

Padrão Singleton

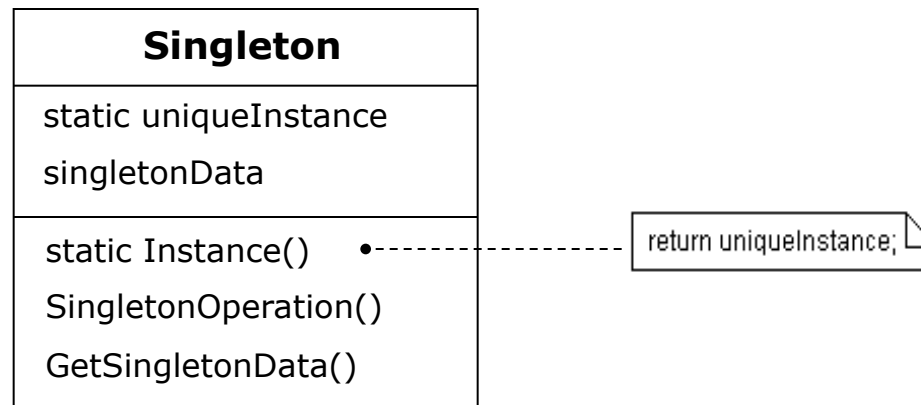
Singleton

- Propósito
 - Garantir que uma classe só tenha uma única instância, e prover um ponto de acesso global a ela.

Singleton - Motivação

- Garantir que apenas um objeto exista, independente do número de requisições que receber para criá-lo.
- Exemplos de aplicação
 - Um único banco de dados
 - Um único acesso ao arquivo de log
 - Uma única fachada (padrão Facade)

Singleton - Estrutura



O construtor é privado.

Singleton – Exemplo de código

```
public class Highlander {  
    private Highlander() {}  
    private static Highlander instancia;  
    public static Highlander obterInstancia() {  
        if (instancia == null)  
            instancia = new Highlander();  
        }  
        return instancia;  
    }  
}
```

*Esta classe
implementa o
design pattern
Singleton*

```
public class Fabrica {  
    public static void main(String[] args) {  
        Highlander h1, h2, h3;  
        //h1 = new Highlander(); // nao compila!  
        h2 = Highlander.obterInstancia();  
        h3 = Highlander.obterInstancia();  
        if (h2 == h3) {  
            System.out.println("h2 e h3 são mesmo objeto!");  
        }  
    }  
}
```

*Esta classe
cria apenas
um objeto
Highlander*