Problem 1: Create a Java program that defines and uses methods to perform arithmetic operations.

Problem Statement: Write a Java program that defines four methods: `add`, `subtract`, `multiply`, and `divide`. Each method should take two numbers as arguments and return the result of the corresponding operation. In the `main` method, call these methods and display the results of the operations on two input numbers.

Solution:

```
java
Copy code
import java.util.Scanner;

public class ArithmeticOperations {

    // Method to add two numbers
    public static int add(int num1, int num2) {
        return num1 + num2;
    }

    // Method to subtract two numbers
    public static int subtract(int num1, int num2) {
        return num1 - num2;
    }

    // Method to multiply two numbers
    public static int multiply(int num1, int num2) {
        return num1 * num2;
    }

    // Method to divide two numbers
    public static double divide(int num1, int num2) {
        if (num2 != 0) {
            return (double) num1 / num2;
        } else {
            System.out.println("Division by zero is not allowed.");
            return 0;
        }
    }

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter the first number: ");
        int num1 = scanner.nextInt();

        System.out.print("Enter the second number: ");
        int num2 = scanner.nextInt();

        // Calling the methods
        System.out.println("Addition: " + add(num1, num2));
        System.out.println("Subtraction: " + subtract(num1, num2));
        System.out.println("Multiplication: " + multiply(num1, num2));
```

```
        System.out.println("Division: " + divide(num1, num2));

        scanner.close();
    }
}
```

Explanation:

- Four methods (`add`, `subtract`, `multiply`, `divide`) are defined.
- Each method takes two integers as arguments and performs the corresponding operation.
- The `main` method prompts the user for two numbers, calls each method, and prints the result.
- For division, a check is done to ensure that division by zero is handled gracefully.

Problem 2: Find the Maximum of Three Numbers

Problem Statement:
Write a method `findMax` that takes three integers as parameters and returns the maximum of the three numbers. Call this method in the `main` function and print the result.

Solution:

```
java
Copy code
public class MaxOfThree {

    // Method to find the maximum of three numbers
    public static int findMax(int num1, int num2, int num3) {
        int max = num1;

        if (num2 > max) {
            max = num2;
        }

        if (num3 > max) {
            max = num3;
        }

        return max;
    }

    public static void main(String[] args) {
        int result = findMax(10, 15, 12);
        System.out.println("Maximum number is: " + result);
    }
}
```

Problem 3: Method Overloading for Addition

Problem Statement:
Write a program that uses method overloading for addition. Create two methods with the same name, `add`, but one should accept two arguments and the other should accept three arguments.

Solution:

```
java
Copy code
public class AdditionOverloading {

    // Method to add two numbers
    public static int add(int num1, int num2) {
        return num1 + num2;
    }

    // Overloaded method to add three numbers
    public static int add(int num1, int num2, int num3) {
        return num1 + num2 + num3;
    }

    public static void main(String[] args) {
        System.out.println("Sum of two numbers: " + add(5, 10));
        System.out.println("Sum of three numbers: " + add(5, 10, 15));
    }
}
```

Problem 4: Reverse a String Using a Method

Problem Statement:
Write a Java method that takes a string as input and returns the reversed string. Call this method in the `main` function and print the reversed string.

Solution:

```
java
Copy code
public class ReverseString {

    // Method to reverse a string
    public static String reverse(String str) {
        String reversed = "";
        for (int i = str.length() - 1; i >= 0; i--) {
            reversed += str.charAt(i);
        }
        return reversed;
    }

    public static void main(String[] args) {
        String result = reverse("hello");
        System.out.println("Reversed string: " + result);
    }
}
```

Problem 5: Calculate the Square of a Number

Problem Statement:
Write a method `square` that takes an integer as input and returns its square. Call this method from the `main` method and print the result.

Solution:

```
java
Copy code
public class SquareNumber {

    // Method to calculate the square of a number
    public static int square(int num) {
        return num * num;
    }

    public static void main(String[] args) {
        int result = square(6);
        System.out.println("Square of 6 is: " + result);
    }
}
```

Problem 6: Convert Celsius to Fahrenheit

Problem Statement:
Write a method `celsiusToFahrenheit` that takes a temperature in Celsius and returns the temperature in Fahrenheit. The formula is:
`Fahrenheit = (Celsius * 9/5) + 32.`

Solution:

```
java
Copy code
public class TemperatureConversion {

    // Method to convert Celsius to Fahrenheit
    public static double celsiusToFahrenheit(double celsius) {
        return (celsius * 9 / 5) + 32;
    }

    public static void main(String[] args) {
        double fahrenheit = celsiusToFahrenheit(25);
        System.out.println("25 degrees Celsius is: " + fahrenheit + "
Fahrenheit");
    }
}
```

Problem 7: Check Even or Odd

Problem Statement:
Write a method `isEven` that checks if a number is even. It should return `true` if the number is even and `false` if it is odd.

Solution:

```
java
Copy code
public class EvenOddCheck {

    // Method to check if a number is even
    public static boolean isEven(int num) {
        return num % 2 == 0;
    }

    public static void main(String[] args) {
        System.out.println("Is 4 even? " + isEven(4));
        System.out.println("Is 7 even? " + isEven(7));
    }
}
```

Problem 8: Convert Minutes to Hours and Minutes

Problem Statement:
Write a method `convertToHoursAndMinutes` that takes the number of minutes as input and prints how many hours and minutes it corresponds to.

Solution:

```
java
Copy code
public class TimeConversion {

    // Method to convert minutes into hours and minutes
    public static void convertToHoursAndMinutes(int minutes) {
        int hours = minutes / 60;
        int remainingMinutes = minutes % 60;
        System.out.println(minutes + " minutes is equal to " + hours + "
hours and " + remainingMinutes + " minutes.");
    }

    public static void main(String[] args) {
        convertToHoursAndMinutes(130);
    }
}
```

Problem 9: Swap Two Numbers Using a Method

Problem Statement:
Write a method `swapNumbers` that takes two integers as input and swaps their values. Display the values before and after swapping.

Solution:

```
java
Copy code
public class SwapNumbers {

    // Method to swap two numbers
    public static void swapNumbers(int a, int b) {
        System.out.println("Before swapping: a = " + a + ", b = " + b);
        int temp = a;
        a = b;
        b = temp;
        System.out.println("After swapping: a = " + a + ", b = " + b);
    }

    public static void main(String[] args) {
        swapNumbers(5, 10);
    }
}
```