

# Debugging Exercise 1: Solutions

## Code 1:

Text

```
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]);  
        }  
    }  
}
```

**Error:** Array out of bounds exception as the loop is trying to access an element from an index that is not present. The for loop condition is wrong as it is trying to access elements from index 0 to 5 but only 4 indices are present. The corrected code is below.

```
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]);  
        }  
    }  
}
```

## Code 2:

```
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 class Main {
    public static void main(String[] args) {
        Car car = new Car("Toyota", "Camry");
        car.start();
        car.stop();
    }
}

```

**Error:** the method stop() is not defined hence only start() method will be printed. To resolve the error we need to define stop() method.

```

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("Stopping the car.");
    }
}

public class Main {
    public static void main(String[] args) {
        Car car = new Car("Toyota", "Camry");
        car.start();
        car.stop();
    }
}

```

Output: Starting the car.  
Stopping the car.

## CODE 3:

```
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;  
    }  
}
```

**Error:** In the divide method, attempting to divide 10 by 0 is not allowed in Java and results in an Arithmetic Expression error. This exception is not caught in the code, so it will terminate the program with an error. Hence it should be used inside a try catch block.

```
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.");  
    }  
}
```

```

try {
    int result = divide(10, 0);
    System.out.println("Result: " + result);
} catch (ArithmeticException e) {
    System.out.println("Cannot divide by zero.");
}
}

```

```

public static int divide(int a, int b) {
    return a / b;
}

```

## CODE 4:

```

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);
    }
}

```

**NO ERROR**

## CODE 5.

```

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);
}
}

```

NO error