

Command

Temos a seguinte classe que representa uma fila de trabalho:

```
class Fila_de_trabalho(object):

    def __init__(self):
        self.__comandos = []

    def adiciona(self, comando):
        self.__comandos.append(comando)

    def processa(self):
        for comando in self.__comandos:
            comando.executa()
```

Ela espera receber comandos que devem ser processados sobre um pedido. É por isso que todos os comandos que herdam de `Comando` devem receber um pedido em seu construtor.

Agora temos a seguinte classe abstrata que representa um comando:

```
from abc import ABCMeta, abstractmethod

class Comando(object):

    __metaclass__ = ABCMeta

    @abstractmethod
    def executa(self):
        pass
```

Agora, implemente dois comandos que herdam de `Comando`: o que finaliza um pedido e o que paga um pedido. Depois de prontos, instancie uma `Fila_de_trabalho` que deve receber os dois comandos criados que operam sobre um pedido.

Cole todo o código aqui. ```

Responda

INSERIR CÓDIGO	FORMATAÇÃO
<div></div>	

