
System Requirements Specification Index

For

Math Functions

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

TABLE OF CONTENTS

1	Project Abstract	3
2	Assessment Tasks	3
3	Template Code Structure	5
3.1	Package: com.yaksha.assignment.MathFunctionsAssignment	5
4	Execution Steps to Follow	5

USE CASE DESCRIPTION

System Requirements Specification

1 PROJECT ABSTRACT

This assessment focuses on evaluating the understanding and ability to work with basic math functions and operations using Java. You need to utilize the Math class to perform operations like rounding numbers, finding square roots, performing exponentiation, and using trigonometric and logarithmic functions.

2 ASSESSMENT TASKS

Task 1: Compute Absolute Value:

- Declare a variable named `num` of `integer` datatype, initialized with the value `-10`.
- Use `Math.abs(num)` to get the absolute value of `num`.
- **Print** the result: `"Absolute Value:"` followed by the absolute value.

Task 2: Find Maximum and Minimum Values:

- Declare a variable named `num1` of `integer` datatype, initialized with the value `25`.
- Declare a variable named `num2` of `integer` datatype, initialized with the value `10`.
- Use `Math.max(num1, num2)` to determine the maximum value.
- Use `Math.min(num1, num2)` to determine the minimum value.
- **Print** both results:
`"Max:"` followed by the maximum value.
`"Min:"` followed by the minimum value.

Task 3: Calculate Square Root:

- Declare a variable named `sqrNum` of `double` datatype, initialized with the value `25.0`.
- Use `Math.sqrt(sqrNum)` to compute the square root.
- **Print** the result: `"Square Root:"` followed by the square root.

Task 4: Perform Exponentiation:

- Declare a variable named `powerResult` of `double` datatype, Compute `2^3` using `Math.pow(2, 3)`. (or)
- Compute `2^3` using `Math.pow(2, 3)`.
- Store the result in a variable named `powerResult` of `double` datatype.
-

- **Print** the result: "2^3:" followed by the value of `powerResult`.

Task 5: Compute Trigonometric Functions:

- Declare a variable named `angleInRadians` of `double` datatype, Convert 45 degrees to radians using `Math.toRadians(45)`.
- Compute the sine value of 45 degrees using `Math.sin(angleInRadians)`.
(or)
- Convert 45 degrees to radians using `Math.toRadians(45)`.
- Compute the sine value of 45 degrees using `Math.sin(angleInRadians)`.
- Store the result in a variable named `angleInRadians` of `double` datatype.
-
- **Print** the result: "Sine(45 degrees):" followed by the sine value.

Task 6: Perform Logarithmic Operations:

- Declare a variable named `logResult` of `double` datatype, Compute the natural logarithm of 10 using `Math.log(10)`.
- (or)
- Compute the natural logarithm of 10 using `Math.log(10)`.
- Store the result in a variable named `logResult` of `double` datatype.
-
- **Print** the result: "Natural Log of 10:" followed by the value of `logResult`.

Expected Output:

```
Absolute Value: 10
Max: 25
Min: 10
Square Root: 5.0
2^3: 8.0
Sine(45 degrees): 0.7071067811865475
Natural Log of 10: 2.302585092994046
```

3 TEMPLATE CODE STRUCTURE

3.1 PACKAGE: COM.YAKSHA.ASSIGNMENT.MATHFUNCTIONSASSIGNMENT

Resources

Class/Interface	Description	Status
MathFunctionsAssignme nt (class)	<ul style="list-style-type: none"> • Main class demonstrating the use of mathematical functions 	Need to be implemented.

	such as: absolute value, max, min, square root, exponentiation, trigonometric functions, and logarithmic operations using the Math class.	
--	--	--

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.MathFunctionsAssignment"
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.