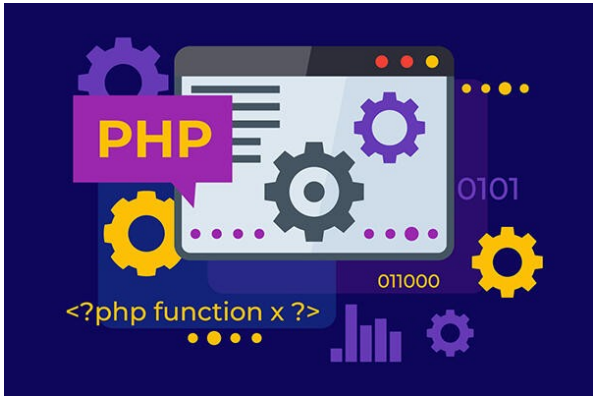
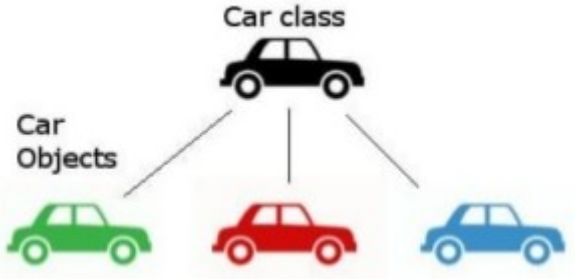
INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

Linguagem PHP

Prof. Daniel Di Domenico

Funções, classes e objetos

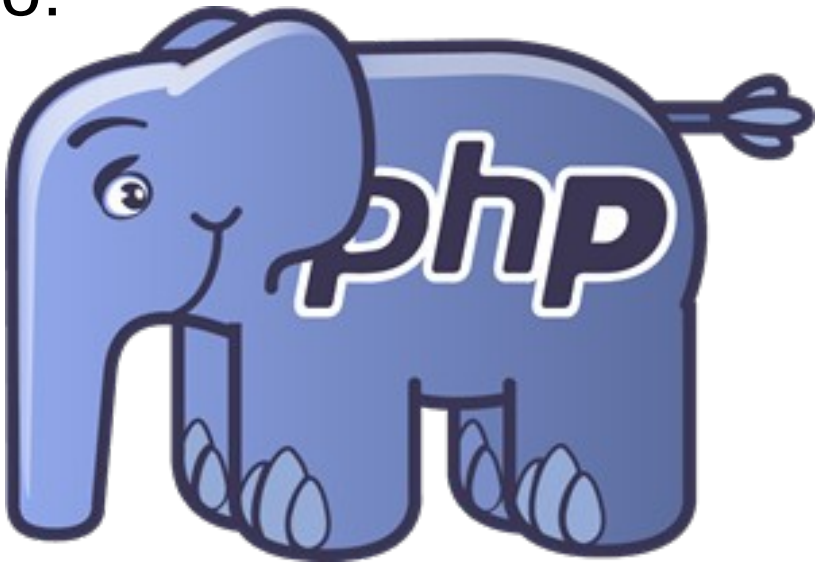




PHP

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- O que já sabemos sobre PHP
  - Características da linguagem
  - Variáveis e tipos
  - Comando de controle de fluxo:
    - IF / ELSE
    - SWITCH
    - WHILE
    - FOR / FOREACH
  - Arrays e constantes



PHP: funções

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Funções são sub-rotinas ou sub-algoritmos que executam comandos específicos

- **Vantagem:** reutilização de código

PHP: funções

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

PHP: funções

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

```
function fazersuco($laranja) {
   cortar $laranja

   espremer $laranja

   return $suco
}
```

PHP: funções - **exemplos**

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

```php
//Com parâmetros e sem retorno
function somar($n1, $n2) {
    echo $n1 + $n2;
}
```

```php
//Sem parâmetros e sem retorno
function dolar(){
    echo 5.25;
}
```

Uso dos parênteses é obrigatório, mesmo sem parâmetros

```php
//Com parâmetros e com retorno
function somar($n1, $n2) {
    return $n1 + $n2;
}
```

```php
//Sem parâmetros e com retorno
function dolar() {
    return 5.25;
}
```

Uso de chaves para o escopo da função é obrigatório

PHP: funções - **chamada**

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Após declarar uma função, basta chamá-la passando os parâmetros necessários

```php
function funcExemplo($param) {
    echo "ARQUIVO: " . __FILE__ . "<br>";
    echo "DIRETÓRIO: " . __DIR__ . "<br>";
    echo "LINHA: " . __LINE__ . "<br>";
    echo "FUNÇÃO: " . __FUNCTION__ . "<br>";
    echo "Parâmetro recebido: " . $param . "<br>";
}

//Chamada da função
funcExemplo(123);
```

PHP: funções - **parâmetros**

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Uma função pode ter parâmetros com **valor padrão**, tornando-se opcionais ao chamá-la

Os parâmetros com valor padrão devem ser os últimos

```php
function login($email, $senha="123") {
    echo "E-mail: " . $email . "<br>";
    echo "Senha: " . $senha . "<br>";
}

//Chamada da função
login("jorge@ifpr.com");
login("ester@ifpr.com", "1234567");
```

Segundo parâmetro não passado: na função, será considerado o valor padrão para **$senha**

PHP: funções - **retorno**

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Uma função pode ter um retorno

```php
function media($v1, $v2, $v3) {
    $media = ($v1 + $v2 + $v3) / 3;
    return $media;
}

//Chamada da função
echo media(2, 4, 7) . "<br>";
$valor = media(1, 22, 10);
echo $valor . "<br>";
```

O retorno da função pode ser utilizado no escopo da chamada

Exercícios

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- **1-** Implemente uma função para calcular o fatorial de um número. Depois, chame essa função para os números de 5 a 12.

- **2-** Implemente duas funções, sendo:
  - a) Calcular e retornar a área de um círculo (PI x Raio x Raio);
  - b) Calcular e retornar a circunferência de um círculo (2 x PI x Raio);

  Ambas funções devem receber o raio do círculo como parâmetro. Para o PI, defina uma constante com o valor de 3.14.

  Faça a chamada da função para 3 círculos com raios distintos.

PHP: orientação a objetos

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- A linguagem PHP suporta orientação a objetos:
  - Conceitos principais:
    - Classes e objetos
    - Atributos e métodos
    - Encapsulamento
    - Modificadores de acesso (público, protegido e privado)
    - Herança
  - Utilizaremos todos esses conceitos para as implementações durante a disciplina

PHP: classes

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Exemplo de uma classe

```php
class Veiculo {
    private $modelo;

    public function __construct() {
        echo "CLASSE: " . __CLASS__ . "<br>";
    }

    public function setModelo($modelo) {
        $this->modelo = $modelo;
        echo "MÉTODO: " . __METHOD__ . "<br>";
    }

    public function getModelo() {
        return $this->modelo;
    }
}
```

O construtor da classe é sempre definido pelo método ***__construct***

O uso de ***$this*** é obrigatório para acesso aos atributos dentro da classe

PHP: objetos

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Exemplo de um objeto

```
//Classe Veiculo declarada no slide anterior

$veiculo = new Veiculo();
$veiculo->setModelo("Gol");
echo $veiculo->getModelo() . "<BR>";
```

ATENÇÃO:
O operador de objetos do PHP é **->**

PHP: herança

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Implementação de uma herança

```
class Ingresso {
   protected $valor;
}

class IngressoVIP extends Ingresso {
   private $valorAdicional;

   public function __construct($valor, $valorAd) {
      $this->valor = $valor;
      $this->valorAdicional = $valorAd;
   }

   public function getValorIngresso() {
      return $this->valor + $this->valorAdicional;
   }
}
```

PHP: herança

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- Exemplo de uso de classe com herança

```php
//Classe IngressoVIP declarada no slide anterior

$ingresso = new IngressoVIP(25, 10);

echo "O valor do ingresso VIP é: R$";
echo $ingresso->getValorIngresso();

echo "<br>";

print_r($ingresso);
```

Impressão do objeto em formado humanizado

Exercícios

INSTITUTO FEDERAL
Paraná
Campus
Foz do Iguaçu

- **1-** Faça um programa com uma classe que possua o nome e o sobrenome de uma pessoa. Esta classe deve ter todos os GETs e SETs dos atributos, bem como um método público para retornar o nome completo da pessoa (nome + sobrenome). Crie objetos para essa classe, sete os atributos e exiba o nome completo de cada pessoa.
- **2-** Faça um programa que declare uma classe Livro com os atributos título, autor, gênero e quantidade de páginas. Após:
  - 2.1: Crie 3 objetos a partir da classe livros;
  - 2.2: Adicione os objetos em um array;
  - 2.3: A partir do array, exiba os atributos dos objetos Livro em uma tabela.