**Assignment 2- Kotlin Higher-Order Functions**

// Part 1: Higher-Order Function Basics

// Task 1: Simple Higher-Order Function

fun applyOperation(a: Int, b: Int, operation: (Int, Int) -> Int): Int {

return operation(a, b)

}

// Example Usage for Task 1

fun main() {

val resultAddition = applyOperation(5, 3) { a, b -> a + b }

println("Result of addition: $resultAddition")

val resultSubtraction = applyOperation(8, 3) { a, b -> a - b }

println("Result of subtraction: $resultSubtraction")

val resultMultiplication = applyOperation(4, 7) { a, b -> a \* b }

println("Result of multiplication: $resultMultiplication")

}

// Task 2: Filtering with Higher-Order Function

fun filterNumbers(numbers: List<Int>, predicate: (Int) -> Boolean): List<Int> {

return numbers.filter(predicate)

}

// Example Usage for Task 2

fun main() {

val numbers = listOf(1, -2, 3, -4, 5)

val positiveNumbers = filterNumbers(numbers) { it > 0 }

println("Positive Numbers: $positiveNumbers")

val evenNumbers = filterNumbers(numbers) { it % 2 == 0 }

println("Even Numbers: $evenNumbers")

}