

Programação orientada a objetos: relacionamento entre objetos

Exercício 1 *Refazer a AV1 (fazer observações).*

Exercício 2 *Fora da classe principal:*

- *Implemente a classe Cliente. A classe tem os atributos “nome”, “CPF” e “endereço”. Implemente também o método “mostra”;*
- *Implemente a classe Conta. A classe tem os atributos “cliente” e “saldo”. Implemente também o método “mostra”. Obs.: o saldo não pode ser negativo.*

Na classe principal:

- *Declare e inicialize um objeto do tipo Conta. Mostre os atributos do objeto fazendo uma chamada ao método específico;*
- *Receba um inteiro n. Declare e inicialize um vetor de n objetos do tipo Conta. Mostre os dados do cliente que possui o maior saldo. Mostre a média dos saldos.*

Exercício 3 *Com relação ao exercício anterior. Fora da classe principal:*

- *Implemente a classe Operacao. A classe tem os atributos “tipo” (D – depósito ou R – retirada) e “valor”.*

Em seguida:

- *Declare e inicialize um objeto do tipo Conta, que, inicialmente, deve possuir saldo 0;*
- *Na classe Conta, implemente o método fazOperacao. O método tem como parâmetro um objeto do tipo Operacao para atualizar o saldo da conta. Obs.: a operação de retirada só é permitida se o saldo não ficar negativo;*
- *Receba um inteiro m. Faça m operações sobre a conta.*

Exercício 4 *Fora da classe principal:*

- *Implemente a classe Aluno. A classe tem os atributos “nome”, “idade” e “média”. Implemente também o método “mostra”;*
- *Implemente a classe Turma. A classe tem como atributos um vetor de n objetos do tipo Aluno, o “curso” da turma e o “ano de ingresso”. Implemente também o método “mostra”;*
- *Na classe Turma, implemente um método que retorna a maior média;*
- *Na classe Turma, implemente um método que retorna o objeto Aluno que possui a maior média.*

Na classe principal:

- *Declare e inicialize um objeto do tipo Turma. Mostre os atributos do objeto fazendo uma chamada ao método específico.*