

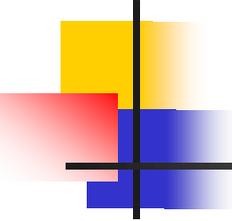


Padrão de Projeto *Singleton*

Universidade Católica de Pernambuco
Ciência da Computação

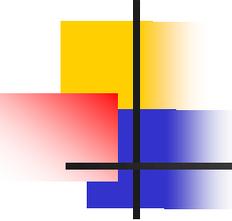
Prof. Márcio Bueno
pooite@marciobueno.com

Fonte: Material da Prof^a Karina Oliveira



Padrão de Projeto *Singleton*

- Objetivo
 - Garantir que uma determinada classe tenha uma **única instância** (único objeto) no sistema.
- Onde aplicar
 - Em muitos contextos pode ser útil ter a garantia de existência de apenas um objeto de uma dada classe, exemplos:
 - Uma fachada única para um dado sistema;
 - Um único gerenciador de conexões para acesso a banco de dados.

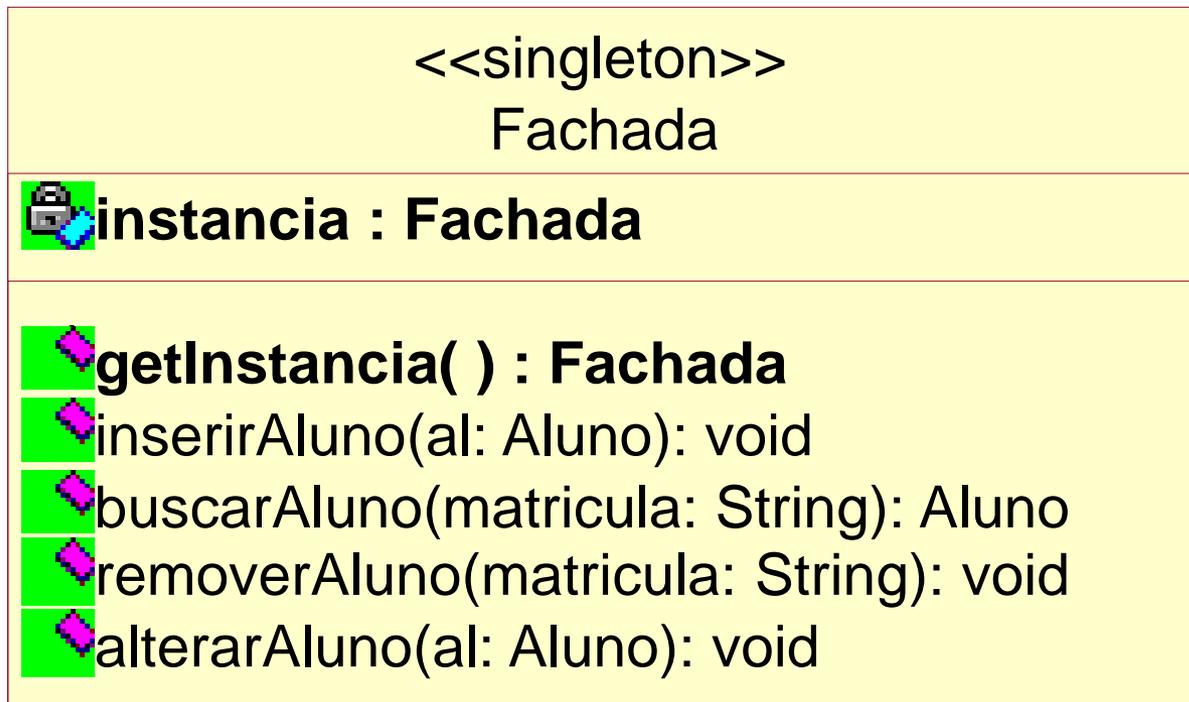


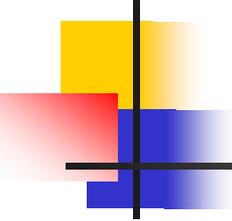
Implementação em Java

1. O **construtor da classe** que utiliza o padrão *singleton* deve ser **privado** para evitar que diversos objetos da classe sejam criados sem controle;
2. Um **atributo estático privado** do mesmo tipo da classe é declarado para manter a instância (objeto) única da classe;
3. Um **método estático público *getInstancia()*** é implementado para criar a única instância da classe e permitir que outros objetos do sistema tenham acesso a mesma.

Implementação em Java

- Exemplo: Sistema de Cadastro de Alunos

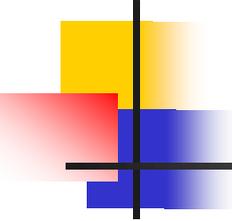




Implementação em Java

- Fachada alterada para utilizar o padrão *singleton*

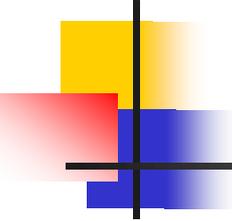
```
public class Fachada implements IFachada {  
    private static Fachada instancia;  
    private Fachada ( ) {  
        this.controladorAluno = new ControladorAluno( );  
    }  
    public static Fachada getInstancia( ) {  
        if (instancia == null) {  
            instancia = new Fachada( );  
        }  
        return instancia;  
    }  
}
```



Implementação em Java

- Acessando a Fachada implementada com *singleton*

```
public class Aplicacao {  
    private static IFachada fachada = new Fachada( );  
    public static void main(String[ ] args) {  
  
        ...  
    }  
}
```



Implementação em Java

- Acessando a Fachada implementada com *singleton*

```
public class Aplicacao {  
    private static IFachada fachada = Fachada.getInstancia( );  
    public static void main(String[ ] args) {  
  
        ...  
    }  
}
```