**Multi-Catch**

A partir do Java 7, a linguagem introduziu uma nova funcionalidade chamada "multi-catch", que permite capturar várias exceções em um único bloco catch. Essa funcionalidade pode tornar o código mais conciso e legível, reduzindo a repetição de código.

O uso de multi-catch é muito simples. Em vez de ter vários blocos catch para lidar com diferentes exceções, você pode agrupá-las em um único bloco usando o caractere | para separar as exceções. Por exemplo, suponha que você tenha escrito o seguinte código:

try {

metodoQuePodeLancarExcecao();

} catch (NumberFormatException e) {

System.out.println("tratando erro...");

} catch (IllegalArgumentException e) {

System.out.println("tratando erro...");

}COPIAR CÓDIGO

Como o tratamento do erro é o mesmo para ambas as exceções, o código anterior poderia ter sido escrito utilizando o multi-catch:

try {

metodoQuePodeLancarExcecao();

} catch (NullPointerException | IllegalArgumentException e) {

System.out.println("tratando erro...");

}COPIAR CÓDIGO

No exemplo anterior, estamos lidando com duas exceções diferentes: NullPointerException e IllegalArgumentException. Se qualquer uma dessas exceções for lançada dentro do bloco try, o mesmo bloco catch será executado.

Uma observação importante de lembrar, é que o uso de multi-catch só é permitido para exceções que não estão relacionadas por uma hierarquia de herança. Se duas exceções compartilham uma hierarquia de herança, você deve lidar com elas em blocos catch separados.