

Suporte para GitHub:

Carregando arquivos pelo navegador: https://drive.google.com/open?id=1Klf0HCJcB_4Q5B7efMXrs_YYDXGwH89U
Tutorial Básico GitHub com Eclipse e EGit Usando Chave SSH: <http://www.youtube.com/watch?v=fFB5azTSGZw>
Usando Github com Github Desktop em Projetos Eclipse: <http://www.youtube.com/watch?v=EgHljYyS4U>
Usando Github com SSH no Terminal Linux com chave gerada no Eclipse: <http://www.youtube.com/watch?v=0s699q5Sja4>
Usando Github com SSH no Terminal Windows com chave gerada no Eclipse: <http://www.youtube.com/watch?v=DaydwPB2WSI>

Para cada exercício, criar um projeto no Github para submeter à tarefa.

Material:

a) Princípios SOLID - <https://youtu.be/7cbGZ0MyfRI>

Exercícios (Resolver descrevendo as soluções e responder a tarefa com DOC ou PDF):

1. Considere o seguinte cenário:

Imagine que você tem uma classe Product que descreve produtos de uma loja e você deseja calcular o desconto sobre um produto, sabendo que naquele existem produtos que a loja não dá desconto. Existe uma interface IDiscount com a assinatura (calculateDiscount()). Entretanto, agora a loja vai dar 20% de desconto para quem for cliente antigo e 50% de desconto para funcionário, qual princípio SOLID está sendo violado? o que fazer com o método de calcular desconto?

2. Considere o seguinte cenário:

Existe uma Classe Employer (Funcionario) que é responsável por criar um funcionário, calcular o salário e gerar um relatório sobre as atividades do mês do departamento com os métodos calculateSalary() e generateReport(). Qual princípio SOLID está sendo violado? o que fazer com os métodos?

3. Considere o seguinte cenário:

Dentro do sistema, há uma classe-mãe Estudante, que representa um estudante de graduação, e a filha dela, EstudantePosGraduacao, tendo o seguinte código:

```
public class Estudante {  
    String nome;  
  
    public Estudante(String nome) {  
        this.nome = nome;  
    }  
  
    public void estudar() {  
        System.out.println(nome + " está estudando.");  
    }  
}
```

```
public class EstudanteDePosGraduacao extends Estudante {  
  
    @Override  
    public void estudar() {  
        System.out.println(nome + " está estudando e pesquisando.");  
    }  
}
```

Para adicionar a funcionalidade entregarTCC() ao sistema, basta colocar esse método na classe Estudante

O código fica assim:

```
class Estudante {  
    String nome;  
  
    public Estudante(String nome) {  
        this.nome = nome;  
    }  
  
    public void estudar() {  
        System.out.println(nome + " está estudando.");  
    }  
  
    public void entregarTCC(){  
        //...  
    }  
}
```

Problemática

Normalmente, estudantes de pós-graduação não entregam TCCs, no entanto, a classe EstudanteDePosGraduacao é filha de Estudante. Uma alternativa seria sobrescrever o método entregarTCC na classe EstudanteDePosGraduacao lançando uma exceção.

Qual princípio SOLID está sendo violado ? o que fazer com os métodos?