

PHP

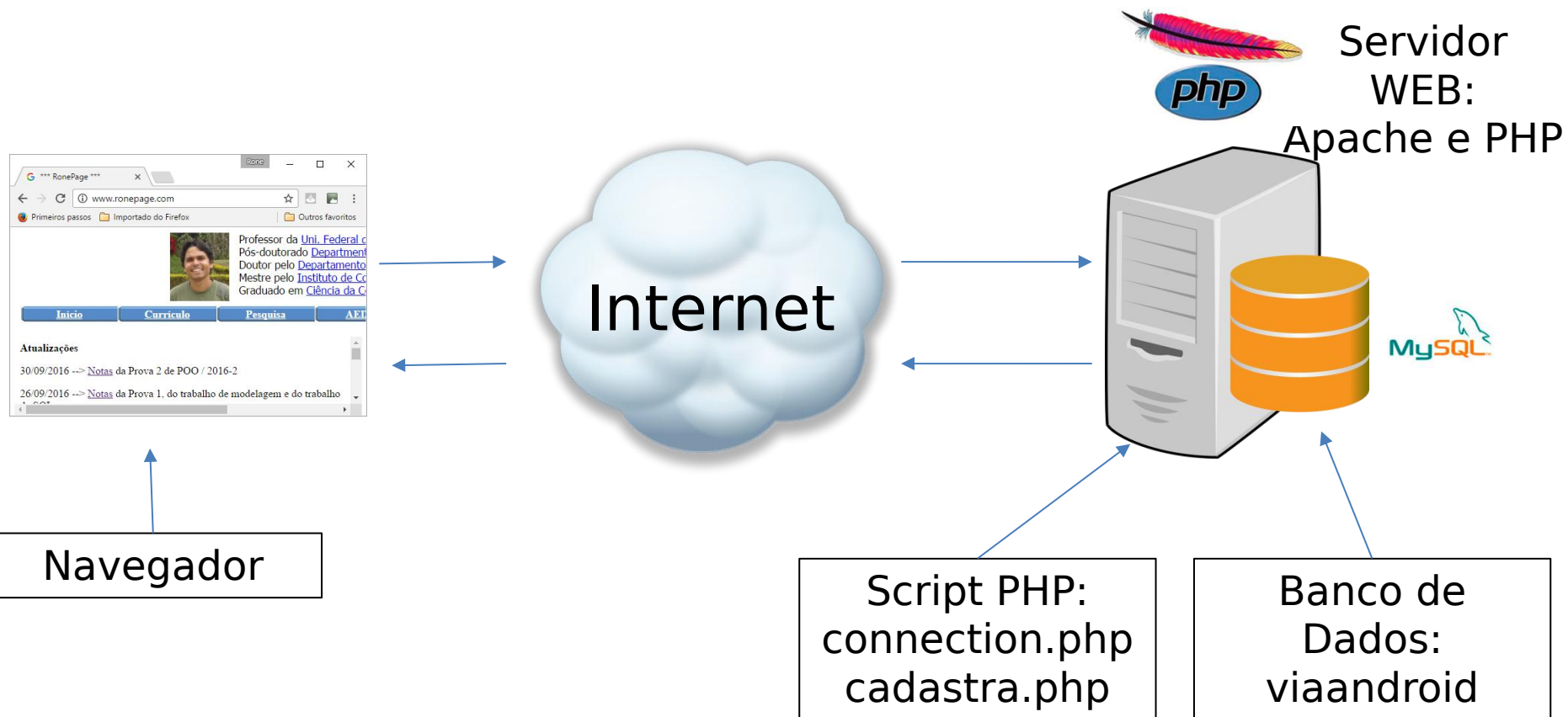
Prof. Rone Ilídio
UFSJ-DTECH-CAP

Aplicativos Necessários

- Servidor Web Apache
- Plugin PHP
- Bando de Dados MySQL
- Obs: o WAMP já vem com os três em um único pacote.

O que é PHP?

- Linguagem de scripts (não compila) para criação de páginas dinâmicas (que mudam o conteúdo de acordo com o contexto).

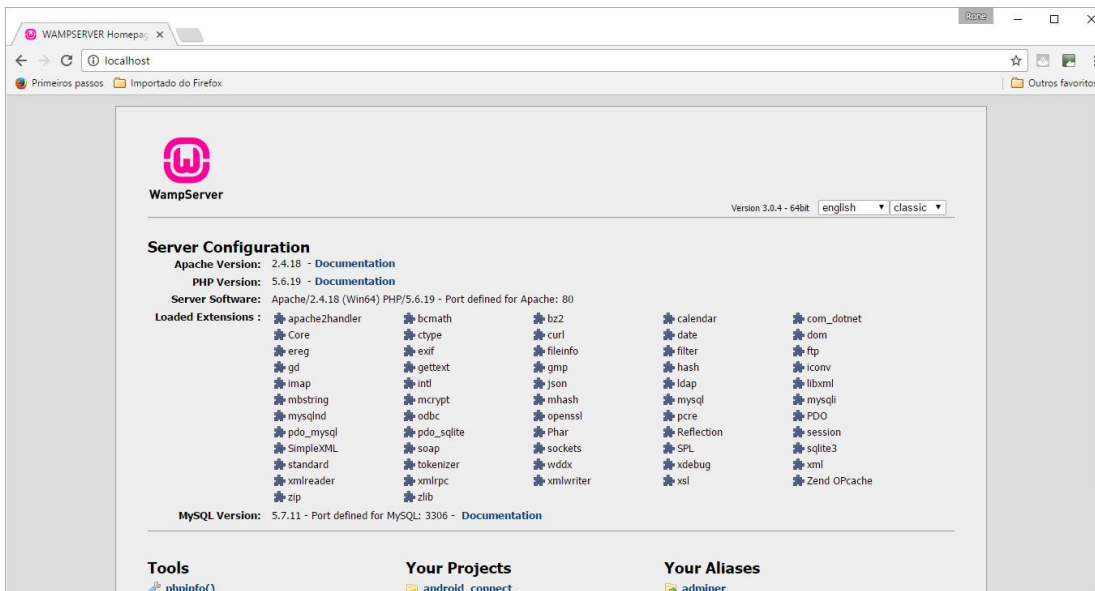


Estrutura

- O Apache cria uma pasta chamada `WWW`
 - Com o WAMP, a pasta normalmente é `c:\wamp\www`
- O conteúdo desta pasta é disponibilizado para acesso remoto
- Cada subpasta é um site diferente
- Exemplos:

Estrutura

- Digite no navegador:
 - <http://localhost>
 - <http://127.0.0.1>



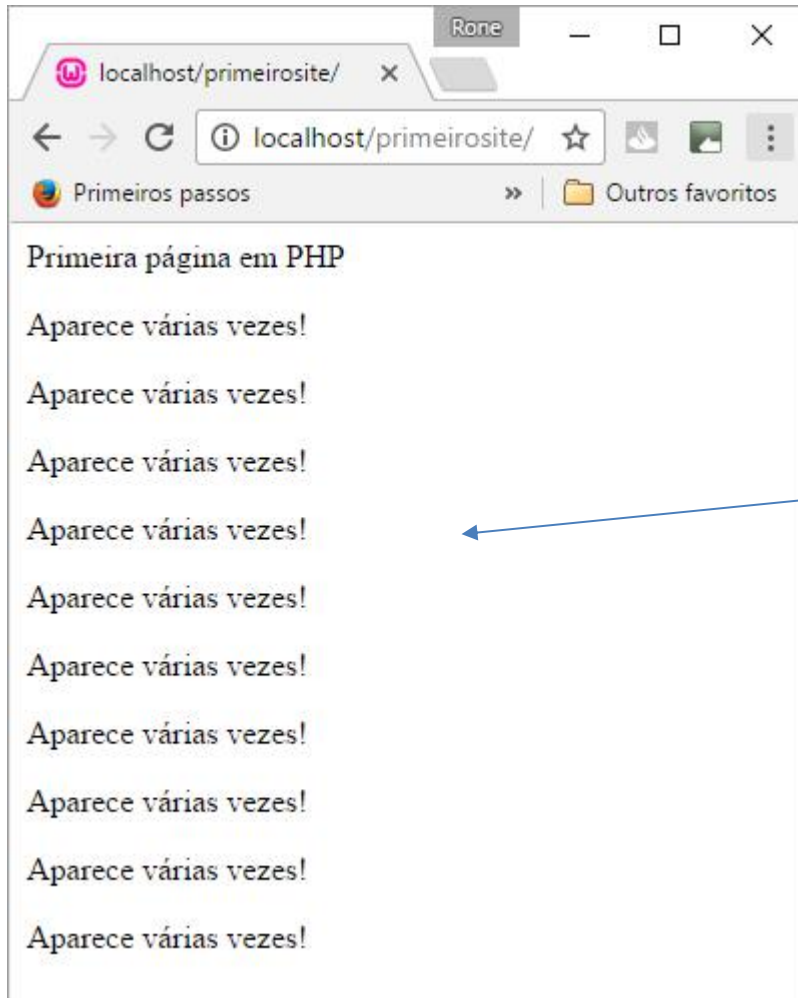
Aparecerá o
conteúdo do
arquivo
`/www/index.php`

Estrutura

- Crie a pasta /www/primeirosite
- Crie o arquivo index.php, dentro desta página, com o seguinte conteúdo:

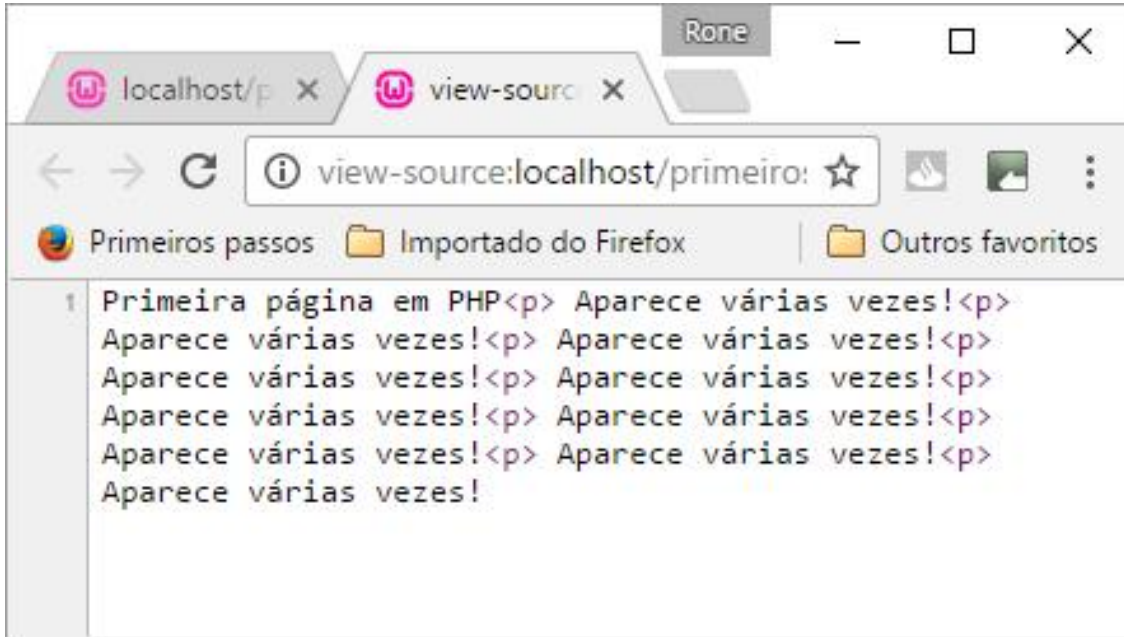
```
<?php
    for($i=0; $i<5; $i++){
        echo "<p> Repetindo ...";
    }
?>
```

Estrutura



Clique com o botão direito do mouse sobre a página e escolha “Exibir código fonte da página”

Estrutura



- O script PHP é executado e o servidor fornece como resultado somente um código HTML
- O código PHP fica protegido no servidor

TAG

- Tag: todo o código php deve ficar dentro da tag `<?php ... ?>`
- Existe a possibilidade de utilizar short-tag, que seria `<? ... ?>`, basta configura o `php.ini`
 - `short_open_tag = on`

TAG

- O código PHP pode ser inserido junto com o código HTML, exemplo:

```
<html>
  <head>
    <title>PHP Teste</title>
  </head>
  <body>
    <?php echo "<p>Olá Mundo</p>"; ?>
  </body>
</html>
```

Declaração de variáveis

- PHP não é uma linguagem fortemente tipada, ou seja, não é necessário especificar o tipo das variáveis
- O símbolo \$ é utilizado na frente de todas as variáveis
- Ex:

```
<?php
```

```
    $a = 10;
```

```
    $b = 20.45;
```

```
    $c = "Programação WEB";
```

```
    echo "Variáveis: <br> a = $a <br> b = $b <br> c = $c";
```

```
?>
```

Funções

- Uma função é definida pela palavra function
- Ex:

```
<?php
```

```
function myFunction($value){  
    $value = $value * $value;  
    return $value;  
}
```

```
echo "<p> Chamando a função --> " . myFunction(8) . " <-- o  
ponto é utilizado para concatenação.";
```

```
?>
```

Funções

- Atenção no escopo das variáveis, veja os código

```
<?php
function myTest() {
    $x = 0;
    echo "<p>".$x;
    $x++;
}
myTest();
myTest();
myTest();
?>
```

```
<?php
function myTest() {
    static $x = 0;
    echo "<p>".$x;
    $x++;
}
myTest();
myTest();
myTest();
?>
```

Funções

- As variáveis globais são armazenadas no vetor `$GLOBALS['nome_da_variavel']`

```
<?php
    $x=0;
    function myTest() {
        echo "<p>".$GLOBALS['x'];
        $GLOBALS['x']++;
    }
    myTest();
    myTest();
    myTest();
?>
```

Comandos Básico do PHP

Comando for

```
<html>
  <head> <title>PHP teste for</title></head>
<body>
  <?php
    echo "<table border=1>";
    for($i=0 ; $i<10; $i++){
      echo "<tr>";
      echo "      <td> conteúdo </td>";
      echo "      <td> conteúdo </td>";
      echo "      <td> conteúdo </td>";
      echo "</tr>";
    }
    echo "</table>";
  ?>
</body>
</html>
```

Nesse exemplo, o for faz a repetição da criação de cada uma das linhas da tabela

Comando while

```
<?php
    $x = 1;
    while($x <= 5) {
        echo "O número é:
    $x <br>";
        $x++;
    }
?>
```

```
<?php
    $x = 1;
    do {
        echo "O número é:
    $x <br>";
        $x++;
    } while($x <= 5) ;
?>
```

Comando if/else

```
<?php
    $t = 23;
    if($t%2 == 0){
        echo "<p> $t é par.";
    }
    else
    {
        echo "<p> $t é impar";
    }
?>
```

Array

- Criado pela função array()

```
<?php
$cars = array("Fusca", "Brasilha", "Fiat 147");
for($x = 0; $x < count($cars); $x++) {
    echo $cars[$x];
    echo "<br>";
}
?>
```

Array

- Outro exemplo:

```
<?php
$cars = array();
for($y = 0; $y < 3; $y++) {
    $cars[$y] = $y;
}
for($x = 0; $x < count($cars); $x++) {
    echo $cars[$x];
    echo "<br>";
}
?>
```

Array associativo

- Array cujos índices são valores informados, não predefinidos.

```
<?php
```

```
$age = array("Pedro"=>"35", "Ze"=>"37", "Ana"=>"43");  
echo "Pedro tem " . $age['Pedro'] . " anos.";
```

```
?>
```

Ou

```
<?php
```

```
$age['Pedro'] = "35";  
$age['Ze'] = "37";  
$age['Ana'] = "43";  
echo "Pedro tem " . $age['Pedro'] . " anos.";
```

```
?>
```

Ordenando Array

- Função sort() em ordem crescente e rsort() decrescente

```
<?php
    $cars = array("Volvo", "BMW", "Toyota");
    sort($cars);

    for($x = 0; $x < count($cars); $x++) {
        echo $cars[$x];
        echo "<br>";
    }
?>
```

Data

- A função `data(...)` retorna uma string com a data atual
- Seu parâmetro informa o formato da data, ex:
 - `$date = date('d-m-Y H:i');` 20-10-2016
09:42
 - d=dia
 - m-mês
 - Y=ano
 - H=hora
 - i=minuto

Data

- Exemplos
 - `$date = date('d')` retorna somente o dia
 - `$tomorrow = date('d/m/Y', strtotime("+10 days"));
echo "
 $tomorrow";` retorna 10 dias no futuro
- Existem vários parâmetros para configura a data, veja todos os exemplos em:

http://php.net/manual/pt_BR/function.date.php

Enviado Dados de um Formulário

- O PHP possui 2 métodos de envio de dados:
 - GET: dados vão na URL
 - POST: dados vão dentro do cabeçalho PHP, ou seja, o usuário comum não tem acesso*

*Esses dados podem ser lidos por sniffers

Método GET

- Crie o seguinte formulário (de acordo com o código no próximo slide)

Cálculo do IMC

Peso

Altura

Calcular IMC

Método GET

```
<html>
<meta charset="utf-8">
<head><title>Cálculo do IMC</title></head>
<body>
  <h3> Cálculo do IMC
  <form method="GET" action="calcularimc.php">
    <p> Peso <br>
    <input type="text" name="peso">
    <p> Altura <br>
    <input type="text" name="altura" > <br> <br>
    <input type="submit" value="Calcular IMC">
  </form>
</body>
</html>
```

Método GET

- Crie um script PHP chamado calcularimc.php
- Utilize o seguinte código

```
<?php
    $peso = $_GET['peso'];
    $altura = $_GET['altura'];
    $imc = $peso / pow($altura,2);
    echo "<p> IMC = $imc";
    echo "<p> <a href='formimc.php'>Voltar</a>";
?>
```

Método GET

- Atributos da tag form:
 - action: define qual script php tratará as informações no formulário
 - method: define se o envio utilizará GET ou POST
- Após clicar o botão do formulário, verifique a url que aparecerá no navegador

http://localhost/aulaphp/calcularimc.php?peso=74&altura=1.72

Delimitador
URL/dados

Nomes dos inputs

Nomes dos inputs

Método GET

- No script PHP temos:
 - O vetor `$_GET['...']` é um chamado de vetor global
 - Cada componente do formulário será uma de suas posições
 - A função `pow(a,b)` calcula o valor de a^b

Método POST

- No formulário, substitua GET por POST
- No script PHP, utilize o vetor `$_POST`
- Verifique que os dados do formulário não estão na URL

Tudo em um único código

- No exemplo a seguir, o formulário e o cálculo do IMC está em um mesmo código
- Esse tipo de programação pode levar a erros
- A função `isset(...)` verifica se uma variável foi criada.

Tudo em um único código

```
<html>

<meta charset="utf-8">

<head><title>Cálculo do IMC</title></head>

<body>

    <?php

        if(isset($_POST['peso']) && isset($_POST['altura']) && $_POST['peso']!=0 && $_POST['altura']!=0){

            $peso = $_POST['peso'];

            $altura = $_POST['altura'];

            $imc = $peso / pow($altura,2);

            echo "<p> IMC = $imc";

        }

    ?>

    <h3> Cálculo do IMC

    <form method="POST" action="formimc.php">

        <p> Peso <br>

        <input type="text" name="peso">

        <p> Altura <br>

        <input type="text" name="altura" > <br> <br>

        <input type="submit" value="Calcular IMC">

    </form>

</body>

</html>
```