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 `sqrtNum` of `double` datatype, initialized with the value `25.0`.
- Use `Math.sqrt(sqrtNum)` 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)` and assign to `angleInRadians` .
- Compute the sine value of `45` degrees using `Math.sin(angleInRadians)`.
- **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)` and assign to `logResult`.
- **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
MathFunctionsAssignment (class)	• Main class demonstrating the use of mathematical functions such as: absolute value, max, min, square root, exponentiation, trigonometric functions, and logarithmic operations using the `Math` class.	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.MathFunctionsAssignment"

   ***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**