

Universidade Católica de Pernambuco

Professor: Augusto César Oliveira

Disciplina: Programação III / POO

Aluno(a): _____ data: ____/____/____

Aula 13 - Tratamento de exceções

1. Faça um programa em JAVA que leia do usuário o nome e um número. Utilize um dos métodos da classe String para obter a letra do nome que se localiza na posição do número informado. Por exemplo: caso o usuário digite "José" e "2", o programa deve imprimir na tela "s", que corresponde ao caractere na posição dois, considerando que o primeiro caractere tem posição zero. Faça o tratamento de exceção para quando o usuário informar um número inválido, o programa mostrar a mensagem "Posição inválida". Por exemplo, se o usuário digitar a posição "10" para o nome "José".
2. Faça um programa em JAVA que leia a idade de 10 pessoas e armazene em um ArrayList. Em seguida, leia um número do usuário e remova da lista a idade localizada na posição informada. Por exemplo, se o usuário informar o número "4", a idade armazenada na posição quatro da lista deve ser removida. Utilize um dos métodos da classe ArrayList para remover um elemento da lista. Faça o tratamento de exceção para quando o usuário informar um número inválido, o programa mostrar a mensagem "Posição inválida". Por exemplo, se o usuário digitar a posição "30".
3. Crie uma classe que aceite a digitação de dois números e faça a divisão entre eles exibindo seu resultado. Sua classe deve tratar as seguintes exceções:
 - a. `ArithmeticException` quando ocorrer uma divisão por zero.
 - b. `InputMismatchException` quando o valor informado não é numérico.
4. Crie uma classe e declare um vetor de inteiros de 10 posições. Feito isso, permita que o usuário digite valores inteiros a fim de preencher este vetor. Não implemente nenhum tipo controle referente ao tamanho do vetor, deixe que o usuário digite valores até que a entrada 0 seja digitada. Uma vez digitado o valor 0, o mesmo deve ser inserido no vetor e a digitação de novos elementos deve ser interrompida. Feita toda a coleta dos dados, exiba-os em tela. Sua classe deve tratar as seguintes exceções:
 - a. `ArrayIndexOutOfBoundsException` quando o usuário informar mais do que 10 valores.
 - b. `InputMismatchException` quando o usuário informar um valor que não é numérico.
5. Você foi contratado para desenvolver um sistema educacional infantil focado em aprendizado de geometria. Modele uma classe chamada "Triangulo" utilizando os conceitos de abstração e encapsulamento. Crie a exceção `DimensoesInvalidasException` e lance essa exceção no construtor e no método set dos lados quando suas dimensões forem inválidas. Considere que um triângulo só é válido se qualquer de seus lados forem maiores que o valor absoluto (módulo) da diferença dos outros dois lados e menor que a soma dos outros dois lados. Após modelar a classe Triangulo e a exceção `DimensoesInvalidasException` siga os seguintes passos:
 - a. Instanciar um objeto Triangulo com lados 3, 4 e 5;
 - b. Mudar a dimensão do primeiro lado para 10;
 - c. Ler do usuário três números;
 - d. Instanciar um objeto Triangular usando os valores lidos no passo anterior como lados.

6. Você foi contratado para desenvolver o sistema de um banco. Modele a classe "**ContaCorrente**" utilizando os conceitos de abstração e encapsulamento. Crie a exceção `SaldoInsuficienteException` e lance esta exceção sempre que alguém tentar sacar um valor maior que o saldo disponível na conta. Após modelar a classe `ContaCorrente` e a exceção `SaldoInsuficienteException` siga os seguintes passos:
- a. Ler do usuário um número de conta e saldo inicial;
 - b. Instanciar um objeto `ContaCorrente` com o número e saldo informado no passo anterior;
 - c. Pedir que o usuário digite o valor de 10 saques na conta.