System Requirements Specification Index

For

Increment & Decrement Operators

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

Table of Contents

1 Project Abstract

Create a Java program that demonstrates the use of increment (++) and decrement (--) operators. Show the difference between pre-increment and post-increment, as well as pre-decrement and post-decrement.

2 Assessment Tasks

1. **Declare 2 variables:**
   - A variable named `count` of `int` datatype, initialized with the value `5`.

2. **Perform Increment and Decrement Operations:**
   Use the declared variable `count` to perform the following operations and print the results:

   - **Pre-increment (`++count`)**:
     1) Increment the value of `count` by `1` **before** printing.
     2) Print the result with the message: `"Pre-increment:"`.
   - **Post-increment (`count++`)**:
     1) Print the value of `count` **first**, then increment it by `1`.
     2) Print the result with the message: `"Post-increment:"`.
   - **Pre-decrement (`--count`)**:
     1) Decrement the value of `count` by `1` **before** printing.
     2) Print the result with the message: `"Pre-decrement:"`.
   - **Post-decrement (`count--`)**:
     1) Print the value of `count` **first**, then decrement it by `1`.
     2) Print the result with the message: `"Post-decrement:"`.

Print the Results:

- Print the results of each operation in separate lines as shown in the expected output.

Expected Output:

```
Pre-increment: 6

Post-increment: 6

Pre-decrement: 6

Post-decrement: 6
```

3 Template Code Structure

3.1 Package: com.yaksha.assignment.IncrementDecrementOperatorsAssignment

Resources

Class/Interface	Description	Status
IncrementDecrementOperatorsAssignment (class)	● Main class containing the logic to perform and demonstrate increment (++) and decrement (--) operations.	Need to be implemented.

4 Execution Steps to Follow

1. **All actions like build, compile, running application, running test cases will be through Command Terminal.**
2. **To open the command terminal the test takers, need to go to Application menu (Three horizontal lines at left top) Terminal New Terminal.**
3. **This editor Auto Saves the code.**
4. **If you want to exit(logout) and continue the coding later anytime (using Save & Exit option on Assessment Landing Page) then you need to use CTRL+Shift+B-command compulsorily on code IDE. This will push or save the updated contents in the internal git/repository. Else the code will not be available in the next login.**
5. **These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.**
6. **To run your project use command:**

   ```
   mvn compile exec:java
   -Dexec.mainClass="com.yaksha.assignment.IncrementDecrementOperatorsAssignment"
   ```
7. **To test your project test cases, use the command**

   ```
   mvn test
   ```

8. **You need to use CTRL+Shift+B - command compulsorily on code IDE, before final submission as well. This will push or save the updated contents in the internal git/repository, and will be used to evaluate the code quality.**