Orientação a objetos

Tratamento de exceções

Tratamento de exceções

- Em Java, uma exceção é um evento que ocorre durante a execução do programa, que interrompe o fluxo normal de execução.
- As exceções são geralmente causadas por erros no código, como uma divisão por zero, tentar acessar uma variável que não foi inicializada, entre outros.
- O tratamento de exceções é uma técnica utilizada para lidar com esses erros, para que o programa não pare abruptamente.
- Ao lidar com exceções, podemos exibir mensagens de erro ao usuário e tentar corrigir o problema no código.

Exemplos de exceções – Exemplo 1

```
3  public class Principal {
4
5      public static void main(String[] args) {
6          // Exemplo 1: Tentando acessar um elemento fora do índice de um array
7          int[] numeros = {1, 2, 3};
8          System.out.println(numeros[3]); // ArrayIndexOutOfBoundsException
9      }
10 }
```

Exemplos de exceções – Exemplo 2

```
5⊖    public static void main(String[] args) {
6         // Exemplo 2: Divisão por zero
7         int a = 10;
8         int b = 0;
9         System.out.println(a/b); // ArithmeticException
10    }
```

Exemplos de exceções – Exemplo 3

```
3  public class Principal {
4
5      public static void main(String[] args) {
6          // Exemplo 3: Tentando converter uma String em um número
7          String numero = "abc";
8          int valor = Integer.parseInt(numero); // NumberFormatException
9      }
10 }
```

Tratando exceções

- Para lidar com exceções em Java, utilizamos o bloco try-catch.
- O bloco try é onde escrevemos o código que pode gerar uma exceção.
- O bloco catch é onde escrevemos o código que trata a exceção.

```
3  public class Principal {
4
5      public static void main(String[] args) {
6          try {
7              // Código que pode gerar uma exceção
8          } catch (Exception e) {
9              // Código que trata a exceção
10         }
11     }
12 }
```

Tratando exceções

```
3  public class Principal {
4
5      public static void main(String[] args) {
6          try {
7              // Código que pode gerar uma exceção
8          } catch (Exception e) {
9              // Código que trata a exceção
10         }
11     }
12 }
```

Tratando exceções

- Estamos tentando acessar um elemento fora do índice de um array.
- Se isso acontecer, o código dentro do bloco catch será executado, exibindo a mensagem de erro "Erro: índice fora do array".

```
5      public static void main(String[] args) {
6          try {
7              int[] numeros = {1, 2, 3};
8              System.out.println(numeros[3]); // ArrayIndexOutOfBoundsException
9          } catch (ArrayIndexOutOfBoundsException e) {
10             System.out.println("Erro: índice fora do array");
11         }
12     }
13 }
```

Tratando exceções

```
5⊖    public static void main(String[] args) {
6         try {
7             int[] numeros = {1, 2, 3};
8             System.out.println(numeros[3]);
9         } catch (Exception e) {
10            System.out.println("Exceção: "+e.toString());
11        }
12    }
```

Tratando exceções

```
public static void main(String[] args) {
    try {
        int[] numeros = {1, 2, 3};
        System.out.println(numeros[3]);
    } catch (ArrayIndexOutOfBoundsException e) {
        System.out.println("Erro: índice fora do array \n"+e.getMessage());
    }
}
```

Tratando exceções

```
public static void main(String[] args) {
    try {
        int[] numeros = {1, 2, 3};
        System.out.println(numeros[3]);
    } catch (ArrayIndexOutOfBoundsException e) {
        System.out.println("Erro: índice fora do array \n"+e.getMessage());
    }catch (Exception e) {
        System.out.println("Exceção: "+e.toString());
    }
}
```

Tratamento de exceções

- Além das exceções nativas do Java, é possível criar suas próprias exceções personalizadas.

- Para criar uma exceção personalizada, é necessário criar uma classe que estenda a classe Exception ou uma de suas subclasses.

Criando exceções

```
3  public class MinhaExcecao extends Exception {
4      public MinhaExcecao() {
5          super();
6      }
7
8      public MinhaExcecao(String message) {
9          super(message);
10     }
11 }
```

Criando exceções

- Na classe MinhaExcecao é uma exceção personalizada que estende a classe Exception.
- Ela possui dois construtores, um que não recebe parâmetros e outro que recebe uma mensagem de erro como parâmetro.
- Para lançar uma exceção personalizada em seu código, basta usar a palavra-chave throw seguida de uma instância da sua exceção personalizada.

Criando exceções

```
5  public class Principal {
6
7      public static void main(String args[]) throws MinhaExcecao {
8          Scanner sc = new Scanner (System.in);
9          double n1, n2;
10         System.out.println("Digite dois numeros: ");
11         n1 = sc.nextDouble();
12         n2 = sc.nextDouble();
13         divide(n1,n2);
14     }
15
16     public static void divide(double n1, double n2) throws MinhaExcecao {
17
18         if (n2== 0) {
19             throw new MinhaExcecao("Impossível realizar divisão por 0");
20         }
21         else {
22             System.out.println("O resultado da divisão é: "+(n1/n2));
23         }
24     }
25 }
```

```java
public class Principal {

    public static void main(String args[]) throws MinhaExcecao {
        Scanner sc = new Scanner (System.in);
        double n1, n2;
        System.out.println("Digite dois numeros: ");
        n1 = sc.nextDouble();
        n2 = sc.nextDouble();
        divide(n1,n2);
    }

    public static void divide(double n1, double n2) throws MinhaExcecao {

        if (n2== 0) {
            throw new MinhaExcecao("Impossível realizar divisão por 0");
        }
        else {
            System.out.println("O resultado da divisão é: "+(n1/n2));
        }
    }
}
```

Desafio

- 1) Crie uma classe abstrata chamada Conta com as atributs Saldo e Limite. Crie também métodos abstratos para Depositar e Sacar.

- 2) Faça implementação concreta desta classe com os métodos Depositar e sacar

- 3) Crie exceções caso o valor do saque seja negativo ou maior que o saldo disponível e se o valor do depósito seja negativo.