Reading 17: A Third Look at Java

# Exercise 1: Summarize

Java uses exceptions, derived from the Throwable class, to handle errors and unexpected events through try, catch, and finally blocks, distinguishing between checked and unchecked exceptions, and allowing custom exceptions for more reliable and debuggable programs.

# Exercise 2: Demonstrate & Explain

In this program, the divide method attempts to divide 10 by 0, which raises an ArithmeticException. The try block contains the code that might throw an exception. When the exception occurs, the catch block catches it and prints an error message. Regardless of whether an exception is thrown, the finally block is executed, printing a message to indicate its execution. The output will show the exception message and the finally block message.

public class Test {

public static void main(String[] args) {

try {

int result = divide(10, 0);

System.out.println("= " + result);

} catch (ArithmeticException e) {

System.out.println(e.getMessage());

} finally {

System.out.println("Complete");

}

}

public static int divide(int a, int b) {

return a / b;

}

}

# Exercise 3: Inquire

How does Java handle multiple catch blocks, and what happens if an exception matches more than one catch block?