

Week 1

Assignment

Debugging Exercise1:

There should be a ; at the end of print statement.

The correct code would be:

```
public class Main{  
    public static void main (String[] args){  
        System.out.println("Hello, World!");  
    }  
}
```

Debugging Exercise2:

```
public class Main {  
    public static void main(String[] args) {  
        int[] numbers = {1, 2, 3, 4,5};  
        System.out.println(numbers[5]);  
    }  
}
```

In the above code the array starts from index0 and ends at index4. So, there is no index5 and `ArrayIndexOutOfBoundsException` is thrown.

To get the last element of the array use the below print statement instead:

```
System.out.println(numbers[4]);
```

Debugging Exercise3:

In the below code, the Dog class has a parameter food in eat method indicating the type of food the dog eats. This parameter overloads the method.

```
class Animal {  
    void eat() {  
        System.out.println("This animal eats food.");  
    }  
}
```

```
class Dog extends Animal {  
    void eat(String food) {  
        System.out.println("This dog eats " + food + ".");  
    }  
}
```

```
public class Main {  
    public static void main(String[] args) {  
        Animal myDog = new Dog();  
        myDog.eat(); // This will call the eat() method in Animal class, not in Dog class  
    }  
}
```

Inheritance: The process of a sub class inheriting the fields and methods of a base class is called Inheritance.

Polymorphism: It is the ability of a variable to behave differently based on the object it is referring to.

In method overloading, a class can have multiple methods with the same name but different parameters. The method that should be executed is defined at the compile time.

In method overriding, the method of a sub class has an implementation that is defined in the super class. Dynamic method dispatch is used at runtime.