

DA1MCTA018-13SA - Programação Orientada a Objetos - Paulo Henrique Pisani - 2021.2

[Painel](#) / [Meus cursos](#) / [POO - DA1MCTA018-13SA - 2021.2](#) / [Classes, objetos, encapsulamento, UML](#) / [\[EP\] Porta](#)

Descrição

[Visualizar envios](#)

[EP] Porta

Data de entrega: segunda, 14 Jun 2021, 23:59
Arquivos requeridos: Porta.java ([Baixar](#))
Tipo de trabalho: Trabalho individual

Exercício: Classe Porta

Autor: Guilherme Aldeia. POO - QS/2020

Exercício de programação orientada a objetos. Implementação de uma classe Porta, utilizando os conceitos de **modificadores de acesso e métodos modificadores**.

Tarefa

Para esse exercício, queremos uma classe que represente uma porta da vida real, que pode ser aberta ou fechada. O objeto poderá ser instanciado diversas vezes.

Cada instância de porta pode ser aberta ou fechada, e deve ter um atributo interno que mantenha registro do estado da porta. O utilizador deve ser capaz de atribuir um nome para a porta, obter o nome da porta, obter o estado da porta (se está aberta ou não), abrir a porta e fechar a porta.

Porta
- aberta : boolean = true - nome : String = "Porta sem nome"
+ setNome(nome: String) + getNome() : String + estaAberta() : boolean + abrir() : int + fechar() : int

Crie a classe pública **Porta** em um arquivo contendo apenas a implementação da classe (nomeie o arquivo como **<Porta.java>** para a submissão).

Importante: O programa principal já existe no sistema de correção automática. Submeta apenas a classe Porta. A classe Porta não pode realizar impressão de dados ou utilizar import.

O estado da porta e o seu nome não podem ser modificados diretamente, portanto devem ser variáveis privadas. Espera-se o seguinte comportamento das funções:

- **setNome(nome : String)**
 - Recebe um novo nome para a porta e salva internamente
- **getNome(): String**
 - Retorna o nome da porta
- **estaAberta(): boolean**
 - retorna verdadeiro caso a porta esteja aberta, e retorna falso caso esteja fechada
- **abrir(): int**
 - altera o estado da porta se estiver fechada, retornando 0 para indicar sucesso. Caso a porta esteja aberta, não altera o estado e retorna 1
- **fechar(): int**
 - análogo ao método anterior, sendo bem sucedida se a porta estiver aberta

Exemplo de uso da classe

```
Porta portaSala = new Porta();
portaSala.setNome("Porta da sala");

Porta portaCozinha = new Porta();
portaCozinha.setNome("Porta da cozinha");

if (portaSala.estaAberta()) {
    portaCozinha.fechar();
}
```

[◀ \[EP\] Menu de Operações 2](#)

Seguir para...

[\[EP\] Lâmpada Inteligente ▶](#)

[VPL](#)



Este é o Ambiente Virtual de Aprendizagem da UFABC para apoio ao ensino presencial e semipresencial. Esta plataforma permite que os usuários (educadores/alunos) possam criar cursos, gerenciá-los e participar de maneira colaborativa.

Informação

[Conheça a UFABC](#)

[Conheça o NTI](#)

[Conheça o Netel](#)

Contato

Av. dos Estados, 5001. Bairro Bangu - Santo André /SP – Brasil. CEP 09210-580.

Siga-nos



[Universidade Federal do ABC](#) - [Moodle](#) (2020)

[Obter o aplicativo para dispositivos móveis](#)