

Tópicos Avançados em Sistemas de Informação

Aula 11 *PHP*



Prof. Me Ivan José dos Reis Filho

Aula anterior

- PHP
 - Tipos de Dados
 - Strings
 - Constantes
 - Operadores
 - If ... Else ... Else if
 - Switch
 - While Loops
 - For Loops



Aula de hoje

- PHP
 - Funções
 - Arrays
 - Sorting Arrays
 - Superglobals



Funções

- O verdadeiro poder do PHP vem de suas funções; Possui mais de 1000 funções internas.
 - Além das funções integradas do PHP, podemos criar nossas próprias funções.
 - Uma função é um bloco de instruções que podem ser usadas repetidamente em um programa.
 - Uma função não será executada imediatamente quando uma página é carregada.
 - Uma função será executada por uma chamada para a função.

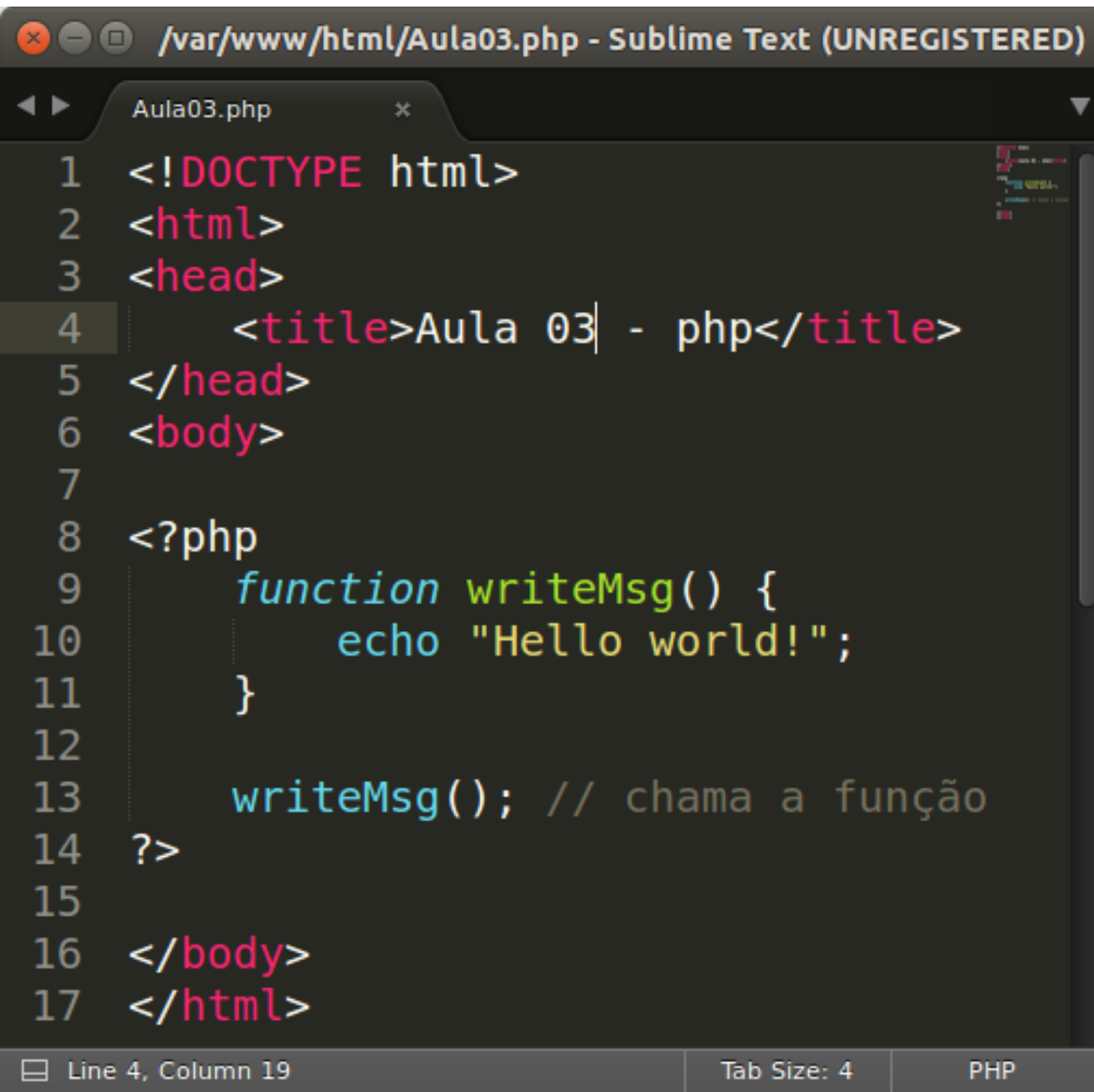
Funções

- Uma declaração de função definida pelo usuário começa com a palavra "function":
- Sintaxe:

```
function functionName() {  
    code to be executed;  
}
```

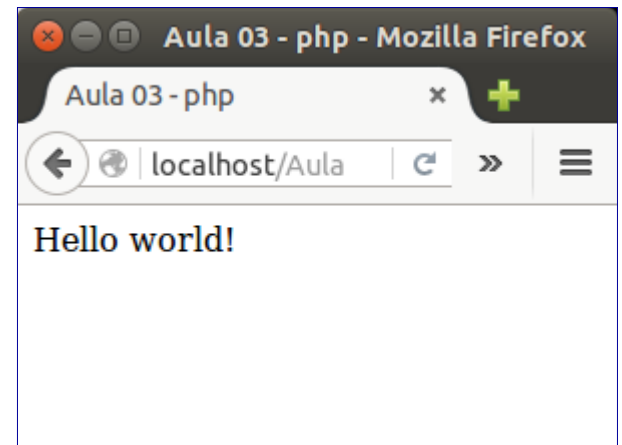
Função

Exemplo



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7
8 <?php
9     function writeMsg() {
10         echo "Hello world!";
11     }
12
13     writeMsg(); // chama a função
14 ?>
15
16 </body>
17 </html>
```

Line 4, Column 19 Tab Size: 4 PHP



Funções

- Argumentos

- As informações podem ser passadas para funções através de argumentos.
- Um argumento é como uma variável.
- Argumentos são especificados após o nome da função, dentro dos parênteses.
- Você pode adicionar tantos argumentos quanto quiser, basta separá-los com uma vírgula.

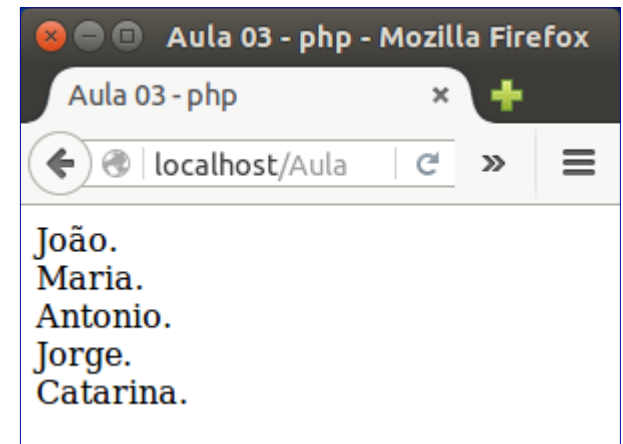
Função

Um Argumento

```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)

Aula03.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8
9     function familyName($nome) {
10         echo "$nome.<br>";
11     }
12
13     familyName("João");
14     familyName("Maria");
15     familyName("Antonio");
16     familyName("Jorge");
17     familyName("Catarina");
18 ?>
19 </body>
20 </html>
```

Line 18, Column 5; Saved /var/www/html/Aula03.php (UTF-8) Tab 9



Função

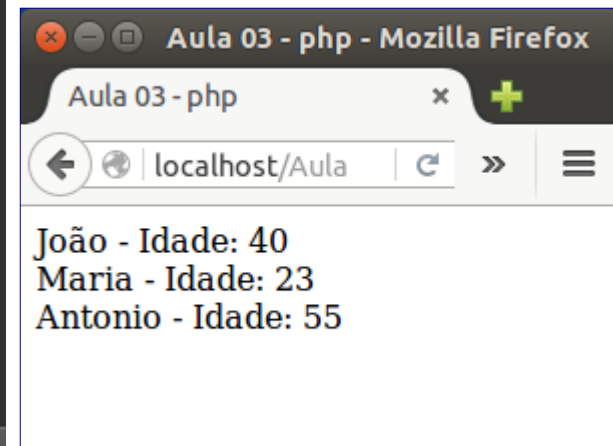
Dois Argumentos

```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)

Aula03.php

1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8
9     function familyName($nome, $idade){
10         echo "$nome - Idade: $idade<br>";
11     }
12
13     familyName("João", "40");
14     familyName("Maria", "23");
15     familyName("Antonio", "55");
16
17 ?>
18 </body>
19 </html>
```

Line 10, Column 24; Saved /var/www/html/Aula03.php (UTF-8) Tab Size: 4 P



Funções

- Argumentos
 - O exemplo a seguir mostra como usar um parâmetro padrão.
 - Se chamamos a função `setHeight ()` sem argumentos, é necessário o valor padrão como argumento

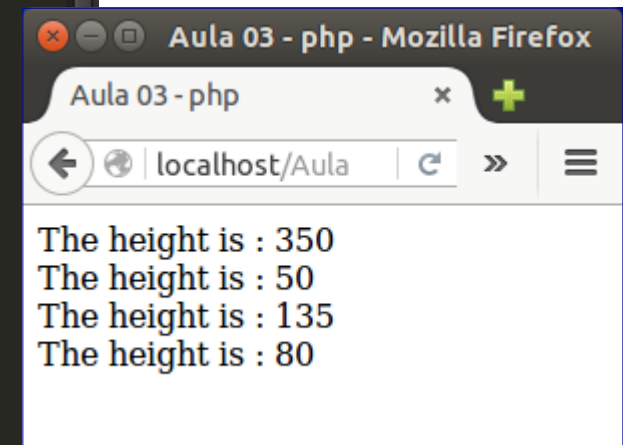
Função

Valor do argumento padrão

```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)

Aula03.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8     function setHeight($minheight = 50) {
9         echo "The height is : $minheight <br>
10             ";
11     }
12     setHeight(350);
13     setHeight(); // Valor padrão será 50
14     setHeight(135);
15     setHeight(80);
16 ?>
17 </body>
18 </html>
```

Line 17, Column 8 Tab Size: 4 PHP



Função

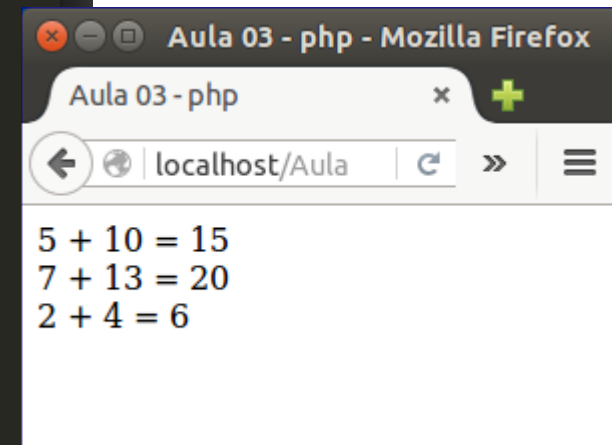
Retornando valor

```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)

Aula03.php

1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8     function soma($x, $y) {
9         $z = $x + $y;
10        return $z;
11    }
12
13    echo "5 + 10 = " . soma(5, 10) . "<br>";
14    echo "7 + 13 = " . soma(7, 13) . "<br>";
15    echo "2 + 4 = " . soma(2, 4);
16 ?>
17 </body>
18 </html>
```

Line 15, Column 27



Agenda

- PHP
 - Funções
 - Arrays
 - Sorting Arrays
 - Superglobals

Arrays

- Um *Array* armazena vários valores em uma única variável:
 - Você pode armazenar uma lista de itens (uma lista de nomes de carros, por exemplo), em apenas uma variável.
 - Exemplo:

```
$cars = array("Volvo", "BMW", "Toyota");
```

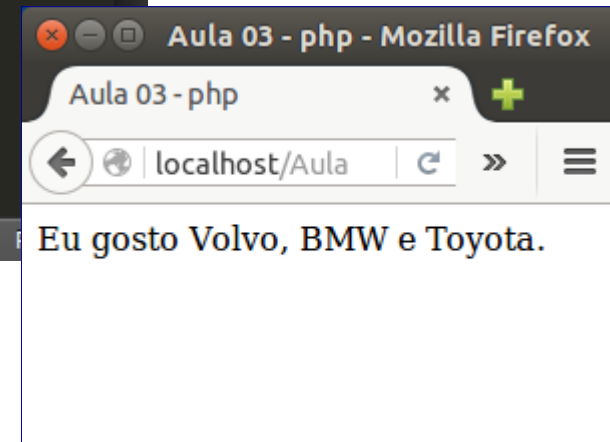
Array

Exemplo

```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)

Aula03.php x
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8
9 $cars = array("Volvo", "BMW", "Toyota");
10
11 echo "Eu gosto " . $cars[0] . ", " . $cars[1]
    . " e " . $cars[2] . ".";
12 ?>
13 </body>
14 </html>
```

Line 14, Column 8 Tab Size: 4



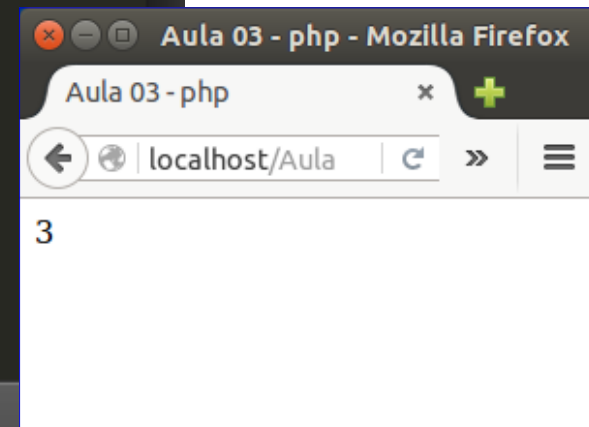
Array

Contador (count)

```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)

Aula03.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8
9 $cars = array("Volvo", "BMW", "Toyota");
10
11 echo count($cars);
12
13 ?>
14 </body>
15 </html>
```

Line 15, Column 8 Tab Size: 4



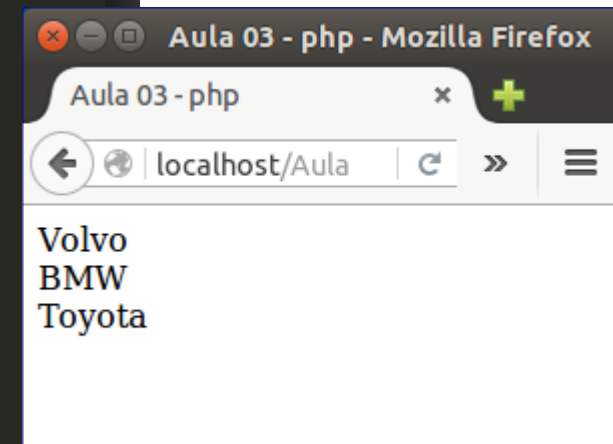
Array

exibição de todos elementos - for

```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)

Aula03.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7     <?php
8         $cars = array("Volvo", "BMW", "Toyota");
9         $tamanho = count($cars);
10
11         for($x = 0; $x < $tamanho; $x++)
12         {
13             echo $cars[$x];
14             echo "<br>";
15         }
16     ?>
17 </body>
18 </html>
```

Line 15, Column 5



Arrays associativo

- *Arrays* associativos são arrays que usam chaves nomeadas que você atribui a eles.
 - Exemplo:

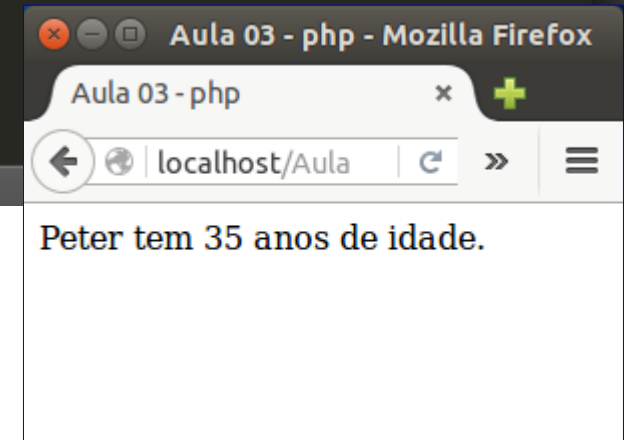
```
$idade = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
```

Array

Associativo

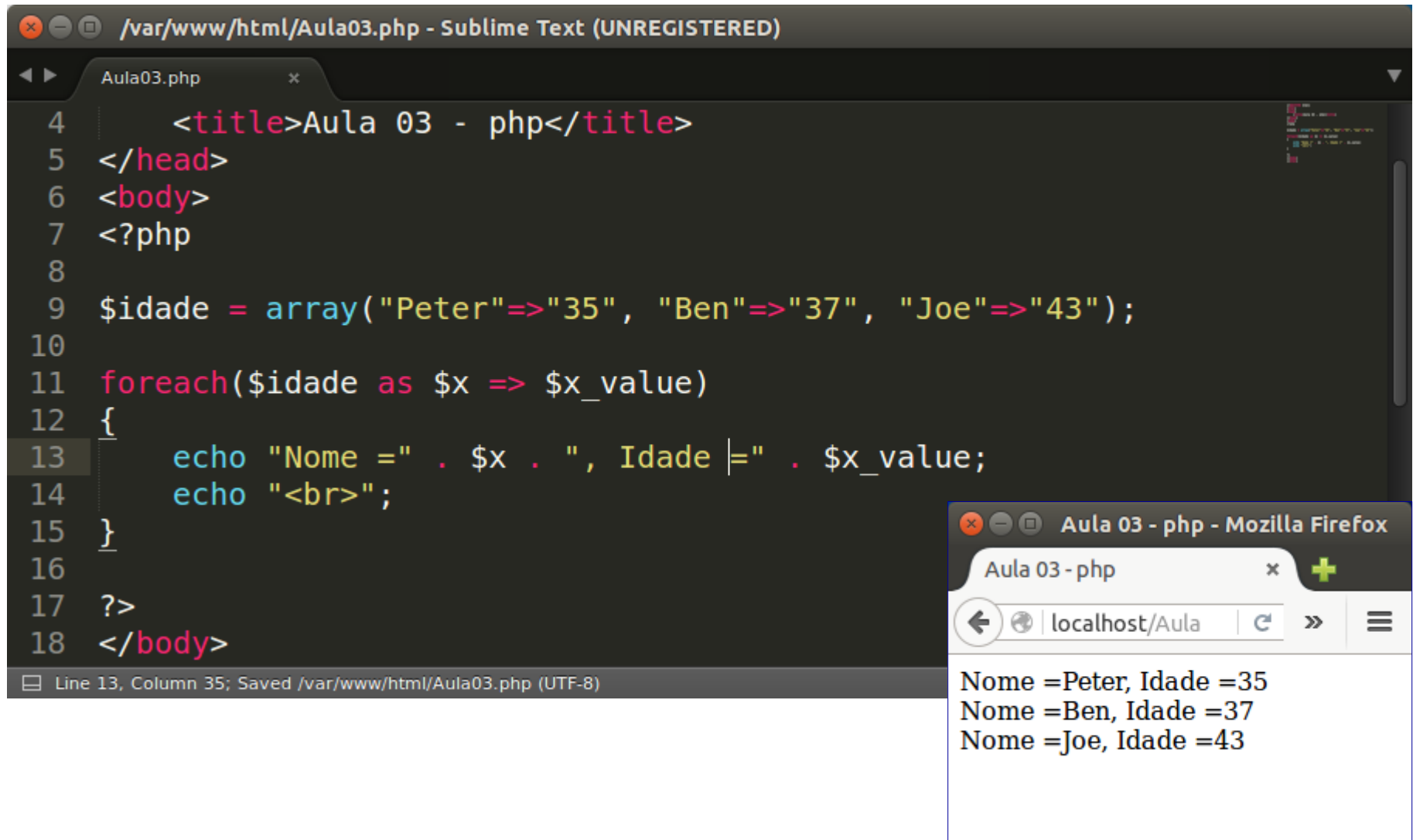
```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)
Aula03.php
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8
9 $idade = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
10
11 echo "Peter tem " . $idade['Peter'] . " anos de idade.";
12
13 ?>
14 </body>
15 </html>
```

Line 15, Column 8



Array

Para percorrer e imprimir todos os valores de uma matriz associativa, você pode usar um loop **foreach**



```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)
Aula03.php
4 <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8
9 $idade = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
10
11 foreach($idade as $x => $x_value)
12 {
13     echo "Nome =" . $x . ", Idade =" . $x_value;
14     echo "<br>";
15 }
16
17 ?>
18 </body>
Line 13, Column 35; Saved /var/www/html/Aula03.php (UTF-8)

Aula 03 - php - Mozilla Firefox
Aula 03 - php
localhost/Aula
Nome =Peter, Idade =35
Nome =Ben, Idade =37
Nome =Joe, Idade =43
```

Agenda

- PHP
 - Funções
 - Arrays
 - Sorting Arrays
 - Superglobals

Sorting Arrays

- Os elementos de uma matriz podem ser classificados em ordem alfabética ou numérica, descendente ou ascendente.
 - **sort ()** - ordenar matrizes em ordem crescente;
 - **rsort ()** - classificar matrizes em ordem decrescente;
 - **asort ()** - classifica as matrizes associativas em ordem crescente, de acordo com o valor;
 - **ksort ()** - classifica as matrizes associativas em ordem crescente, de acordo com a chave;
 - **arsort ()** - classifica as matrizes associativas em ordem decrescente, de acordo com o valor;
 - **krsort ()** - classifica as matrizes associativas em ordem decrescente, de acordo com a chave;

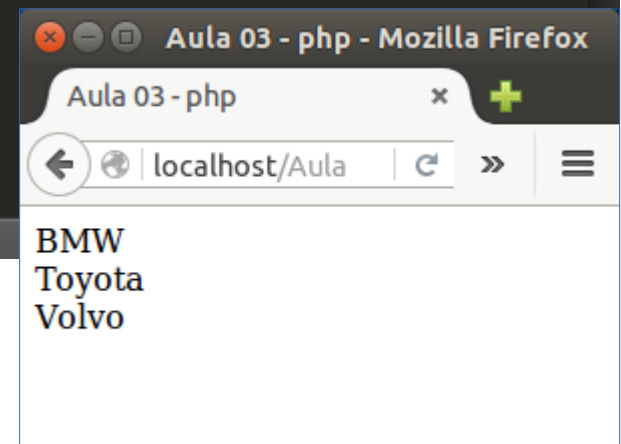
Sorting Arrays

Sort

```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)

Aula03.php x
1 <!DOCTYPE html>
2 <html>
3 <head>
4     <title>Aula 03 - php</title>
5 </head>
6 <body>
7 <?php
8     $carros = array("Volvo", "BMW", "Toyota");
9     sort($carros);
10
11     foreach ($carros as $valor) {
12         echo "$valor <br>";
13     }
14 ?>
15 </body>

Line 13, Column 6
```



Exercícios

Arrays Sortings

1. Ordene e exibe na tela os números abaixo:

```
$numeros = array(4, 6, 2, 22, 11);
```

2. Ordene em ordem decrescente e exibe:

```
$carro = array("Volvo", "BMW", "Toyota");
```

3. Ordene um Array associativa em ordem crescente, de acordo com o valor (nome):

```
$idade = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
```

4. Ordene as matrizes associativas do exercício anterior em ordem decrescente, de acordo com a chave (idade);

Agenda

- PHP
 - Funções
 - Arrays
 - Sorting Arrays
 - Superglobals

Superglobals

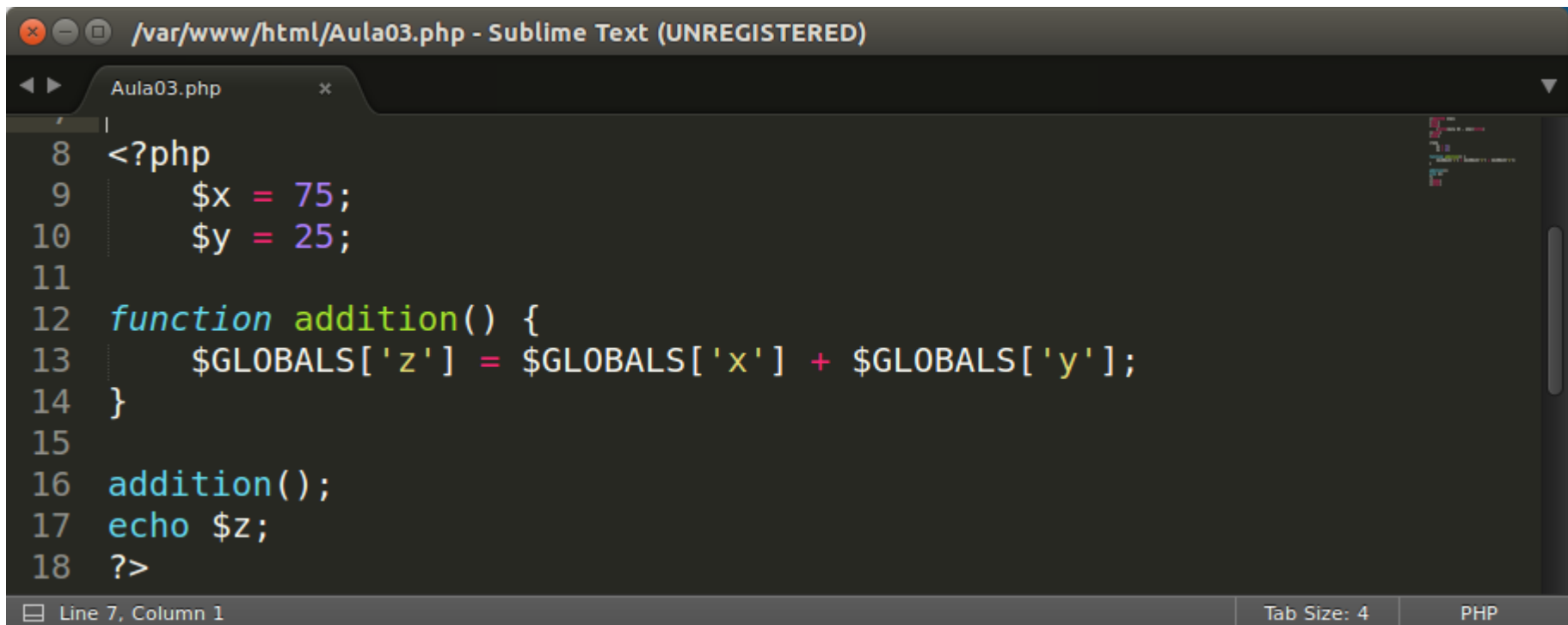
- Várias variáveis predefinidas em PHP são "superglobais":
 - Significa que elas sempre estão acessíveis, independentemente do alcance;
 - Pode acessá-las de qualquer função, classe ou arquivo sem ter que fazer nada de especial.

Superglobals

- As variáveis superglobal são:
 - `$GLOBALS`
 - `$_SERVER`
 - `$_REQUEST`
 - `$_POST`
 - `$_GET`
 - `$_FILES`
 - `$_ENV`
 - `$_COOKIE`
 - `$_SESSION`

\$GLOBALS

- Variável usada para acessar variáveis globais de qualquer lugar no script PHP (também dentro de funções ou métodos).

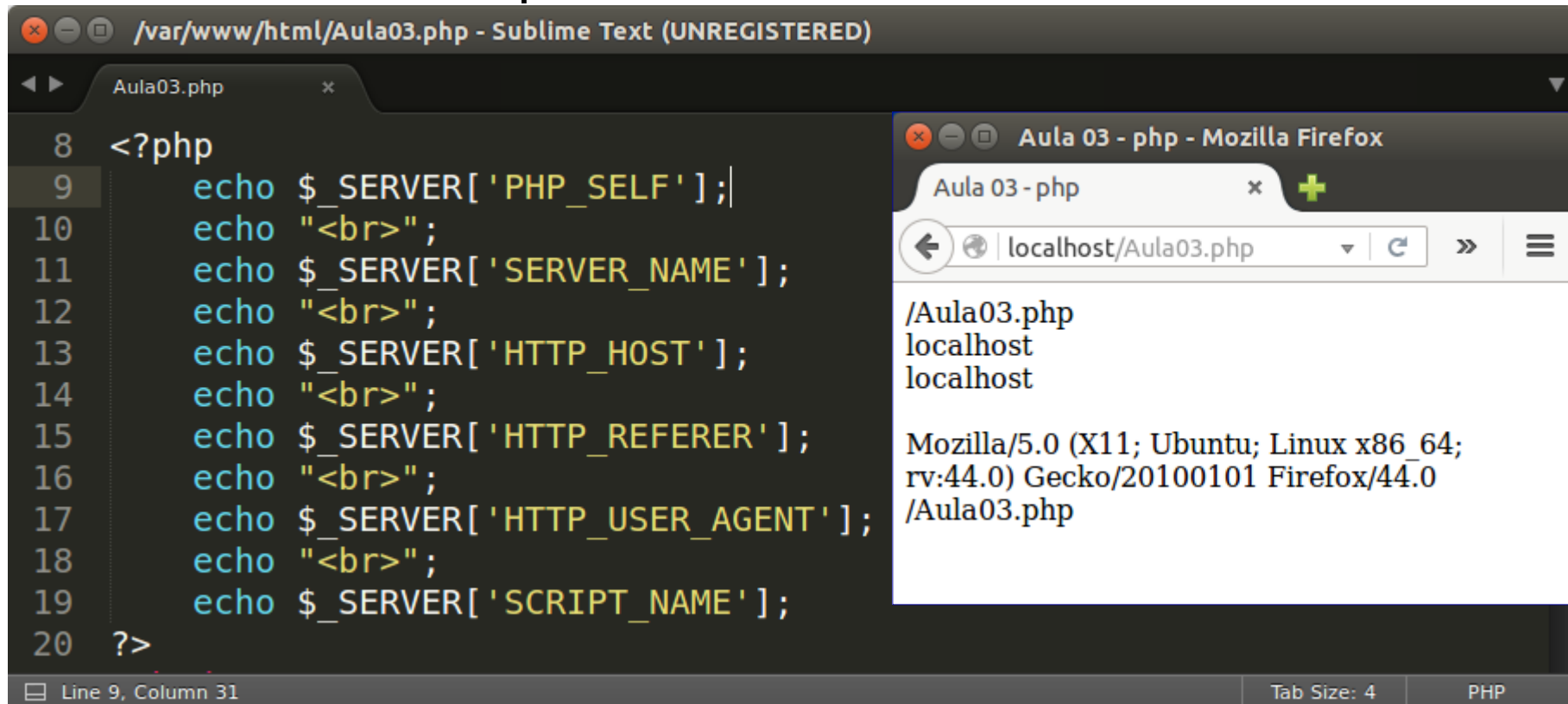


```
/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)
Aula03.php
8 <?php
9     $x = 75;
10    $y = 25;
11
12    function addition() {
13        $GLOBALS['z'] = $GLOBALS['x'] + $GLOBALS['y'];
14    }
15
16    addition();
17    echo $z;
18    ?>
```

Line 7, Column 1 Tab Size: 4 PHP

\$_SERVER

- \$_SERVER é uma super variável global do PHP que contém informações sobre cabeçalhos, caminhos e locais de script.



The image shows a Sublime Text editor window titled "/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)" and a Mozilla Firefox browser window titled "Aula 03 - php - Mozilla Firefox". The browser window displays the output of the PHP script, which is the output of the code shown in the Sublime Text editor.

```
8 <?php
9 echo $_SERVER['PHP_SELF'];|
10 echo "<br>";
11 echo $_SERVER['SERVER_NAME'];
12 echo "<br>";
13 echo $_SERVER['HTTP_HOST'];
14 echo "<br>";
15 echo $_SERVER['HTTP_REFERER'];
16 echo "<br>";
17 echo $_SERVER['HTTP_USER_AGENT'];
18 echo "<br>";
19 echo $_SERVER['SCRIPT_NAME'];
20 ?>
```

The browser window shows the output of the script:

```
/Aula03.php
localhost
localhost

Mozilla/5.0 (X11; Ubuntu; Linux x86_64;
rv:44.0) Gecko/20100101 Firefox/44.0
/Aula03.php
```

The status bar at the bottom of the Sublime Text window indicates "Line 9, Column 31", "Tab Size: 4", and "PHP".

`$_REQUEST`

- Usado para coletar dados depois de enviar um formulário HTML.
- Exemplo:
 - Um formulário com um campo de entrada e um botão de envio. Quando um usuário envia os dados clicando em "Enviar", os dados do formulário são enviados para o arquivo especificado no atributo “action” do `<form>`.
 - No próximo slide, apontamos para este arquivo para o processamento de dados do formulário.
 - Se você deseja usar outro arquivo PHP para processar os dados do formulário, substitua-o pelo nome do arquivo escolhido. Então, podemos usar a variável global `$_REQUEST` para coletar o valor do campo de entrada:

\$_REQUEST

```

/var/www/html/Aula03.php - Sublime Text (UNREGISTERED)
Aula03.php
6 <body>
7
8 <form method="post" action="<?php echo $_SERVER['PHP_SELF'];?>">
9     Name: <input type="text" name="fname">
10     <input type="submit">
11 </form>
12
13 <?php
14 if ($_SERVER["REQUEST_METHOD"] == "POST")
15 {
16     // Coleta o valor de entrada do campo
17     $nome = $_REQUEST['fname'];
18     if (empty($nome)) {
19         echo "Nome está vazio";
20     } else {
21         echo $nome;
22     }
23 }
24 ?>
25 </body>

```

Line 21, Column 17

Aula 03 - php - Mozilla Firefox

Aula 03 - php

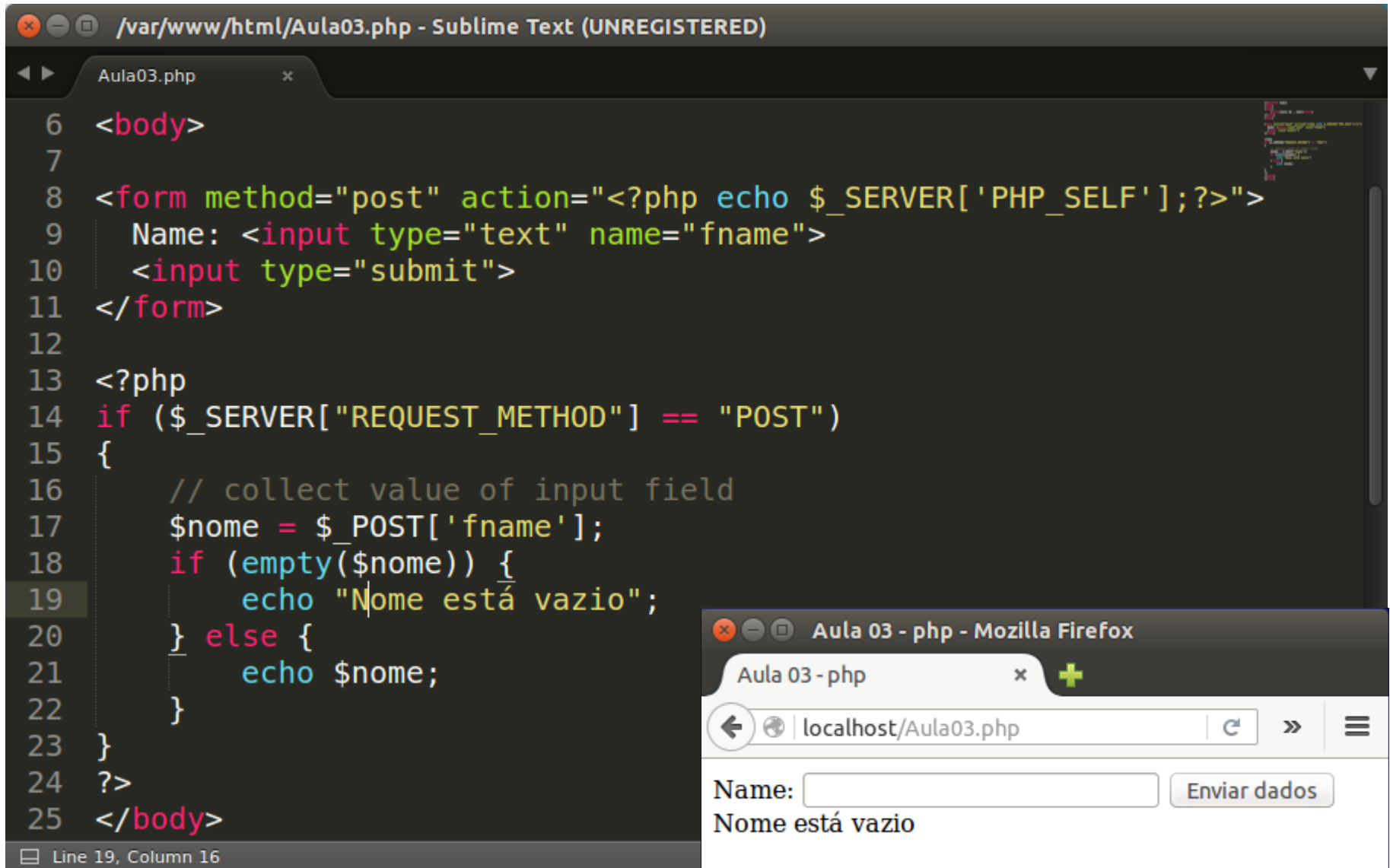
localhost/Aula03.php

Name:

Nome está vazio

Enviar dados

\$_POST



The image shows a development environment with a code editor and a web browser. The code editor, Sublime Text, displays a PHP file named Aula03.php. The code defines an HTML form with a text input field and a submit button. The form's action is a PHP script that checks if the request method is POST. If it is, it echoes the value of the 'fname' input field; otherwise, it echoes 'Nome está vazio'.

```
6 <body>
7
8 <form method="post" action="<?php echo $_SERVER['PHP_SELF'];?>">
9     Name: <input type="text" name="fname">
10     <input type="submit">
11 </form>
12
13 <?php
14 if ($_SERVER["REQUEST_METHOD"] == "POST")
15 {
16     // collect value of input field
17     $nome = $_POST['fname'];
18     if (empty($nome)) {
19         echo "Nome está vazio";
20     } else {
21         echo $nome;
22     }
23 }
24 ?>
25 </body>
```

The web browser, Mozilla Firefox, shows the rendered page at localhost/Aula03.php. It displays the form with the text 'Nome está vazio' below the input field, indicating that the form has been submitted and the input was empty.

Line 19, Column 16

Próxima aula

- MySQL Database
 - MySql Database
 - MySql Connect
 - MySql Create DB
 - MySql Create Table