Assignment 4 - Switch Case and Ternary

This assignment consists of two problems designed to practice your understanding of **Switch Case** and **Ternary Operator**. Complete the following problems and submit your solutions for review.

Problem 1: Simple Calculator (Switch Case)

Objective: Write a program that acts as a simple calculator. The program should read two numbers and an operator (`+`, `-`, `*`, `/`) from the user and perform the corresponding operation using a `switch` statement. If the operator is not valid, print "Invalid operator."

Requirements:

- Use a `switch` statement to handle different cases for each arithmetic operation (`+`, ``, ``, `/`).

- Handle division by zero gracefully and print an appropriate error message.

- Print an error message if the operator is not recognized.

Example Output:

```
Input first number: 8
Input second number: 4
Input operator (+, -, *, /): *
Output: Result: 32.0
```

Example Output for Invalid Operator:

```
Input first number: 5
Input second number: 3
Input operator (+, -, *, /): ^
Output: Invalid operator.
```

Example Output for Division by Zero:

```
Input first number: 10
Input second number: 0
Input operator (+, -, *, /): /
Output: Error: Division by zero is not allowed.
```

Problem 2: Find the Greater Number (Ternary Operator)

Objective: Write a program that reads two integers from the user and uses the **ternary operator** to find and print the greater of the two numbers. Print a message if both numbers are equal.

Requirements:

- Use the ternary operator to compare the two numbers.

- Print the greater number along with a message, or state that both numbers are equal.

Example Output:

```
Input first number: 15
Input second number: 20
Output: The greater number is 20.
```

Example Output for Equal Numbers:

```
Input first number: 10
Input second number: 10
Output: Both numbers are equal.
```

Instructions:

1. Write the complete Java code for each problem in separate files or methods.

2. Test your programs with various input cases to ensure correctness.

3. Add comments in your code to explain your logic where necessary.

Good luck, and happy coding!