ▶ **Numeri interi (integer)**
 - 1936
- ▶ **Numeri con la virgola (float)**
 - 1936.27
- ▶ **Booleani (vero e falso)**
 - TRUE oppure FALSE

PHP

Operazioni con i numeri

- **Addizione, sottrazione, moltiplicazione, divisione**

- $+ - * /$

- `$tre = 1 + 2; // 3`

- `1 + 2` è una **operazione**

PHP

Operazioni con i numeri

- ▶ **Addizione, sottrazione, moltiplicazione, divisione**
 - + - * /
 - `$tre = 1 + 2; // 3`
 - 1 + 2 è una **operazione**
- ▶ **Usare sempre le parentesi per controllare l'ordine,**
 - `$risultato = 1 + 2 * 3 + 4; // 11`
 - `$risultato = (1 + 2) * (3 + 4); // 21`

PHP

Operazioni con i numeri

- ▶ **Addizione, sottrazione, moltiplicazione, divisione, ..**

- + - * /

- `$tre = 1 + 2; // 3`

- `1 + 2` è una **operazione**

- ▶ **Usare sempre le parentesi per controllare l'ordine,**

- `$risultato1 = 1 + 2 * 3 + 4; // 11`

- `$risultato2 = (1 + 2) * (3 + 4); // 21`

- `$risultato3 = $risultato1 - $risultato2; // -10`

PHP

Procedure e funzioni

- ▶ **Gruppo di istruzioni che agiscono su dei parametri**
- ▶ **Esempio**

```
// Per sommare 1 e 2 uso
```

```
$numero1 = 1;
```

```
$numero2 = 2;
```

```
$somma = $numero1 + $numero2;
```

```
echo $somma;
```

PHP

Procedure e funzioni

- ▶ **Gruppo di istruzioni che agiscono su dei parametri**
- ▶ **Esempio**

```
// Per sommare 3 e 4 uso
```

```
$numero1 = 3;
```

```
$numero2 = 4;
```

```
$somma = $numero1 + $numero2;
```

```
echo $somma;
```

PHP

Procedure e funzioni

- ▶ **Gruppo di istruzioni che agiscono su dei parametri**
- ▶ **Esempio**
- ▶ **CHE NOIA.**
- ▶ **Una procedura fa il lavoro per noi!**

PHP

Procedure e funzioni

- ▶ **Gruppo di istruzioni che agiscono su dei parametri**
- ▶ **Esempio**

```
// Per sommare due numeri uso
```

```
function addizione($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    echo $somma;  
}
```


PHP

Procedure e funzioni

```
// Per sommare due numeri uso  
function addizione($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    echo $somma;  
}  
  
addizione(1 , 2); // scrive 3  
addizione(3, 4); // scrive 7  
addizione(5, 5); // scrive 10
```

PHP

Procedure e funzioni

- ▶ **addizione()** «**stampa**» solo il risultato.
- ▶ **Sarebbe più comodo se restituisse il numero**
 - Possiamo stamparlo o usarlo in altro modo
- ▶ **Funzioni**

```
// restituisce la somma di due numeri
```

```
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}  
  
somma(1, 2); // non scrive niente
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}  
  
$risultato = somma(1, 2);  
echo $risultato;           // scrive 3
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}  
  
$risultato = somma(3, 4);  
echo $risultato;           // scrive 7
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}
```

```
$risultato = somma(1, 2) + somma(3, 4)  
echo $risultato;           // scrive ???
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}  
  
$risultato = somma(1, 2) + somma(3, 4)  
echo $risultato;           // scrive 10
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}
```

```
$risultato = somma(1, 2) * somma(3, 4)  
echo $risultato;           // scrive ???
```


PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}  
  
$risultato = somma(1, 2) * somma(3, 4)  
echo $risultato;           // scrive 21
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}
```

```
$risultato = somma(somma(1, 2), 3);  
echo $risultato;           // scrive ???
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}  
  
$risultato = somma(somma(1, 2), 3);  
echo $risultato;           // scrive 6
```

PHP

Procedure e funzioni

```
// restituisce la somma di due numeri  
function somma($numero1, $numero2) {  
    $somma = $numero1 + $numero2;  
    return $somma;  
}
```

```
echo somma(somma(1, 2), 3);           // scrive 6
```

PHP

Array

- ▶ **Collezioni di dati sotto un unico nome**
 - Costo della spesa nei 12 mesi
 - Nomi dei quadri in un museo
 - Nome degli studenti del corso ISW

PHP

Array

- ▶ **Collezioni di dati sotto un unico nome**
 - Nome degli studenti del corso ISW
- ▶ **Invece che**
\$studente1 = "Alessandra";
\$studente2 = "Markus";
\$studente3 = "Naomi";

PHP

Array

- ▶ **Collezioni di dati sotto un unico nome**

- Nome degli studenti del corso ISW

- ▶
`$studenti[0] = "Alessandra";`

- `$studenti[1] = "Markus";`**

- `$studenti[2] = "Naomi";`**

- ▶ **La variabile array è `$studenti`, i numeri tra parentesi quadra di chiamano **chiavi** e servono per accedere agli elementi**

PHP

Array

- ▶ **Collezioni di dati sotto un unico nome**
 - Nome degli studenti del corso ISW
- ▶ **Invece che**
\$studenti[0] = "Alessandra";
\$studenti[1] = "Markus";
\$studenti[2] = "Naomi";
- ▶ **La variabile array è \$studenti, i numeri tra parentesi quadra di chiamano **chiavi****

```
echo $studenti[0]; // scrive Alessandra
```


PHP

Array associativi

- ▶ **Gli array con chiavi numeriche preservano l'ordine degli elementi**
- ▶ **Gli array **associativi** hanno stringhe come chiavi e non preservano l'ordine**

```
$capitali["italia"] = "Roma";
```

```
$capitali["germania"] = "Berlino";
```

```
$capitali["francia"] = "Parigi";
```

```
echo $capitali["italia"]; // scrive Roma
```

PHP

Array associativi

- ▶ **Per stampare il contenuto intero di una array (per motivi di ispezione) non si usa echo, ma print_r()**
- ▶ **print_r(\$array) è una funzione provveduta da PHP**
 - \$array è l'array da stampare

print_r(\$capitali);

PHP

Modificare Array

▸ **Modificare un elemento**

- `$nomeArray[chiave] = valore` // stringhe, numeri, ..

▸ **Cancellare un elemento**

- `unset($nomeArray[chiave]);` // le chiavi vengono scalate se numeri

▸ **Contare gli elementi**

- `$n = count($nomeArray);` // n contiene la dimensione

▸ **Ordinare un array**

- `sort($nomeArray);` // per array numerici
- `asort($nomeArray);` // per array associativi

PHP

Attraversare un Array

- ▶ **Per «fare qualcosa» per ogni elemento di un array**
 - Si usano molte tecniche
 - Una di queste è foreach

// vedremo i cicli la prossima volta

```
foreach ( $capitali as $citta ) {  
    echo $citta;  
    echo " <br/>";  
}
```

Scrive Roma
 Berlino
 Parigi

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

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>
```