

Nome: Beatriz Cardoso de Oliveira	NUSP: 12566400
Nome: Beatriz Lomes da Silva	NUSP: 12548038
Nome: Heitor Tanoue de Mello	NUSP: 12547260
Nome: Pedro Oliveira Torrente	NUSP: 11798853

1. Uma exceção é uma classe em java que representa um erro que ocorre durante a execução de um programa. Basicamente, quando acontece algo inesperado ao rodar o código, isso se dá por meio de erros de lógica ou acesso a recursos inacessíveis. As exceções verificadas (checked) são instâncias de subclasses do tipo `Exception` e ocorrem durante a execução do código, mas não dependem da lógica do programa, normalmente associadas às entradas e saídas. Já as exceções não-verificadas (unchecked) são instâncias de subclasse do tipo `RuntimeException`, estas estão associadas a erros ou defeitos na lógica do programa que podem causar problemas irreversíveis em tempo de execução (run time).

Exemplos de Checked:

- `IOException`
- `ClassNotFoundException`
- `CloneNotSupportedException`

Exemplos de Unchecked:

- `NumberFormatException`
- `NullPointerException`
- `ArrayIndexOutOfBoundsException`

2. O bloco *finally* é utilizado para garantir que uma execução de código depois de um *try*, mesmo que uma *Exception* tenha sido jogada. Mesmo que tenha um *return* no *try* ou no *catch*, o bloco *finally* é sempre executado.

Um exemplo da sua utilização é fechar arquivos e outras coisas abertas após um erro acontecer.

3. Cada comando *try* delimita uma região do programa onde falhas são detectadas e define os *catches* que capturam e tratam falhas. A hierarquização das exceções dá muita flexibilidade de programação, mas ressalta-se que a ordem de apresentação dos *catches* não é livre. A ordem, de cima para baixo, deve ser de uma classe mais específica para a mais geral da hierarquia. Isso porque a busca pelo *catch* é feita de cima para baixo, sendo concluída quando for encontrado o primeiro que satisfizer o teste é um relativo ao tipo do seu parâmetro. A correção da ordem é automaticamente verificada pelo compilador, que, por exemplo, não aceita a ordenação de *catches* usada no método errado do programa abaixo, porque o primeiro *catch* iria capturar todas as exceções de sua hierarquia, e, consequentemente, as cláusulas que o seguem seriam inúteis.