

# TRATAMENTO DE ERROS





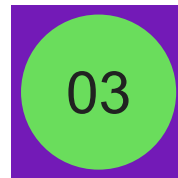
01

Tipos de erros



02

Throwable

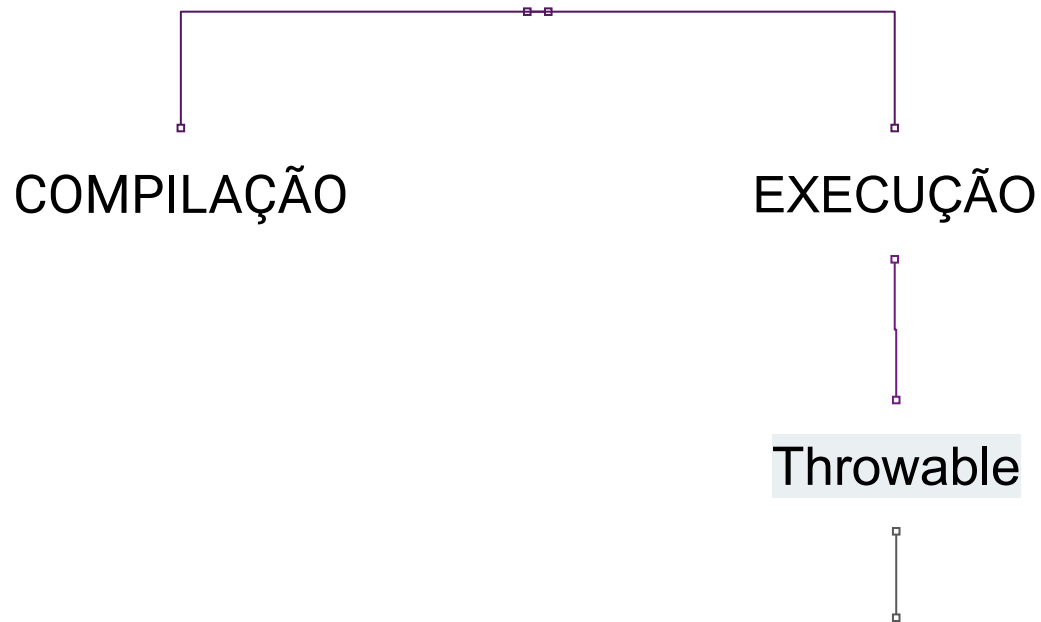


03

Exceções checadas e  
não checadas

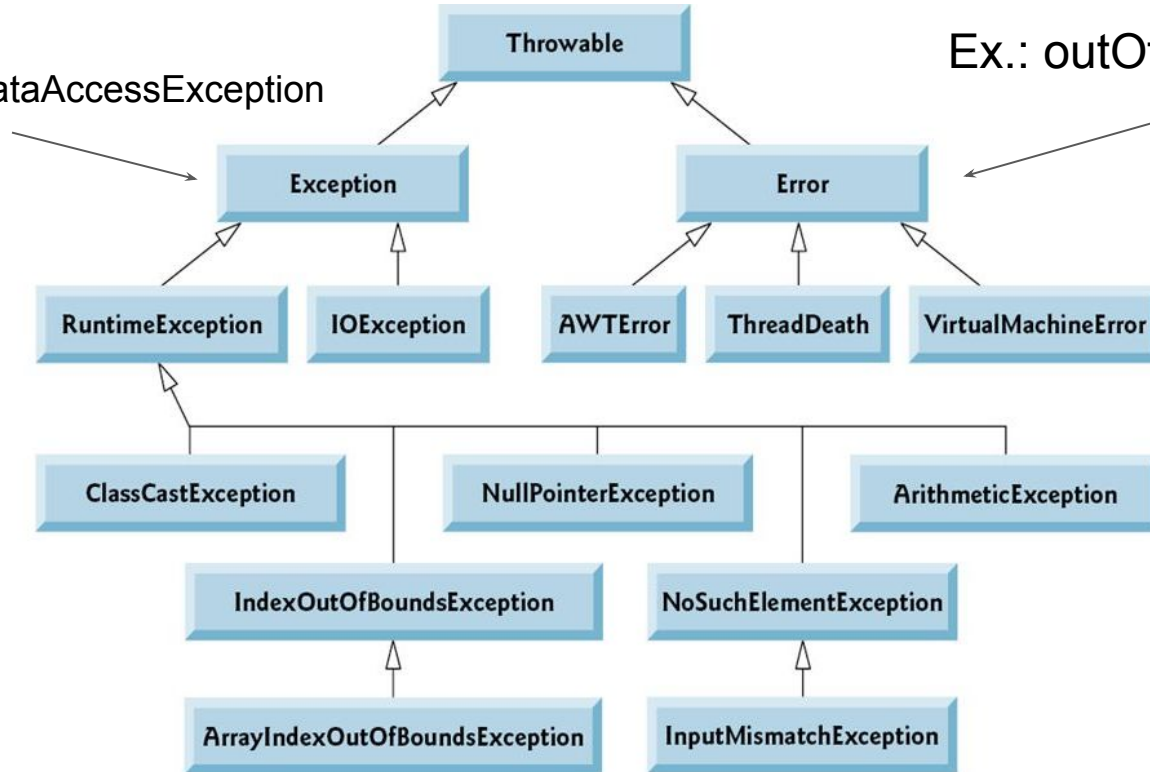
# Tipos de erros

---



# Throwable

Ex.: EmptyResultDataAccessException



Ex.: outOfMemoryError

# Exceções checadas e não checadas

---

As Checked Exceptions são exceções que, havendo alguma falha dentro do método, o desenvolvedor DEVERÁ executar operações para tratar estes erros...

Exemplo: SQLException, IOException, ParseException

As uncheckedExceptions são todas as exceptions derivadas da classe RuntimeException, ou seja, são exceções que podem ser lançadas durante operações realizadas normalmente pela JVM e que não precisam necessariamente ser tratadas.

Exemplos: NullPointerException, ArrayIndexOutOfBoundsException, ClassCastException e NumberFormatException

# Try / Catch

---

A instrução ***try*** permite definir um bloco de código a ser testado quanto a erros enquanto está sendo executado.

A instrução ***catch*** permite definir um bloco de código a ser executado, **se ocorrer um erro no bloco try**.

## Sintaxe

```
try {  
    // Block of code to try  
}  
catch(Exception e) {  
    // Block of code to handle errors  
}
```

# Exceções aula

Produto p = null;

```
try {  
    nomeDoProduto(p);  
} catch (Exception e) {  
    System.out.println(e.getMessage());  
}
```

```
//NullPointerException  
//ArithmeticException
```

```
try {  
    System.out.println(10/0);  
} catch (Exception e) {  
    System.out.println(e.getMessage());  
}
```

```
System.out.println("Fim!");
```

```
public class Produto {  
  
    String nome;  
    Double preco;  
  
    public Produto() {  
    }  
  
    +getters  
    +setters
```

# Finally

---

A instrução **finally** permite que você execute o código, depois **try...catch**, independentemente do resultado:

```
public class Main {  
    public static void main(String[] args) {  
        try {  
            int[] meusNumeros = {1, 2, 3};  
            System.out.println(meusNumeros[10]);  
        } catch (Exception e) {  
            System.out.println("Algo deu errado.");  
        } finally {  
            System.out.println("O 'try catch' está concluído.");  
        }  
    }  
}
```



# Algumas Exceções

---

**ArrayIndexOutOfBoundsException** ocorre quando tenta-se acessar um elemento após o final de um array;

**ClassCastException** ocorre quando faz-se uma tentativa de fazer uma coerção em um objeto que não tem um relacionamento é um com o tipo especificado no operador de coerção;

**NullPointerException** ocorre quando uma referência null é utilizada onde um objeto é esperado;

Somente classes que **estendem Throwable** (pacote java.lang ) direta ou indiretamente podem ser utilizadas com o tratamento de exceções.

VAMOS PRATICAR?