# Tratamento de Exceções

**RunTimeException** 🡪 é um tipo de exceção não verificada (**unchecked**) em Java.

- Isso significa que **o compilador não obriga** você a tratá-la. **Mas quebra o código**

*Exemplos:*

*NullPointerException*

*ArrayIndexOutOfBoundsException*

*ArithmeticException (divisão por zero)*

Ou seja, são erros que acontecem em tempo de execução por falhas lógicas ou descuido do programador

**Checked Exceptions** 🡪 O compilador **obriga** você a tratar com try/catch ou declarar com throws no método.

- São filhas diretas de Exception (mas não de RuntimeException).

⚠️ Se você esquece de tratar, o código nem compila.

**Throwable** 🡪 É o pai de todas as exceções e erros

- Erros são problemas graves que não devem ser tratados pelo programa *(ex: falta de memória OutOfMemoryError, erro de stack StackOverflowError)*

**🔹 Blocos importantes**

- No Java, o tratamento é feito com os blocos try, catch, finally e throw.

* try → onde você coloca o código que pode gerar exceção.
* catch → captura e trata a exceção.
* finally → bloco opcional, sempre executa (muito usado para fechar recursos, como arquivos ou conexões).
* throw → usado para lançar uma exceção manualmente.
* throws → usado na assinatura do método para indicar que ele pode lançar exceções.