

Programação Orientada a Objetos com PHP

Método Construtor e Encapsulamento

profº Mauricio Conceição Mario

Encapsulamento

O Encapsulamento é um recurso da Orientação a Objetos, um mecanismo que provê proteção de acesso aos membros internos de um objeto. O conceito é de que certas propriedades de uma classe só podem ser manipuladas por métodos dela mesma, como medida de garantir a responsabilidade sobre estas propriedades.

private	Membros declarados private só podem ser acessados dentro da classe em que foram declarados; devem ser acessados através de métodos. Símbolo UML → (-)
protected	Podem ser acessados dentro da classe em que foram declarados e de classes descendentes; não podem ser acessados através de objetos (variáveis de instância) em aplicações que fazem uso da classe onde foi criado. Símbolo UML → (#)
public	Podem ser acessados livremente. Símbolo UML → (+)

Método Construtor:

- Utilizado para definir o comportamento inicial de um objeto (no momento de sua criação);
- O método construtor é executado automaticamente no momento em que se faz uma instância de um objeto através de um operador *new*;
- Quando não é definido um método construtor, automaticamente as propriedades do objeto criado são inicializadas com valor NULL;
- Definição de um método construtor em uma determinada classe: `__construct()`

Método Destrutor (ou finalizador):

- Método executado automaticamente quando o objeto é desalocado da memória, ou quando lhe é atribuído o valor NULL, ou quando é utilizada a função **unset()** sobre o mesmo, ou quando o programa é finalizado.
- O método destrutor pode ser utilizado para finalizar conexões, apagar arquivos temporários criados durante o ciclo de vida do objeto, entre outras possibilidades.



```
1  <?php
2  class Padaria
3  {
4      private $padaria;
5      private $pão_doce;
6
7      function __construct($padaria, $pão_doce){
8          $this->padaria = $padaria;
9          $this->pão_doce = $pão_doce;
10     }
11
12     public function set_padaria($padaria){
13         $this->padaria = $padaria; }
14
15     public function set_pão_doce($pão_doce){
16         $this->pão_doce = $pão_doce; }
17
18     public function get_padaria( ){
19         return $this->padaria;}
20
21     public function get_pão_doce( ){
22         return $this->pão_doce;}
23     }
24     ?>
25
```

neste caso não é necessário o uso dos métodos set(), pois os atributos são inseridos através do método construtor.

C:\xampp\htdocs\Acessa_Padaria.php - Notepad++

Arquivo Editar Localizar Visualizar Formatar Linguagem Configurações Macro Executar Plugins Janela ?



Padaria.class.php x Acessa_Padaria.php x Dados_Complementares.class.php x I_Aluno.php x Manipula_Fazenda.php x Usa_Interface.php x

```
1 <?php
2 include_once 'Padaria.class.php';
3
4 //inserindo valores dos atributos pelo do método construtor
5 $broa_milho = new Padaria("Padaria Jardim Rádio Clube", "pão doce de frutas cristalizadas");
6 //depois de fazer atribuições com método construtor
7 echo nl2br ('depois de fazer atribuições com método construtor'."\n");
8 echo nl2br ($broa_milho->get_padaria()."\n");
9 echo nl2br ($broa_milho->get_pão_doce()."\n");
10
30 ?>
```

localhost/Acessa_Padaria.php x +

← → ↻ ⓘ localhost/Acessa_Padaria.php

depois de fazer atribuições com método construtor
Padaria Jardim Rádio Clube
pão doce de frutas cristalizadas

Exemplo: uso de métodos Construtor e Destrutor.

```
1  <?php
2  class Pessoa
3  {
4      private $código;
5      private $nome;
6      private $altura;
7      private $idade;
8      private $nascimento;
9      private $escolaridade;
10     private $salário;
11
12     //método construtor
13     function __construct($código, $nome, $altura, $idade, $nascimento, $escolaridade, $salário){
14         $this->código = $código;
15         $this->nome = $nome;
16         $this->altura = $altura;
17         $this->idade = $idade;
18         $this->nascimento = $nascimento;
19         $this->escolaridade = $escolaridade;
20         $this->salário = $salário;
21     }
22     public function get_código(){
23         return $this->código;
24     }
25     public function get_nome(){
26         return $this->nome;
27     }
28     public function get_altura(){
29         return $this->altura;
30     }
```

Exemplo: uso de métodos Construtor e Destrutor.

```
27
28
29
30
31 public function get_idade() {
32     return $this->idade;
33 }
34 public function get_nascimento() {
35     return $this->nascimento;
36 }
37 public function get_escolaridade() {
38     return $this->escolaridade;
39 }
40 public function get_salário() {
41     return $this->salário;
42 }
43 //método destrutor
44 function __destruct()
45 {echo "Objeto {$this->nome} finalizado...<br>\n";}
46 }
47 ?>
```

```
Acessa_construtor.php x Usa_Interface.php x Serviços_Reforma.php x Cliente_Reforma.php x aplica_reforma.php x Pessoa.class.php x
1 <?php
2 include_once 'Pessoa.class.php';
3
4 //criação de objeto
5 $Willian = new Pessoa(100304, "Patrick Willian", 1.75, 19, "03/05/1974", "1º ano faculdade", 756.99);
6
7 //manipulação do objeto
8 echo nl2br ("nome: "."\t".$Willian->get_nome()."\n");
9 echo nl2br ("código: "."\t".$Willian->get_código()."\n");
10 echo nl2br ("idade: "."\t".$Willian->get_idade()."\n");
11 echo nl2br ("altura: "."\t".$Willian->get_altura()."\n");
12 echo nl2br ("nascimento: "."\t".$Willian->get_nascimento()."\n");
13 echo nl2br ("escolaridade: "."\t".$Willian->get_escolaridade()."\n");
14 echo nl2br ("salário: "."\t".$Willian->get_salário()."\n");
15
16
17 ?>
18
```

localhost/Acessa_construtor.php x

localhost/Acessa_construtor.php

nome: Patrick Willian
código: 100304
idade: 19
altura: 1.75
nascimento: 03/05/1974
escolaridade: 1º ano faculdade
salário: 756.99
Objeto Patrick Willian finalizado...

Exercício proposto:

- 8a. Criar uma classe Dados_Complementares.class, com dados adicionais da classe Pessoa.class. Utilizar a aplicação Acessa_construtor para inserir e consultar os dados pessoais já existentes e complementares do Sr. Willian.
- 8b. Criar uma classe Conta_Bancária.class, com dados bancários relativos à classe Pessoa.class. Atualizar a aplicação Acessa_construtor para inserir e consultar dados da conta bancária e pessoais do Sr. Willian.



```
1 <?php
2 class Dados_Complementares
3 {
4     private $endereço;
5     private $estado_civil;
6     private $grau_instrução;
7
8     public function __construct($endereço, $estado_civil, $grau_instrução)
9     {
10         $this->endereço = $endereço;
11         $this->estado_civil = $estado_civil;
12         $this->grau_instrução = $grau_instrução;
13     }
14
15     public function get_endereço()
16     {
17         return $this->endereço;
18     }
19
20     public function get_estado_civil()
21     {
22         return $this->estado_civil;
23     }
24
25     public function get_grau_instrução()
26     {
27         return $this->grau_instrução;
28     }
29
30 }
31 ?>
```



```
1 <?php
2 include_once 'Pessoa.class.php';
3 include_once 'Dados_Complementares.class.php';
4
5 //criação de objeto
6 $Willian = new Pessoa(100304, "Patrick Willian", 1.75, 19, "03/05/1974", "1° ano faculdade", 756.99);
7 $complemento = new Dados_Complementares("Rua B nº 20, Jardim Itatins - Peruíbe/SP", "solteiro", "2° grau técnico");
8
9 //manipulação do objeto
10 echo nl2br ("nome:."\t".$Willian->get_nome()."\n");
11 echo nl2br ("código:."\t".$Willian->get_código()."\n");
12 echo nl2br ("idade:."\t".$Willian->get_idade()."\n");
13 echo nl2br ("altura:."\t".$Willian->get_altura()."\n");
14 echo nl2br ("nascimento:."\t".$Willian->get_nascimento()."\n");
15 echo nl2br ("escolaridade:."\t".$Willian->get_escolaridade()."\n");
16 echo nl2br ("salário:."\t".$Willian->get_salário()."\n");
17 echo nl2br ("\nDADOS COMPLEMENTARES\n");
18 echo nl2br ("endereço:."\t".$complemento->get_endereço()."\n");
19 echo nl2br ("estado civil:."\t".$complemento->get_estado_civil()."\n");
20 echo nl2br ("grau de instrução:."\t".$complemento->get_grau_instrução()."\n\n");
21 ?>
22
```



localhost/Acessa_construtor_l.php x



Email – MAURICIO CONCEICAO M x



(10) Geral (PG SI I-A-M-SIST.INTE x



localhost/Acessa_construtor_l.php

nome: Patrick Willian

código: 100304

idade: 19

altura: 1.75

nascimento: 03/05/1974

escolaridade: 1º ano faculdade

salário: 756.99

DADOS COMPLEMENTARES

endereço: Rua B nº 20, Jardim Itatins - Peruíbe/SP

estado civil: solteiro

grau de instrução: 2º grau técnico

Objeto Patrick Willian finalizado...

Referências Bibliograficas

- *php* Programando com Orientação a Objetos
Pablo Dall'Oglio - editora Novatec - 2014
- Desenvolvimento web com PHP e MySQL
Evaldo Junior Bento – editora Casa do Código -
2018