

Curso de PHP

PHP

FATEC - Jundiaí

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Programação Orientada a Objeto (OOP)

A programação orientada a objetos (object-oriented programming OOP) é um conjunto de técnicas para organizar o código em torno de entidades ou objetos representados por seu código.

Esse tipo de organização pode ser uma maneira bastante esclarecedora de pensar sobre problemas de programação, pois ela conduz a uma maneira muito natural de modelar situações do mundo real.

Um benefício extra da OOP é a herança de classes. Uma vez que você define uma classe, pode definir novas subclasses herdadas da classe original, de modo que a subclasse tenha os mesmos tipos de dados e comportamentos, exceto quando você especificar o contrário.

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Programação Orientada a Objeto (OOP)

Antes de entrarmos no OOP, vamos tentar abranger as terminologias usadas dentro do PHP.

Classe: Uma classe é uma coleção de atributos e métodos. Classes são tipos, isto é, “rascunhos” para a criação de objetos.

Objeto: é uma instância de uma classe. Um objeto é uma construção feita a partir de uma classe.

Variável membro: é uma variável identificada que pertence a uma classe particular. Em PHP chamamos de atributos.

Função membro: é uma função específica a uma classe, também são conhecidas como método.

Subclasse: é uma classe que herda, ou estende, uma classe previamente definida.

Superclasse: é uma classe que é herdada ou estendida.

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Programação Orientada a Objeto (OOP)

- Classes e objetos
- Quando usar classes
- Classes em PHP

Hoje em dia, aplicações profissionais são estruturadas em classes, que representam alguma coisa (entidade) que a aplicação tem que manipular.

Classes definem os dados (variáveis) e a funcionalidade (funções) de cada uma dessas entidades.

Objetos são variáveis de uma classe específica que agrupam todas as variáveis correspondentes a uma instância de uma entidade.

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Programação Orientada a Objeto (OOP)

Exemplos de:

Classes

- Curso
- Carro

Objetos

- Curso sobre "PHP"
- Fiat Marea

A classe Curso define variáveis como:

- Título
- Sala
- Duração

O Curso sobre PHP tem os seguintes valores para estas variáveis:

- Nome: Linguagem de Programação – PHP
- Sala: 211
- Duração: 01:00hr

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Programação Orientada a Objeto (OOP)

Funcionalidade da Classe

A classe Curso define uma função para inscrever uma pessoa;

A classe Curso define uma função para dizer se ainda há vagas disponíveis;

A classe Curso define uma função para listar todos os inscritos;

A classe Curso define uma função para alterar dados da pessoa inscrita;

Porque Usar Classes?

Classes podem ser reutilizadas em vários programas;

As classes agrupam todos os dados e funções relacionados a uma entidade, tornando mais fácil modificar ou manter os programas que utilizam esta entidade.

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Programação Orientada a Objeto (OOP)

Classes em PHP

- Podem derivar de outras classes (Herança);
- Suportam construtores (funções com o nome da mesma classe);
- Não tem restrições de visibilidade – todos os atributos e métodos são públicos;
- Declaradas com o comando class;
- Instanciadas com o comando new;
- Acessamos variáveis e funções da classe com o operador ->;
- São auto-referenciáveis pela variável \$this;

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Uma classe é um conjunto de variáveis e funções relacionadas a essas variáveis. Uma vantagem da utilização de programação orientada a objetos é poder usufruir do recurso de encapsulamento de informação. Com o encapsulamento o usuário de uma classe não precisa saber como ela é implementada, bastando para a utilização conhecer a interface, ou seja, as funções disponíveis. Uma classe é um tipo, e portanto não pode ser atribuída a uma variável. Para definir uma classe, deve-se utilizar a seguinte sintaxe:

```
Class nome_classe [extends Classe_Pai]
{
    [atributos]
    [funções]
}
```

Uma classe é utilizada para descrever um objeto. Um objeto poderá utilizar todas as variáveis e funções existentes em sua classe.

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Atributos

Atributos são definidos da seguinte maneira:

```
var $teste = "meu teste";
```

As definições de atributo não podem ser inicializadas com um valor não-estático, ou seja, devemos configurar um valor inicial da variável para uma string, um inteiro, um número de ponto flutuante ou um array, mas não para um objeto.

Um atributo pode ser de qualquer tipo do PHP. Podemos declarar o mesmo atributo quantas vezes quisermos, mas apenas o último a ser inicializado tem efeito.

Atributo declarado em uma classe pode ser inicializado ou não. Se um atributo receber uma atribuição de valor quando for declarado em um classe, ele é inicializado.

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Funções

As funções definidas dentro de uma classe, são definidas da mesma forma com se estivessem fora da classe.

Para acessar um atributo do objeto, utilizamos a sintaxe `$this->variável`. Não utilize a forma `$this->$variável`, pois ela não funciona. A forma da `$this->$variável` funcionará apenas se a `$variável` for uma string contendo o nome do atributo que você deseja utilizar. Isso também funciona para as funções.

```
var $atribuição;  
$nome_do_atributo = "atribuição";  
echo $this->$nome_do_atributo;
```

```
function função1() { ... }  
$nome_da_função = "função1";  
Echo $this->$nome_da_função;
```

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Construtores

Um construtor é uma função especializada definida em uma classe. Tem o mesmo nome da classe e pode receber argumentos. Cada classe pode ter apenas um construtor.

Construtores são chamados automaticamente quando uma instância da classe – um objeto – é criada. Eles são úteis para inicializar os atributos de um objeto com valores de inicialização razoáveis.

```
<?php
class teste {
    var $curso;
    function teste {
        echo "passando por aqui...<br>";
    }
}
?>
```

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Construtores

Um construtor é uma função definida na classe que é automaticamente chamada no momento em que a classe é instanciada (através do operador new). O construtor deve ter o mesmo nome que a classe a que pertence.

```
class conta {
    var $saldo;

    function conta () {
        $this->saldo = 0;
    }

    function saldo() {
        return $this->saldo;
    }

    function credito($valor) {
        $this->saldo += $valor;
    }
}
```

Podemos perceber que a classe conta agora possui um construtor, que inicializa a variável \$saldo com o valor 0.

Um construtor pode conter argumentos, que são opcionais, o que torna esta ferramenta mais poderosa. No exemplo acima, o construtor da classe conta pode receber como argumento um valor, que seria o valor inicial da conta.

Vale observar que para classes derivadas, o construtor da classe pai não é automaticamente herdado quando o construtor da classe derivada é chamado.

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Construtores

```
<?php
class minhaclasse {
    var $var1; // não tem valor inicializado...
    var $var2 = "xyz";
    var $var3 = 100;
    // constructor
    function minhaclasse() {
        $this->var1 = "foo";
        $this->var2 = "bar";
        return true;
    }
}
$teste = new minhaclasse();
$var_classe = get_class_vars(get_class($teste));
foreach ($var_classe as $nome => $valor) {
    echo "$nome : $valor<br>";
}
?>
```

Resultado:

// Antes PHP 4.2.0

var2 : xyz
var3 : 100

// Depois PHP 4.2.0

var1 :
var2 : xyz
var3 : 100

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Herança

Uma classe no PHP pode ser derivada de outra classe. A classe derivada "herda" todas as funções e atributos da classe da qual foi derivada, sem precisar declará-los novamente.

```
class super
{
    function super_echo() {
        echo "Superclasse<br>";
    }
}
```

```
class sub extends super
{
    function sub_echo() {
        echo "Subclasse<br>";
    }
}
```

```
$obj = new sub(); // instancia um objeto
$obj -> sub_echo(); // chama a função da classe
$obj -> super_echo(); // chama função herdada
```

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Herança

Uma classe do PHP só pode herdar de uma outra classe, mas esta superclasse também pode herdar de outra superclasse. Classes em PHP podem ter apenas um pai, mas também podem ter um avô.

```
class super
{
    function super_echo() {
        echo "Superclasse<br>";
    }
}

class sub extends super
{
    function sub_echo() {
        echo "Subclasse<br>";
    }
}

class sub1 extends sub
{
    function sub1_echo() {
        echo "Subclasse 1<br>";
    }
}
```

Avô →

Pai →

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

Herança - Anulação

Quando uma subclasse tem uma função com o mesmo nome da função de uma superclasse, a subclasse anula a função da superclasse.

```
class super
{
    function teste() {
        echo "Mostra Superclasse<br>";
    }
}

class sub extends super
{
    function teste() {
        echo "Mostra Subclasse<br>";
    }
}
```

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

```
<?php
class Pessoa {
    var $nome;
    var $sobrenome;
    function imprime_nome() {
        echo $this->nome . $this->sobrenome . "<br>";
    }
    function muda_nome($novo_nome) {
        $this->nome = $novo_nome;
        $this->imprime_nome();
    }
}
$p = new Pessoa;
$p->nome = "Regina";
$p->sobrenome = "Duarte";
$p->imprime_nome();
$p->muda_nome("Gabriela");
?>
```

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

```
<?php
class CarrinhoDeCompras
{
    var $itens;
    function adiciona_item ($artnr, $num) {
        $this->itens[$artnr] = $num;
    }
    function remove_item ($artnr, $num) {
        if ($this->itens[$artnr] > $num) {
            $this->itens[$artnr] -= $num;
            return true;
        } else {
            return false;
        }
    }
}
$carrinho = new CarrinhoDeCompras;
$carrinho->adiciona_item("Banana", 12);
?>
```

Prof. Cláudio Farias Rossoni

PHP – Aula 6

Classes e Objetos

PHP

Exercício em sala de aula: classe.php

Prof. Cláudio Farias Rossoni

PHP – Aula 6

PHP

Termino da aula

Prof. Cláudio Farias Rossoni