INSTITUTO FEDERAL
São Paulo

PHP
Aula 3

Prof. DR. Eder Pansani
epansani@gmail.com / epansani@ifsp.edu.br

Conteúdo da aula

INSTITUTO FEDERAL
São Paulo

Nesta aula iremos aprender sobre variáveis, tipos de dados, constantes e o tratamento de requisições em PHP.

INSTITUTO FEDERAL
São Paulo

Variáveis

Variáveis

INSTITUTO FEDERAL
São Paulo

Variáveis são identificadores utilizados para representar valores mutáveis que só existem durante a execução do programa.

Variáveis

INSTITUTO FEDERAL
São Paulo

Elas são armazenadas na memória RAM e seu conteúdo é destruído após a execução do programa.

De forma resumida, variáveis são "contêineres" para armazenar informações.

Variáveis

INSTITUTO FEDERAL
São Paulo

No PHP, uma variável começa com o símbolo “$”, seguido pelo nome da variável

```php
<?php
$txt = "Hello world!";
$x = 5;
$y = 10.5;
?>
```

Variáveis

INSTITUTO FEDERAL
São Paulo

Ao contrário de outras linguagens, no PHP **não é necessário fazer a declaração prévia das variáveis** além de **não ser necessário informar um tipo** para as mesmas.

Variáveis

INSTITUTO FEDERAL
São Paulo

Na linguagem PHP as variáveis são criadas no instante em que um valor é atribuído a elas pela primeira vez.

Além disso a linguagem identifica o tipo da variável de acordo com seu conteúdo de forma automática.

Variáveis

INSTITUTO FEDERAL
São Paulo

Regras para criação de variáveis:

- Uma variável deve começar com um “$” seguido pelo nome da variável
- Um nome de variável deve começar com uma letra ou o caractere *underline*.

Variáveis

INSTITUTO FEDERAL
São Paulo

- Um nome de variável não pode começar com um número
- Um nome de variável pode conter apenas caracteres alfanuméricos e sublinhados (*underliune*) (Az, 0-9 e _)
- Os nomes das variáveis diferenciam maiúsculas de minúsculas (*case sensitive*)

Variáveis

INSTITUTO FEDERAL
São Paulo

- Um nome de variável não pode começar com um número
- Um nome de variável pode conter apenas caracteres alfanuméricos e sublinhados (*underline*) (Az, 0-9 e _)
- Os nomes das variáveis diferenciam maiúsculas de minúsculas (*case sensitive*)

Variáveis

INSTITUTO FEDERAL
São Paulo

PHP é uma linguagem livremente tipada.

Isto é, não é necessário informar qual o tipo de dado de uma variável antes de sua atribuição.

Variáveis

INSTITUTO FEDERAL
São Paulo

O PHP associa automaticamente um tipo de dado à variável, dependendo de seu valor. Como os tipos de dados não são definidos em sentido estrito, você pode fazer coisas como adicionar uma ***string*** a um ***inteiro*** sem causar um erro.

Variáveis

INSTITUTO FEDERAL
São Paulo

No PHP 7, foram adicionadas declarações de tipo. Isso fornece uma opção para especificar o tipo de dados esperado ao declarar uma função e, ao habilitar o requisito estrito, ele lançará um "Erro Fatal" em uma incompatibilidade de tipo.

```php
<?php

$x = 12;
var_dump($x);
$x = 1.234;
var_dump($x);
$x = "hello";
var_dump($x);
$x = 'hellop';
var_dump($x);
$x = true;
var_dump($x);
$x = [1, '14', null];
var_dump($x);

?>
```

```
int(12)
float(1.234)
string(5) "hello"
string(6) "hellop"
bool(true)
array(3) {
  [0]=>
  int(1)
  [1]=>
  string(2) "14"
  [2]=>
  NULL
}
```

INSTITUTO FEDERAL
São Paulo

Tipos de dados

Tipos de dados

INSTITUTO FEDERAL
São Paulo

As variáveis podem armazenar dados de diferentes tipos para finalidades diferentes.

O PHP suporta os tipos de dados: ***string, int, float boolean, Array, Object, NULL e Resource***.

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Uma ***string*** é uma sequência de caracteres alfanuméricos, como "Hello world!".

Para declará-la, podemos utilizar aspas simples (‘’) ou aspas duplas (“”).

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Exemplo de uma ***string***

```php
<?php
$x = "Hello world!";
$y = 'Hello world!';

echo $x;
echo "<br>";
echo $y;
?>
```

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Importante, há uma diferença significativa na declaração de uma *string* com aspas duplas ou aspas simples. Todo conteúdo dentro de aspas duplas é avaliado pelo PHP. Assim, se a *string* contém uma variável , esta será traduzida para seu valor.

Tipos de dados

INSTITUTO FEDERAL
São Paulo

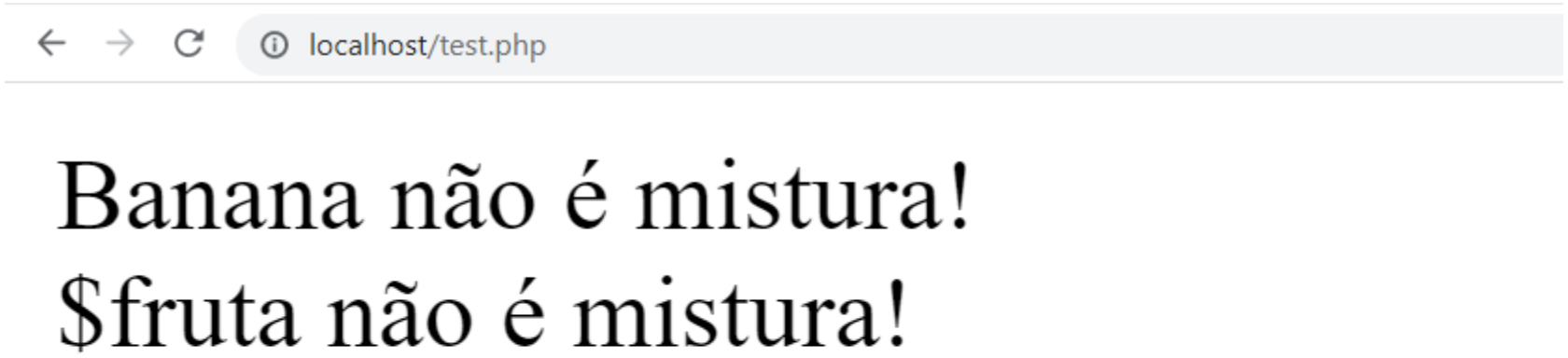
Exemplo,

```php
<?php

$fruta = "Banana";

echo "$fruta não é mistura!";
echo '$fruta não é mistura!';

?>
```

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Exemplo,



Tipos de dados

INSTITUTO FEDERAL
São Paulo

Um tipo de dados **inteiro** é um número não decimal entre -2.147.483.648 e 2.147.483.647.

Números inteiros devem ter ao menos um dígito, não podem ter ponto decimal, podem ser positivos ou negativos e declarados em outras bases (hexadecimal, octal, binária)

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Um ***Array*** é um tipo de dados que armazena vários valores em uma única variável.

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

Tipos de dados

Arrays também pode ser alterados dinamicamente com a adição ou remoção itens.

```
array(3) {
  [0]=>
  string(5) "Volvo"
  [1]=>
  string(3) "BMW"
  [2]=>
  string(6) "Toyota"
}
```

Tipos de dados

INSTITUTO FEDERAL
São Paulo

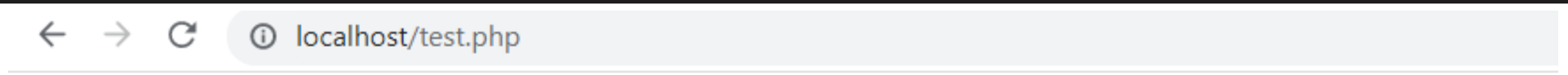
Importante, para se exibir o conteúdo de um *Array* em PHP é necessário especificar seu índice.

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Observe,

```php
<?php

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

```



Warning: Array to string conversion in **C:\xampp\htdocs\test.php** on line **5**
Array

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Observe,

```php
<?php

$cars = array("Volvo","BMW","Toyota");
echo $cars[0];

```

Volvo

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Caso seja necessário mostrar todo o conteúdo de um *Array* (para debug por exemplo) pode usar a função *print_r*

print_r — Imprime informação sobre uma variável de forma legível

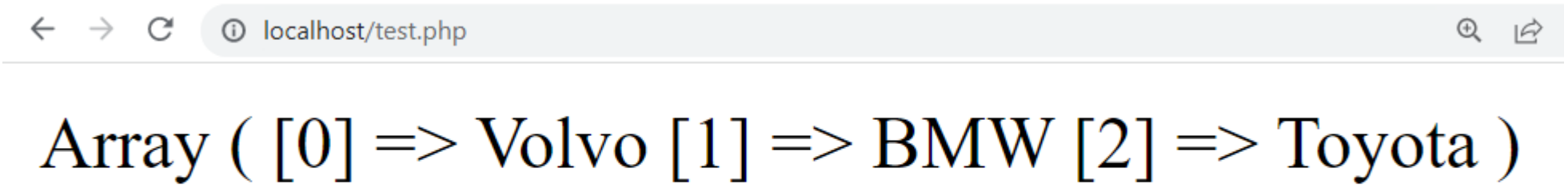
https://www.php.net/manual/pt_BR/function.print-r.php

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Observ

```php
<?php

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

```



Tipos de dados

INSTITUTO FEDERAL
São Paulo

Um tipo **objeto** é uma entidade com um determinado comportamento definido por seus métodos (ações) e propriedades (dados).

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Classes e objetos são os dois principais aspectos da programação orientada a objetos.

Uma classe é um modelo para objetos e um objeto é uma instância de uma classe.

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Quando os objetos individuais são criados, eles herdam todas as propriedades e comportamentos da classe, mas cada objeto terá valores diferentes para as propriedades.

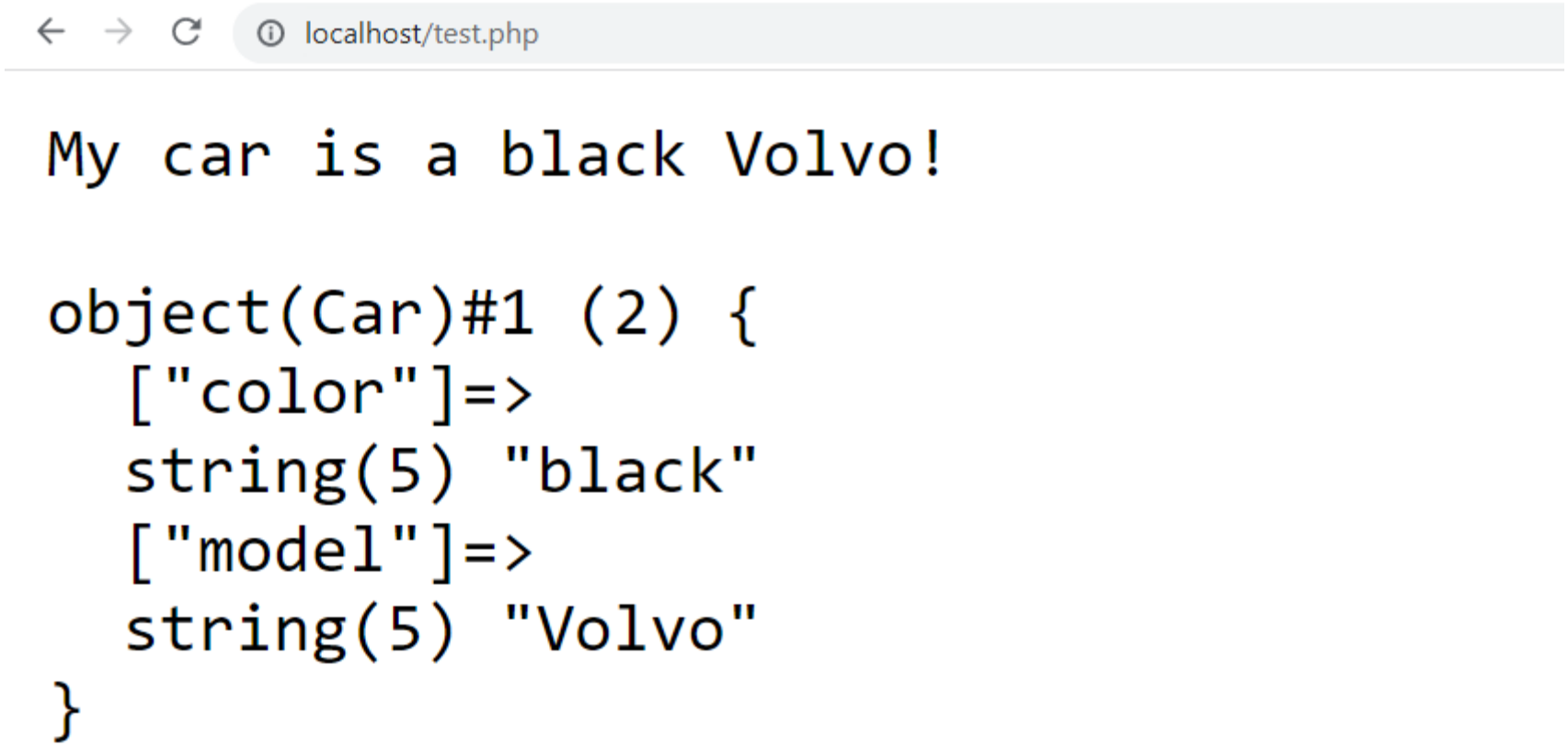
T

```php
<?php
class Car
{
    public $color;
    public $model;
    public function __construct($color, $model)
    {
        $this->color = $color;
        $this->model = $model;
    }
    public function message()
    {
        return "My car is a " . $this->color . " " . $this->model . "!";
    }
}
$myCar = new Car("black", "Volvo");
echo $myCar->message();

var_dump($myCar);

```

INSTITUTO FEDERAL
São Paulo

Tipos de dados

INSTITUTO FEDERAL
São Paulo



Tipos de dados

INSTITUTO FEDERAL
São Paulo

O tipo de **recurso** especial não é um tipo de dados real. Ele é o armazenamento de uma referência a funções e recursos externos ao PHP.

Um exemplo comum de uso do tipo de dados de recurso é uma chamada de banco de dados.

Tipos de dados

INSTITUTO FEDERAL
São Paulo

As funções ***mysqli_connect()*** e ***pg_connect()***, por exemplo, ao conectarem-se ao banco de dados, retornam uma variável de referência do tipo **recurso**.

Tipos de dados

INSTITUTO FEDERAL
São Paulo

Mais informações:

https://www.w3schools.com/php/php_datatypes.asp

INSTITUTO FEDERAL
São Paulo

Constantes

Constantes

INSTITUTO FEDERAL
São Paulo

Constantes são como **variáveis**, exceto que, uma vez definidas, elas não podem ser alteradas ou indefinidas.

Constantes

Sua função é armazenar valores que não podem/devem sofrer modificações durante a execução do programa. Assim como as variáveis elas possuem identificadores porém elas não tem o símbolo do “$”.

Constantes

INSTITUTO FEDERAL
São Paulo

As regras de **nomenclatura** de constantes seguem as mesmas regras das variáveis (com exceção do sinal cifrão) e por **convenção** utilizam **letras maiúsculas**.

Constantes

INSTITUTO FEDERAL
São Paulo

Para criar uma constante, usamos a função define()

```php
<?php
define("GREETING", "Welcome to W3Schools.com!");
echo GREETING;
?>
```

Constantes

INSTITUTO FEDERAL
São Paulo

Existem várias constantes pré-definidas no núcleo do PHP, elas armazenam a versão atual do PHP instalado, o número do release, sistema operacional, o símbolo para o “fim de linha” para a plataforma entre outras.

Constantes

INSTITUTO FEDERAL
São Paulo

Constantes Pré-definidas:

https://www.php.net/manual/pt_BR/reserved.constants.php

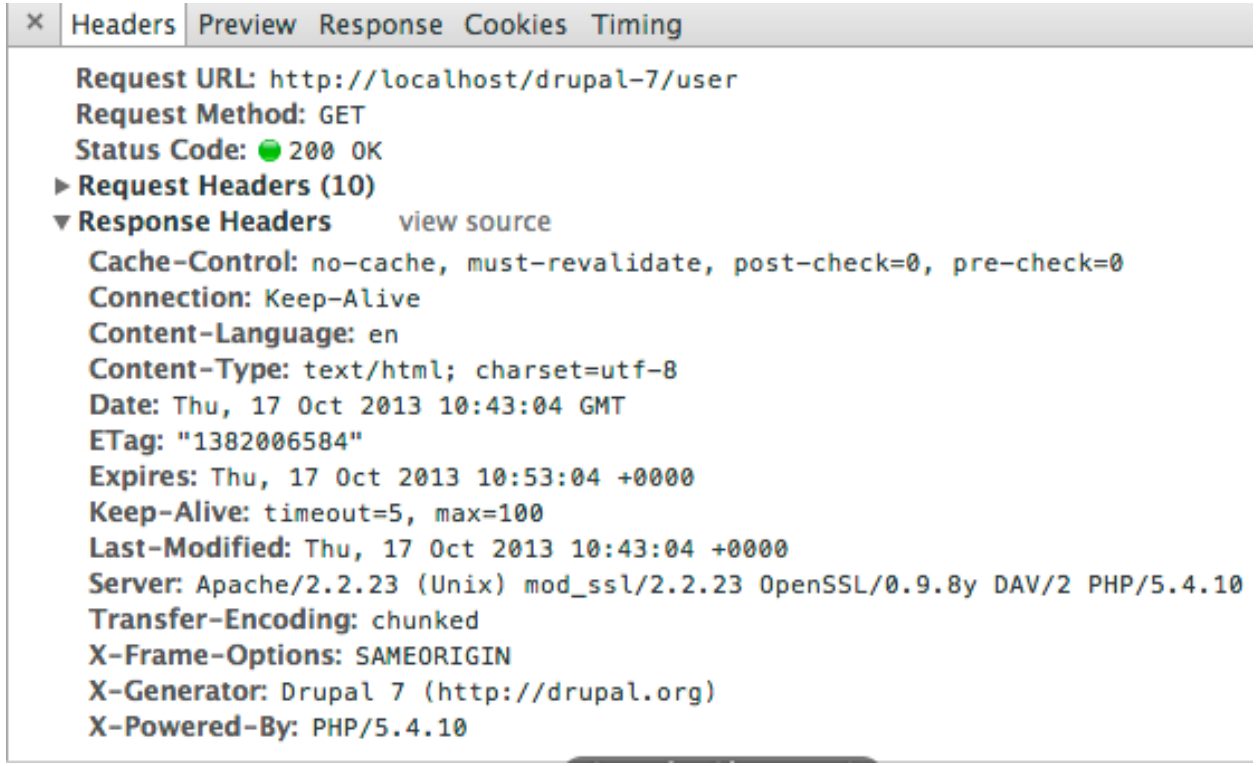
INSTITUTO FEDERAL
São Paulo

Tratamento de requisições

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

O **protocolo HTTP** é baseado em *streams* de texto (ASCII).

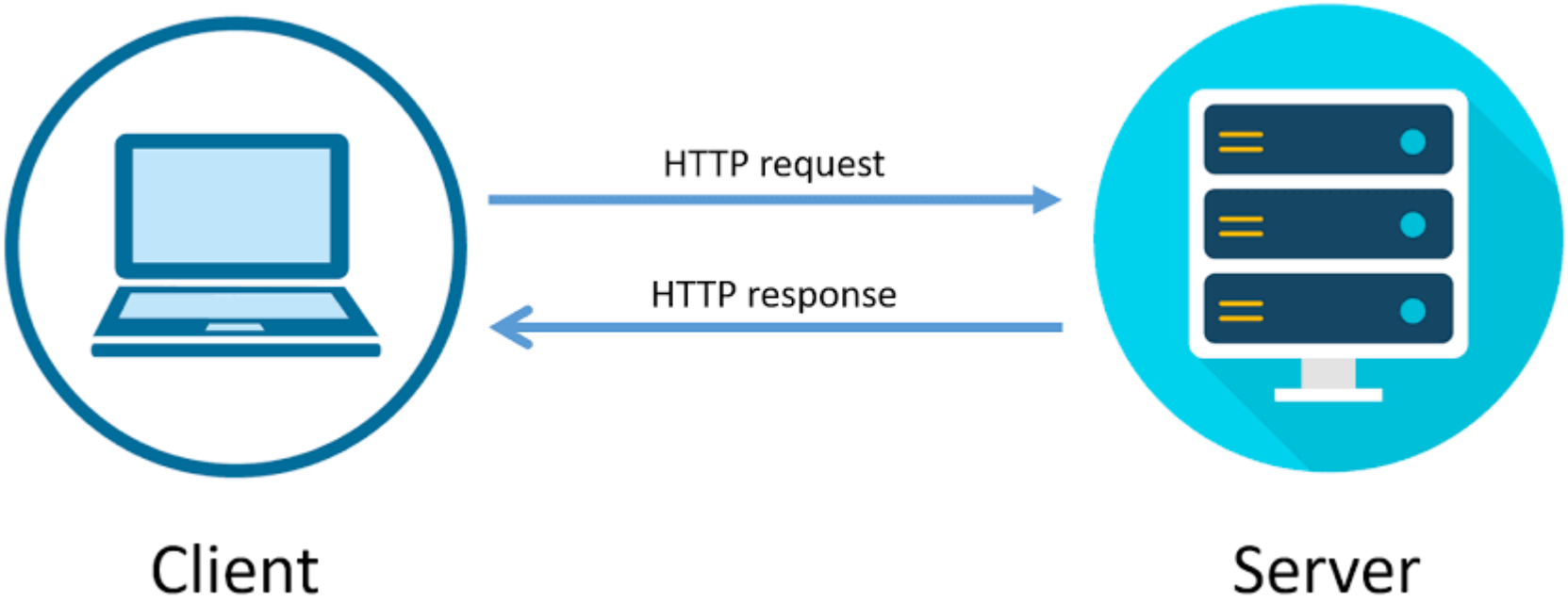


Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

De forma sucinta seu funcionamento é baseado em um cliente que abre um *socket* para falar com o servidor e, nesse *socket*, envia requisições (***requests***), as quais o servidor responderá com respostas (***responses***).

Tratamento de requisições



Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

As requisições são compostas de:

- Informações da requisição (*request line*)
- Cabeçalhos (*headers*)
- Corpo (*body*)

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

As informações da requisição (*request-line*) contém três elementos:

- o método HTTP (verbo)
- o alvo da requisição (geralmente uma URL)
- a versão do protocolo (HTTP) que está sendo utilizada.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Um **método HTTP**, um verbo (como GET, PUT ou POST) ou um nome (como HEAD ou OPTIONS), que descrevem a ação a ser executada. Por exemplo, GET indica que um recurso deve ser obtido ou POST significa que dados são inseridos no servidor.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

No **cabeçalho** podem ser informadas várias coisas, como a codificação, o tipo do conteúdo (texto plano, html, json, etc), o tamanho (em bytes) além de dados sobre cache, idioma, data de envio e muitas mais.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Já no **corpo da requisição** é onde ficam os dados enviados no formato especificado pelo cabeçalho.

Nem todas as requisições tem um corpo, as que pegam recursos, como GET, HEAD, DELETE, ou OPTIONS, usualmente não precisam de um.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

A linha inicial de uma **resposta HTTP,** chamada de linha de status, contém além da versão do protocolo e de um texto de status um **código de status**.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Uma linha de status típica se parece com:

```
HTTP/1.1 404 Not Found.
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Um **código de status** indica o sucesso ou falha da requisição.

HTTP STATUS CODES

2xx Success

200	Success / OK

3xx Redirection

301	Permanent Redirect
302	Temporary Redirect
304	Not Modified

4xx Client Error

401	Unauthorized Error
403	Forbidden
404	Not Found
405	Method Not Allowed

5xx Server Error

501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout

INFIDIGIT

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Lista de códigos de estado HTTP

https://pt.wikipedia.org/wiki/Lista_de_c%C3%B3digos_de_estado_HTTP

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Conteúdo complementar:

Mensagens HTTP

https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Messages

https://www.treinaweb.com.br/blog/o-que-e-http-request-get-post-response-200-404

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

No contexto da disciplina vamos trabalhar neste momento com os dois principais métodos de envio e recepção, POST e GET.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Lidaremos com requisições devido a natureza da forma como são implementados os programas em PHP, onde temos uma página onde são inseridas as informações (formulário) e uma outra que recebe esta informação (destino).

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Sendo assim vamos criar duas páginas:

- Um formulário
- Uma página destino (extensão PHP obrigatóriamente)

Tratamento de requisições

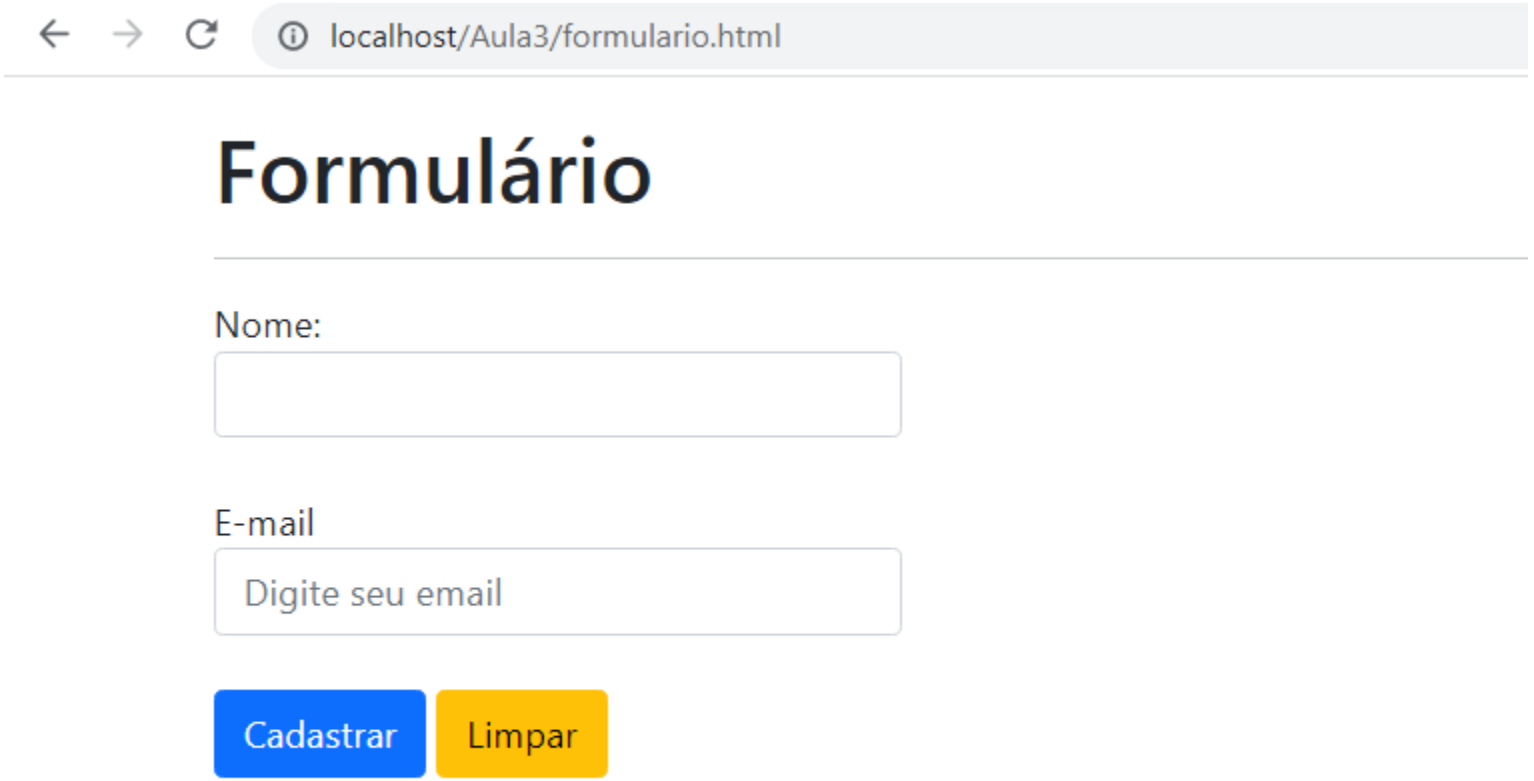
INSTITUTO FEDERAL
São Paulo

Na página formulario.html (ou formulario.php) crie um **formulário com dois campos, nome e e-mail** por exemplo.

Não se esqueça dos botões de enviar em limpar.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo



```
12 <body>
13     <h1>Formulário</h1>
14     <hr>
15
16     <form action="destino-sb.php" method="POST">
17         <label for="nome">Nome:</label>
18         <input type="text" name="nome" id="nome" required>
19         <br><br>
20
21         <label for="email">E-mail</label>
22         <input type="email" name="email" id="email"
23         placeholder="Digite seu email" required>
24         <br><br>
25
26         <button type="submit">Cadastrar</button>
27         <button type="reset">Limpar</button>
28     </form>
29
30 </body>
```

Tratamento de requisições

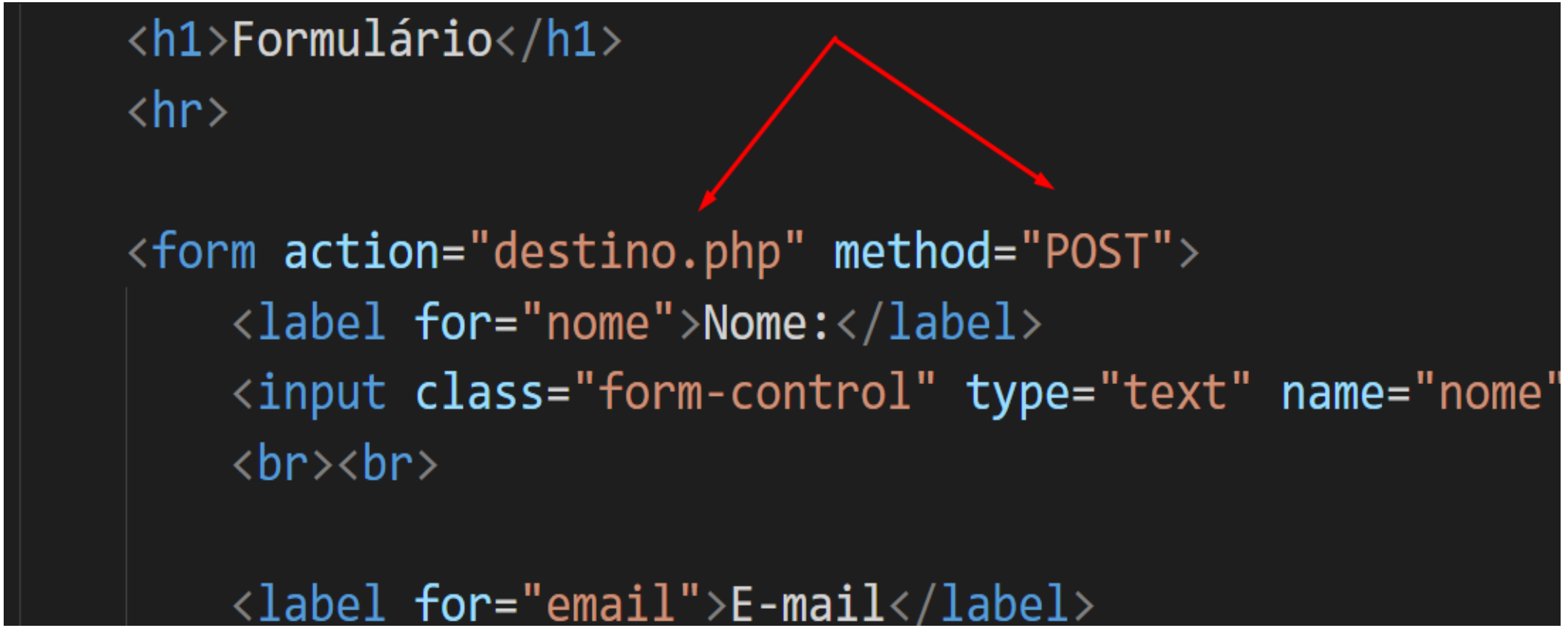
INSTITUTO FEDERAL
São Paulo

Atenção!

Agora (mais do que nunca) é preciso ter uma atenção especial aos **atributos *action* e *method*** do formulário (<form>).

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo



```
<h1>Formulário</h1>
<hr>

<form action="destino.php" method="POST">
    <label for="nome">Nome:</label>
    <input class="form-control" type="text" name="nome"
    <br><br>

    <label for="email">E-mail</label>
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

O atributo ***action*** serve para definir qual a página de destino deste formulário, ou seja, quando o formulário for submetido qual a página receberá a requisição com os dados que foram informados pelo usuário.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Ao passo que o atributo ***method*** serve para definir qual o método de envio da requisição.

Como foi dito utilizaremos neste momento os métodos POST e GET.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método GET

É o método padrão. Quando não for especificado nenhum valor no atributo *method* os dados serão enviados usando GET.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método GET

Quando o método GET é usado os dados do formulário são enviados de forma visível no campo de endereço da página.

```
/action_page.php?firstname=Mickey&lastname=Mouse
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método GET

O método GET dá origem a uma matriz de variáveis ($_GET) passadas para por meio dos parâmetros de URL.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método GET

Neste método o navegador pega as informações do formulário e as coloca junto com a URL de destino, separando o endereço da URL dos dados com um ponto de interrogação.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método GET

As informações enviadas de um formulário com o método GET são visíveis para todos (todos os nomes e valores das variáveis são exibidos na URL).

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método GET

Sendo assim ele deve ser usado apenas para enviar dados não confidenciais e ele **NUNCA** **deve ser usado para enviar senhas** ou outras informações confidenciais.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método GET

GET também tem **limites na quantidade de informações** a serem enviadas. A limitação é de **cerca de 2.000 caracteres**.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método GET

No entanto, como as variáveis são exibidas na URL, é possível **marcar a página como favorito**. Isso pode ser útil em alguns casos.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método POST

O método POST envia os dados colocando-os no **corpo da mensagem**. Ele deixa a URL separada dos dados que serão enviados.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método POST

As informações enviadas com o método POST são **invisíveis** para outros (todos os nomes/valores são incorporados no corpo da solicitação HTTP) e **não têm limites na quantidade de informações** a serem enviadas.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Método POST

Além disso, o POST suporta funcionalidades avançadas, como suporte para entrada binária de várias partes durante o *upload* de arquivos para o servidor.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Voltando ao exemplo de aula...

O próximo passo é criar uma página de destino (**destino.php**). Observe que esta página **deve ter a extensão PHP**, caso contrário ela não funcionará.

Tratamento de requisições

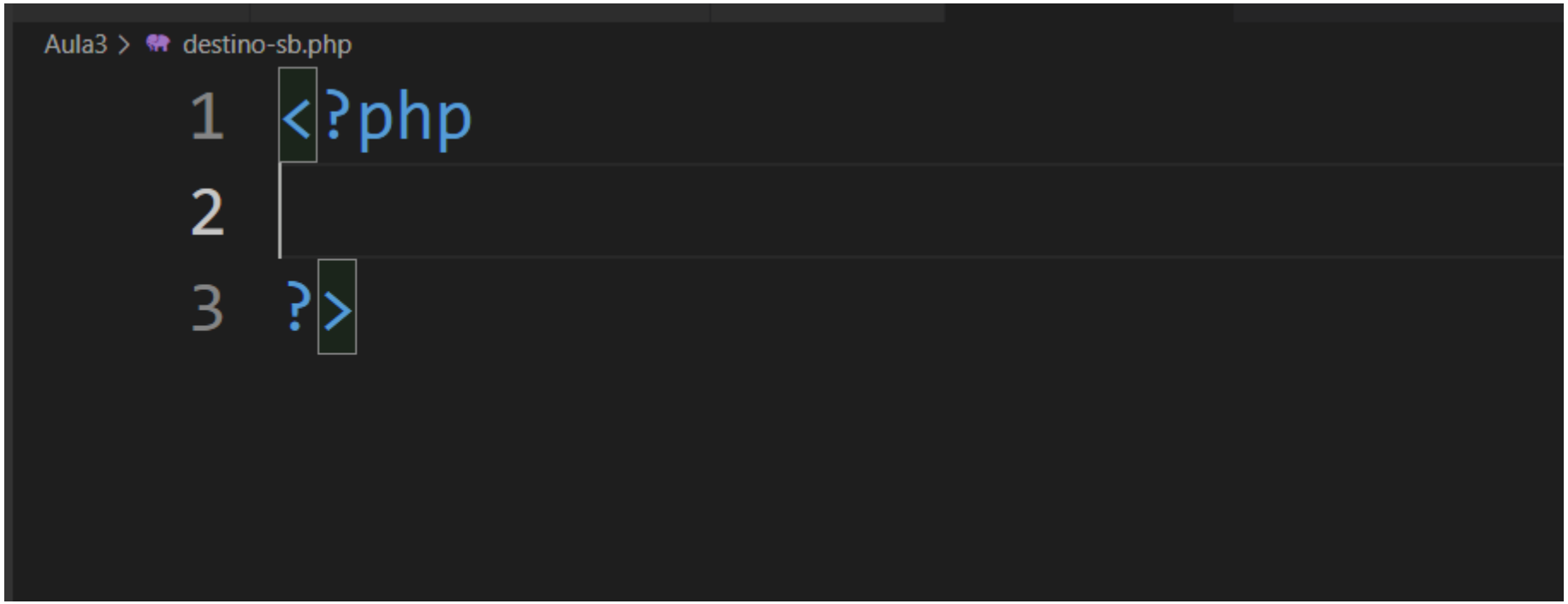
INSTITUTO FEDERAL
São Paulo

É possível criar apenas uma página “vazia”, ou seja, sem a estrutura básica de uma página HTML, porém isso estaria incorreto do ponto de vista do HTML.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Observe,



Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

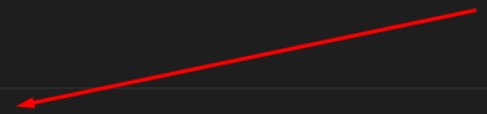
Do ponto de vista do PHP essa página seria funcional, porém ela não seria uma página HTML válida.

Por isso crie a estrutura básica HTML e no interior do BODY coloque o código PHP.

Tratamento de requisições

INSTITUTO FEDERAL São Paulo

```
2  <html lang="pt-br">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Destino</title>
9  </head>
10
11 <body>
12     <h1>Destino</h1>
13     <hr>
14     <?php
15
16     ?>
17 </body>
18 </html>
```



Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Agora vamos mostrar o ***array superglobal*** **$_POST** que é onde as requisições do tipo POST são armazenadas.

O comando comando ***print_r*** é utilizado (ao invés do ***echo***) pois estamos lidando com um *Array*.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

```
11  <body>
12      <h1>Destino</h1>
13      <hr>
14      <?php
15          print_r($_POST);
16      ?>
17  </body>
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Agora execute a página do formulário, preencha as informações e clique em enviar (gravar, cadastrar, etc).

E observe o resultado da página destino.

Tratamento de requisições

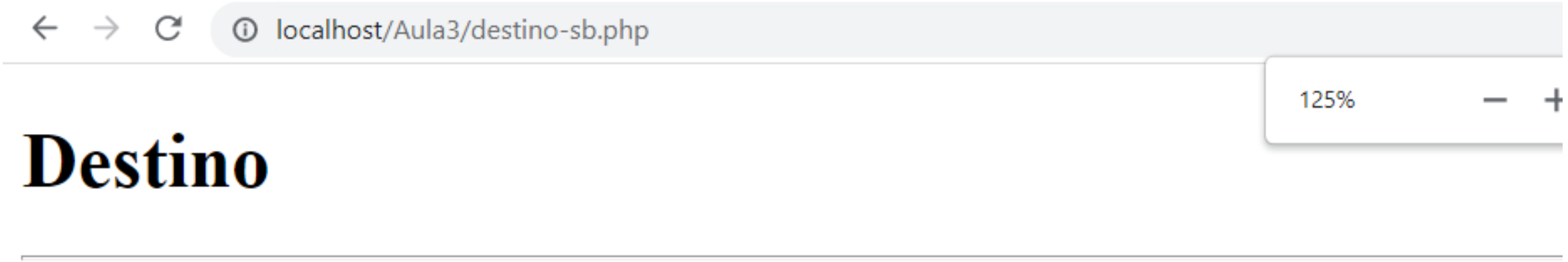
INSTITUTO FEDERAL
São Paulo

Agora execute a página do formulário, preencha as informações e clique em enviar (gravar, cadastrar, etc).

E observe o resultado da página destino.

Tratamento de requisições

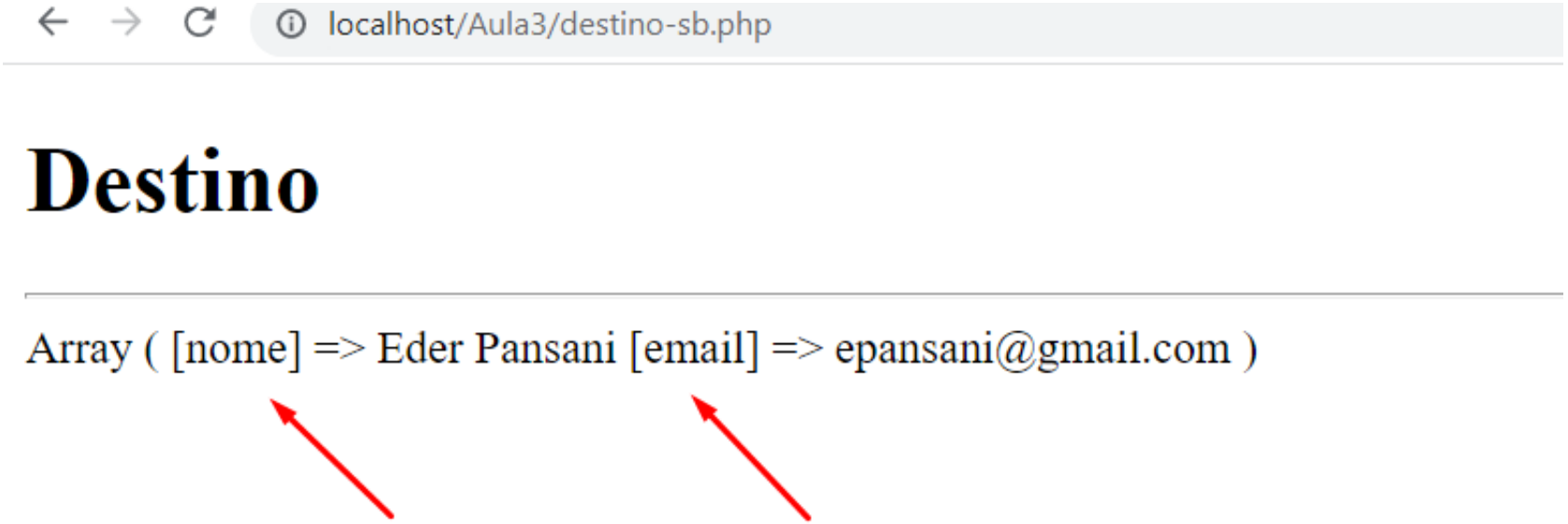
INSTITUTO FEDERAL
São Paulo



Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Preste atenção aos nomes dos índices do *Array*



Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Estes nomes são os nomes que definimos no formulário, especificamente nos atributos "*name*" de cada campo.

```
16      <form action="destino-sb.php" method="POST">
17          <label for="nome">Nome:</label>
18          <input type="text" name="nome" id="nome" required>
19          <br><br>
20
21          <label for="email">E-mail</label>
22          <input type="email" name="email" id="email"
23          placeholder="Digite seu email" required>
24          <br><br>
25
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Se estivéssemos usando o método GET a lógica seria a mesma, porém mudando o “$_POST” por “$_GET”.

Mas isso é só um teste, vamos melhorar o código.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Para pegar cada um dos valores em uma variável distinta podemos utilizar este comando:

```
14      <?php
15          $nome = $_POST["nome"];
16          $email = $_POST["email"];
17
18          echo "<p>Nome informado: $nome<br>Email: $email</p>
19      ?>
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Esta é a forma mais comum, que vocês poderão encontrar em exemplos pela internet.

Porém ela não é a mais adequada, existe uma forma melhor e um pouco mais segura.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

É possível usar a função *filter_input* para pegar os valores:

```php
<?php
    $nome = filter_input(INPUT_POST, "nome", FILTER_SANITIZE_SPECIAL_CHARS);
    $email = filter_input(INPUT_POST, "email", FILTER_SANITIZE_EMAIL);

    echo "<p>Nome informado: $nome<br>Email: $email</p>";
?>
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

A função ***filter_input*** obtém a variável externa pelo nome e opcionalmente a filtra.

https://www.php.net/manual/pt_BR/function.filter-input.php

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Diversos filtros podem ser definidos, para texto se caracteres HTML, e-mails, números inteiros, float, URL entre outros.

Lista de filtros para sanitização (Limpeza):

https://www.php.net/manual/pt_BR/filter.filters.sanitize.php

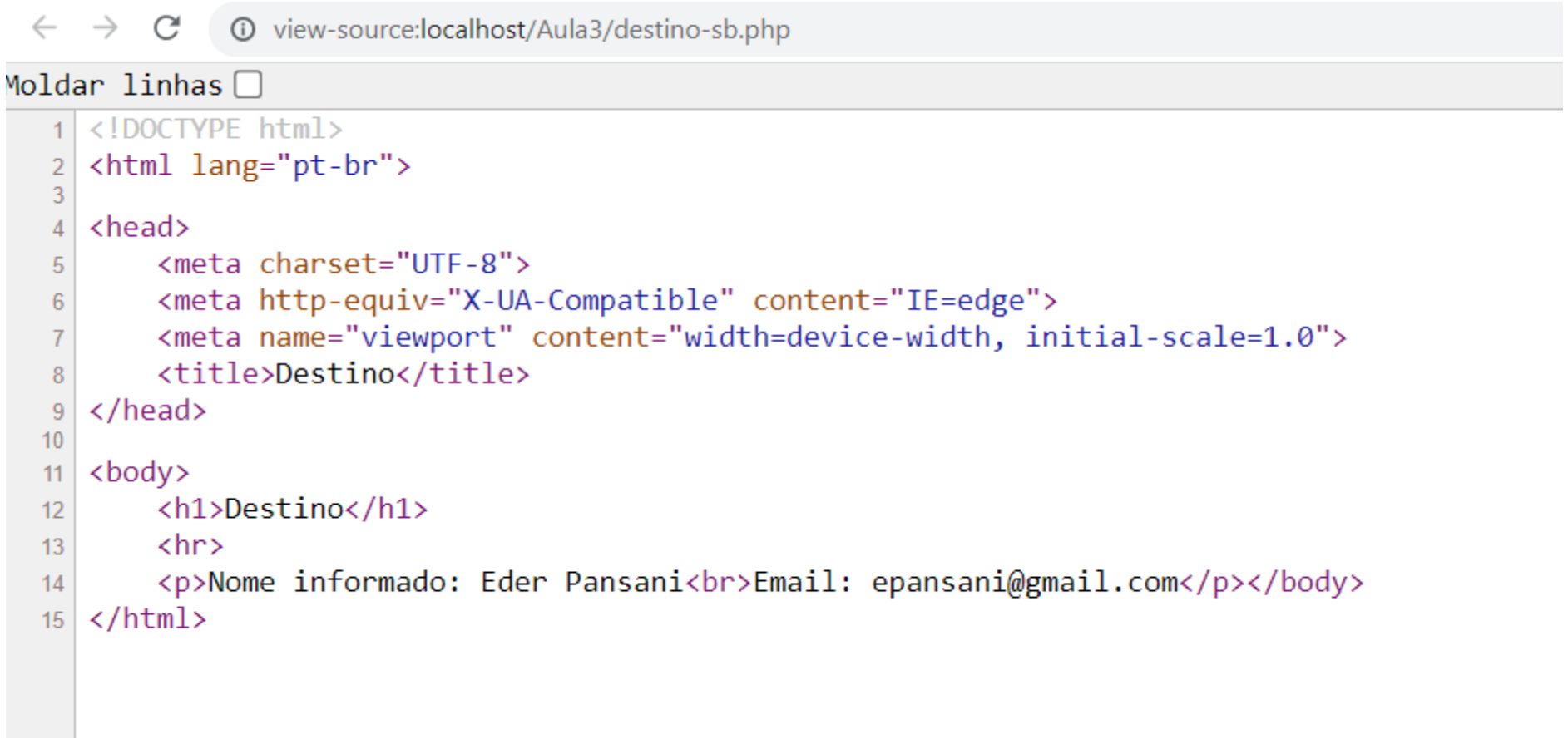
Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Importante, como o **PHP é processado no servidor**, se você tentar exibir o código fonte (opção Exibir-Código-Fonte em seu navegador), verá que não é exibida nenhuma linha em PHP, apenas o código HTML é mostrado.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo




```
1  <!DOCTYPE html>
2  <html lang="pt-br">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Destino</title>
9  </head>
10
11 <body>
12     <h1>Destino</h1>
13     <hr>
14     <p>Nome informado: Eder Pansani<br>Email: epansani@gmail.com</p></body>
15 </html>
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Outro fato importante é que **o código PHP pode ser inserido em qualquer lugar dentro do código HTML e/ou CSS**, desta forma é possível tornar estes conteúdos mais dinâmicos.

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Vá até a página destino.php e adicione o seguinte código:

```
15      <?php
16          $nome = filter_input(INPUT_POST, "nome", FILTER_SANITIZE_SPECIAL_CHARS);
17          $email = filter_input(INPUT_POST, "email", FILTER_SANITIZE_EMAIL);
18
19          echo "<p>Nome informado: $nome<br>Email: $email</p>";
20      ?>
21      <style>
22          body{
23              background-color: <?php echo $nome; ?>;
24          }
25      </style>
26  </body>
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Estamos mesclando o PHP com o HTML.

Muita atenção, observe que código HTML/CSS está fora das tags PHP (<?php e ?>) e código PHP (o echo) está dentro destas tags...

Tratamento de requisições

Por fim volte ao formulário e coloque o nome de uma cor (em inglês) no campo nome:

Tratamento de requisições

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Neste trecho é possível usar uma sintaxe mais curta:

```
20      ?>
21      <style>
22          body{
23              background-color: <?php echo $nome; ?>;
24          }
25      </style>
26  </body>
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Assim:

```
20      ?>
21      <style>
22          body{
23              background-color: <?= $nome; ?>;
24          }
25      </style>
26  </body>
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

A linguagem PHP possui uma *tag* curta “<?= ” que é uma forma abreviada para “<?php echo”.

```
Você pode utilizar a tag curta <?= 'imprima essa string' ?>.
É o equivalente de <?php echo 'print this string' ?>.
```

Tratamento de requisições

INSTITUTO FEDERAL
São Paulo

Mais informações sobre *tags* curtas:

https://www.php.net/manual/pt_BR/language.basic-syntax.phptags.php

INSTITUTO FEDERAL
São Paulo

Praticando

Praticando

INSTITUTO FEDERAL
São Paulo

Vamos agora praticar um pouco com um exercício.

Praticando

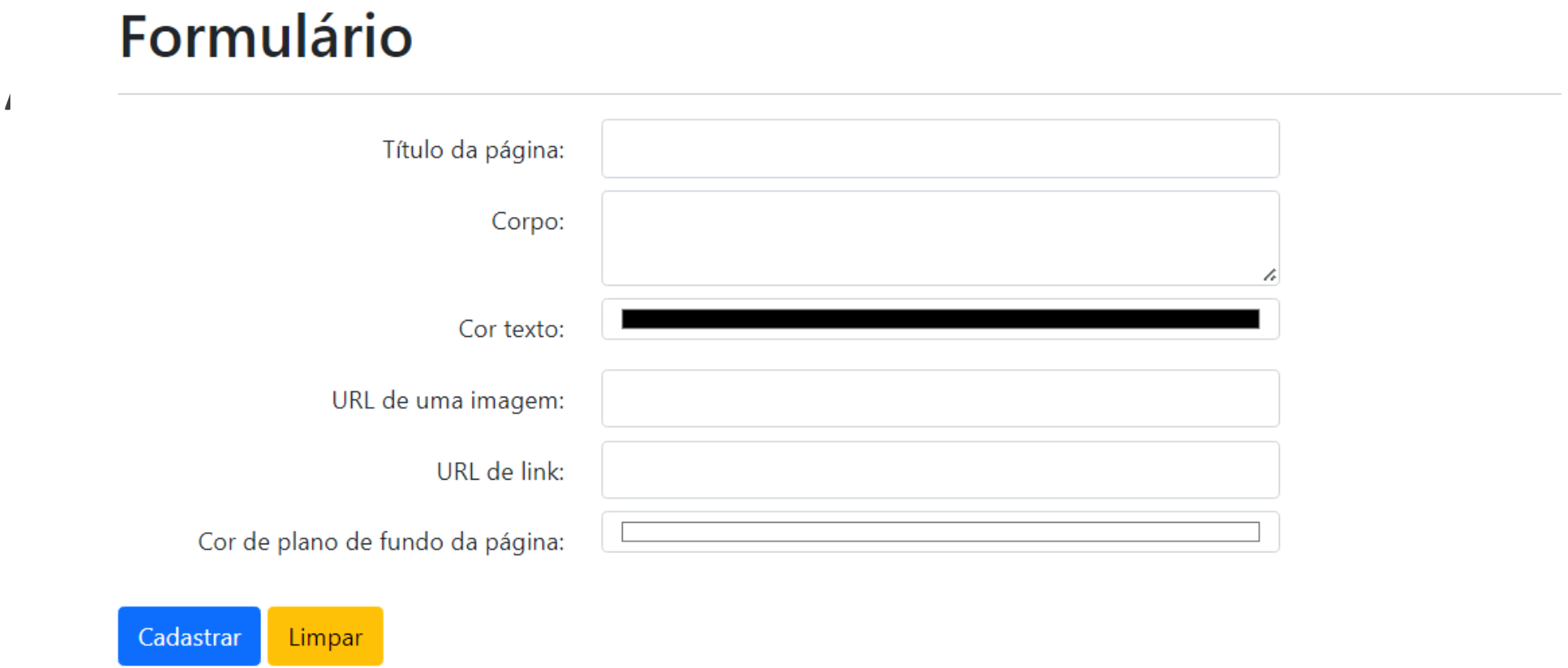
INSTITUTO FEDERAL
São Paulo

Crie um formulário com os campos:

- Título da página (campo texto)
- Corpo (campo texto longo)
- Cor do texto (campo cor)
- URL de uma imagem (campo URL)
- URL de link (campo URL)
- Cor de plano de fundo da página (campo cor)

Praticando

INSTITUTO FEDERAL
São Paulo



Praticando

INSTITUTO FEDERAL
São Paulo

Agora crie uma página de destino, que receberá estes dados (usando o método POST) e criará e formata essa página destino de acordo com as informações (título, texto do corpo, cor do plano de fundo e do texto) provenientes do formulário.

Praticando

INSTITUTO FEDERAL
São Paulo

O campo URL de uma imagem dará origem a uma imagem.

O campo URL de um link dará origem a um link de se abrirá em uma nova janela.

Formulário

Título da página:	IFSP Campus Votuporanga
Corpo:	O Câmpus Votuporanga do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) está com inscrições abertas para o Processo Seletivo
Cor texto:	
URL de uma imagem:	https://vtp.ifsp.edu.br/images/IFSP_Votuporanga_foto.png
URL de link:	https://vtp.ifsp.edu.br
Cor plano de fundo da página:	

Cadastrar Limpar

localhost/Aula3/destino-post.php

IFSP Campus Votuporanga

O Câmpus Votuporanga do Instituto Federal de Educação, Ciência e Tecnologia de São Paulo (IFSP) está com inscrições abertas, para o Processo Seletivo Simplificado dos Cursos Superiores. A classificação ocorrerá por Ordem de Inscrição.
asdsd

https://vtp.ifsp.edu.br

INSTITUTO FEDERAL São Paulo

INSTITUTO FEDERAL
São Paulo

Fim!

Fim!

INSTITUTO FEDERAL
São Paulo

Dúvidas?

Perguntas?

Sugestões?

PROF. DR. EDER PANSANI
EPANSANI@GMAIL.COM / EPANSANI@IFSP.EDU.BR