
System Requirements Specification Index

For

Type Casting Operations

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

TABLE OF CONTENTS

1	Project Abstract	3
2	Assessment Tasks	3
3	Template Code Structure	4
3.1	Package: com.yaksha.assignment.TypeCastingAssignment	4
4	Execution Steps to Follow	4

USE CASE DESCRIPTION

System Requirements Specification

1 PROJECT ABSTRACT

Type Casting Operations is a console-based Java application that demonstrates the usage of both implicit and explicit type casting. The program showcases how primitive types like `int` and `double` can be converted into one another, emphasizing the difference between automatic (implicit) and manual (explicit) type casting.

2 ASSESSMENT TASKS

1. Declare 2 variables:

- A variable named `intValue` of `int` datatype, initialized with the value `10`.
- A variable named `doubleValue` of `double` datatype, initialized with the value `9.5`.

2. Perform Type Casting:

- Implicitly cast the `intValue` (of type `int`) into a `double` variable named `implicitCast`.
- Explicitly cast the `doubleValue` (of type `double`) into an `int` variable named `explicitCast` using `(int)`.

3. Print the Results:

- Print the results i.e, `implicitCast` and `explicitCast` with appropriate labels in separate lines as shown in the expected output.

4. Expected Output:

Implicit casting (int to double): 10.0

Explicit casting (double to int): 9

3 TEMPLATE CODE STRUCTURE

3.1 PACKAGE: COM.YAKSHA.ASSIGNMENT.TypeCastingAssignment

Resources

Class/Interface	Description	Status
TypeCastingAssignment (class)	<ul style="list-style-type: none">Main class containing the logic to perform implicit and explicit type casting 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. 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.
5. To run your project use command:
sudo JAVA_HOME=\$JAVA_HOME /usr/share/maven/bin/mvn compile exec:java
-Dexec.mainClass="com.yaksha.assignment.TypeCastingAssignment"
***If it asks for the password, provide password : pass@word1**
6. To test your project test cases, use the command
sudo JAVA_HOME=\$JAVA_HOME /usr/share/maven/bin/mvn test
***If it asks for the password, provide password : pass@word1**