Debugging Exercise 1: Array Manipulation

Error : i < numbers.length

public class ArrayManipulation { public static void main(String[] args) {

int[] numbers = {1, 2, 3, 4, 5};

for (int i = 0; i < numbers.length; i++)

{ System.out.println(numbers[i]);

}

}

}

Explanation : The correct loop condition should be i < numbers.length to avoid an ArrayIndexOutOfBoundsException.

Debugging Exercise 2: Object-Oriented Programming

Error : stop() method is absent in Car class.

class Car {

private String make; private String model;

public Car(String make, String model) {

this.make = make; this.model = model;

}

public void start() {

System.out.println("Starting the car.");

}

public void stop() {

System.out.println("Stoping the car.");

}

}public class Main {

public static void main(String[] args) { Car car = new Car("Toyota", "Camry");

car.start(); car.stop();

}

}

Explaination : The Car class doesn't have a stop method defined, and you're trying to call it in the Main class. To fix this, you can add a stop method to the Car class.

Debugging Exercise 3: Exception Handling

Error : no error.

public class ExceptionHandling { public static void main(String[] args) {

int[] numbers = {1, 2, 3, 4, 5};

try {

System.out.println(numbers[10]);

} catch (ArrayIndexOutOfBoundsException e) {

System.out.println("Array index out of bounds.");

}

int result = divide(10, 0);

System.out.println("Result: " + result);

}

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

return a / b;

}

}

Exercise 4:

Error : No error

public class Fibonacci { public static int fibonacci(int n) {

if (n <= 1) return n;

else return fibonacci(n-1) + fibonacci(n-2);

}

public static void main(String[] args) {

int n = 6;

int result = fibonacci(n);

System.out.println("The Fibonacci number at position " + n + " is: " + result);

}

}

Exercise4:

Error : No error.

import java.util.\*;

public class PrimeNumbers { public static List<Integer> findPrimes(int n) { List<Integer> primes = new ArrayList<>(); for (int i = 2; i <= n; i++) { boolean isPrime = true; for (int j = 2; j < i; j++) { if (i % j == 0) {

isPrime = false; break;

}

}

if (isPrime) { primes.add(i);

}

}

return primes;

}

public static void main(String[] args) {

int n = 20;

List<Integer> primeNumbers = findPrimes(n);

System.out.println("Prime numbers up to " + n + ": " + primeNumbers);

}

}