# PROGRAMAÇÃO ORIENTADA A OBJETOS POO

Sistemas para Internet



Prof.º: Paulo Silva

# Modificadores de acesso e Métodos especiais de classe

Aula 2





# Modificadores de acesso.



## Modificadores de acesso



Os modificadores de acesso são padrões de **visibilidade** de acessos às classes, atributos e métodos.

Modificadores são palavras-chaves reservadas pelo PHP, ou seja, palavras reservadas não podem ser usadas como nome de métodos, classes ou atributos.





public, private e protected (público, privado e protegido)



# public (público)



Uma declaração com o modificador **public** pode ser acessada de qualquer lugar e por qualquer entidade que possa visualizar a classe a que ela pertence.

```
<?php

class Conta
{
    protected $saldo;
    public $data_criacao;
}</pre>
```







Os membros da classe definidos como **private** não podem ser acessados ou usados por nenhuma outra classe.

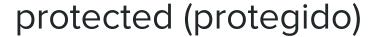
Esses atributos e métodos também não podem ser visualizados pelas classes

herdadas.

```
<?php

class Conta
{
    protected $saldo;
    public $data_criacao;
    private $ativa = false;
}</pre>
```







O modificador **protected** torna o membro acessível às classes do mesmo pacote ou através de herança, seus membros herdados não são acessíveis a outras classes fora do pacote em que foram declarados.

```
<?php
class Conta
{
    protected $saldo;
    public $data_criacao;
    private $ativa = false;
}</pre>
```



## Exercícios



#### 1. Crie as seguintes classes:

#### Funcionario

- + nome: String
- departamento: String
- # salario: float
  - demitido: boolean
  - + demitir()
- # bonificar(float valor)

#### Conta

- + numero: int
- # saldo: float
  - limite: float
  - + consultarLimite()
- + depositar(double valor)
- + sacar(double valor)





# Métodos especiais





# Métodos - Getters e Setters



### Getters e Setters



O modificador private faz com que ninguém consiga modificar, nem mesmo ler, o atributo em questão.

Com isso, temos um problema: como fazer para mostrar o atributos de um objeto, já que nem mesmo podemos acessá-lo para leitura?







Usamos **Getters** para obter informações Esse tipo de método sempre retorna um valor.



# Setters - Métodos Modificadores



Usamos **Setters** para definir valores. Esse tipo de método geralmente não retorna valores.



# Exercícios



1. Crie os métodos getters e setters dos atributos privados e protegidos das classes de **Funcionario** e **Conta**.





# Métodos Construtores



# Métodos Construtores



Quando usamos a palavra chave *new*, estamos construindo um objeto.

Sempre quando o *new* é chamado, ele executa o **construtor** da classe.

O construtor da classe é um bloco declarado como:







```
<?php
class Conta
   protected $saldo;
   public $data criacao;
   private $ativa;
   public function construct($saldo, $data criacao, $ativa)
       $this->saldo = $saldo;
       $this->data criacao = $data criacao;
       $this->ativa = $ativa;
```



## O construtor default



Até agora, as nossas classes não possuíam nenhum construtor. Então como é que era possível dar *new*, se todo *new* chama um construtor obrigatoriamente?

Quando você não declara nenhum construtor na sua classe, o PHP cria um para você. Esse construtor é o construtor default, ele não recebe nenhum argumento e o corpo dele é vazio.

A partir do momento que você declara um construtor, o construtor default não é mais fornecido.





# Obrigado!







MANZANO, José Augusto G., COSTA JR., Roberto da. **Programação de Computadores com Java**. Érica, 2014.

FURGERI, Sérgio. Java 8 - Ensino Didático - Desenvolvimento e Implementação de Aplicações. Érica, 2015.

