Problem Statement 1:

1. Print "Hello" and Your Name on Separate Lines

```
public class HelloName {
    public static void main(String[] args) {
        System.out.println("Hello");
        System.out.println("Your Name"); // Replace 'Your Name' with your actual name
    }
}
```

---

2. Print the Sum of Two Numbers

```
public class SumOfTwoNumbers {
    public static void main(String[] args) {
int num1 = 10;        int num2 = 20;
int sum = num1 + num2;
        System.out.println("The sum of " + num1 + " and " + num2 + " is: " + sum);
    }
}
```

---

3. Program Using Primitive Data Types

```
public class PrimitiveDataTypes {     public
static void main(String[] args) {        double
doubleValue = 5.75;        char charValue = 'A';
float floatValue = 3.14f;        boolean
booleanValue = true;
        int intValue = 42;
        System.out.println("Double Value: " + doubleValue);
        System.out.println("Char Value: " + charValue);
        System.out.println("Float Value: " + floatValue);
        System.out.println("Boolean Value: " + booleanValue);
        System.out.println("Int Value: " + intValue);
    }
```

```
}
```

---

4. Calculate Remainder Without

Modulus Operator public class

```
RemainderWithoutModulus {     public static
void main(String[] args) {
      int a = 10;        int b = 3;
int remainder = a - (a / b) * b;
      System.out.println("The remainder when " + a + " is divided by " + b + " is: " + remainder);
   }
}
```

---

5. Code Snippet Analysis and

Explanation The provided code snippet has

two errors:

1. Public should be lowercase: public.
2. Into should be corrected to int.

Here's the corrected code: public class

```
ModulusError {     public static void
main(String[] args) {        Double a =
10.0;
      int b = 5;
      System.out.println(a % b);
   }
}
```

OP:
0.0
Explanation: 10.0 % 5 = 0.0.

---

6. Type Casting int to float

```
public class IntToFloatCasting {
```

```
public static void main(String[]
args) {        int intValue = 42;
float floatValue = (float)
intValue; // Explicit type casting
        System.out.println("The int value: " + intValue);
        System.out.println("The float value after casting: " + floatValue);
    }
}
```

Problem Statement 2:

---

1. Add, Subtract, Multiply, and Divide Two Numbers

```
import java.util.Scanner; public class
ArithmeticOperations {     public static void
main(String[] args) {
        Scanner sc = new Scanner(System.in);
System.out.println("Enter two numbers:");
double num1 = sc.nextDouble();        double
num2 = sc.nextDouble();
        System.out.println("Addition: " + (num1 + num2));
        System.out.println("Subtraction: " + (num1 - num2));
        System.out.println("Multiplication: " + (num1 * num2));
        System.out.println("Division: " + (num1 / num2));
    }
}
```

---

2. Sum Up to N Numbers and Sum of Divisors

```
import java.util.Scanner; public class
SumCalculator {     public static void main(String[]
args) {
```

```
    Scanner sc = new Scanner(System.in);
System.out.println("Enter a number N:");
int n = sc.nextInt();

    int sum = 0;        for (int
i = 1; i <= n; i++) {           sum
+= i;
    }
    System.out.println("Sum of first " + n + " numbers: " + sum);
System.out.println("Enter a number to find sum of divisors:");
int num = sc.nextInt();        int divisorSum = 0;        for (int i = 1; i
<= num; i++) {           if (num % i == 0) {               divisorSum += i;
       }
    }
    System.out.println("Sum of divisors of " + num + " is: " + divisorSum);
  }
}
```

---

3. Greatest of Three Numbers

```
import java.util.Scanner; public class
GreatestOfThree {    public static void
main(String[] args) {
    Scanner sc = new Scanner(System.in);
System.out.println("Enter three numbers:");
int a = sc.nextInt();        int b = sc.nextInt();
int c = sc.nextInt();
    int greatest = Math.max(a, Math.max(b, c));
    System.out.println("The greatest number is: " + greatest);
  }
}
```

---

4. Print Values in Even Cells of an Array

```
public class EvenCellsArray {
public static void main(String[] args) {
int[] array = {10, 20, 30, 40, 50};
      System.out.println("Values in even cells:");
      for (int i = 0; i < array.length; i += 2) {
         System.out.println("Cell " + i + ": " + array[i]);
      }
   }
}
```

---

5. Rectangle Class with Default and Parameterized Constructor

```
class Rectangle {    private int length;
private int breadth;    // Default constructor    public Rectangle() {
this.length = 10;       this.breadth = 10;
   }
   // Parameterized constructor    public Rectangle(int length, int breadth) {
this.length = length;       this.breadth = breadth;
   }
   public int getArea() {       return this.length * this.breadth;
   }
   public void display() {
      System.out.println("Length: " + this.length + ", Breadth: " + this.breadth);
   }
}
```

```
public class TestRectangle {     public static
void main(String[] args) {        Rectangle
defaultRect = new Rectangle();
defaultRect.display();
      System.out.println("Area: " + defaultRect.getArea());
Rectangle paramRect = new Rectangle(15, 20);
paramRect.display();
      System.out.println("Area: " + paramRect.getArea());
   }
}
```

---

6. Member Class with Employee and Manager Subclasses

```
class Member {     String name;
   int age;
   String phoneNumber;
String address;     double
salary;     public void
printSalary() {
      System.out.println("Salary: " + salary);
   }
}
class Employee extends Member {
   String specialization;

   public void display() {
      System.out.println("Employee Name: " + name);
      System.out.println("Age: " + age);
      System.out.println("Phone Number: " + phoneNumber);
      System.out.println("Address: " + address);
      System.out.println("Specialization: " + specialization);
      printSalary();
```

```
    }
}
class Manager extends Member {
String department;     public void
display() {
        System.out.println("Manager Name: " + name);
        System.out.println("Age: " + age);
        System.out.println("Phone Number: " + phoneNumber);
        System.out.println("Address: " + address);
System.out.println("Department: " + department);
        printSalary();
    }
}
public class MemberTest {     public
static void main(String[] args) {
Employee emp = new Employee();
emp.name = "John Doe";        emp.age
= 30;
        emp.phoneNumber = "1234567890";
emp.address = "123 Elm Street";        emp.salary =
50000;        emp.specialization = "Software
Development";

        Manager mgr = new Manager();
mgr.name = "Jane Smith";        mgr.age
= 45;        mgr.phoneNumber =
"0987654321";
        mgr.address = "456 Oak Street";
mgr.salary = 80000;        mgr.department =
"IT";
```

```
        System.out.println("Employee Details:");
emp.display();

        System.out.println("\nManager Details:");
mgr.display();
    }
}
```

---

7. Check Prime Number by Passing

Parameter public class PrimeChecker {

```
public static boolean isPrime(int number) {
if (number <= 1) return false;

        for (int i = 2; i <= Math.sqrt(number); i++) {
if (number % i == 0) return false;
        }
        return true;
    }
    public static void main(String[] args) {        int num =
29; // Replace with any number to check
        if (isPrime(num)) {
            System.out.println(num + " is a prime number.");
        } else {
            System.out.println(num + " is not a prime number.");
        }
    }
}
```