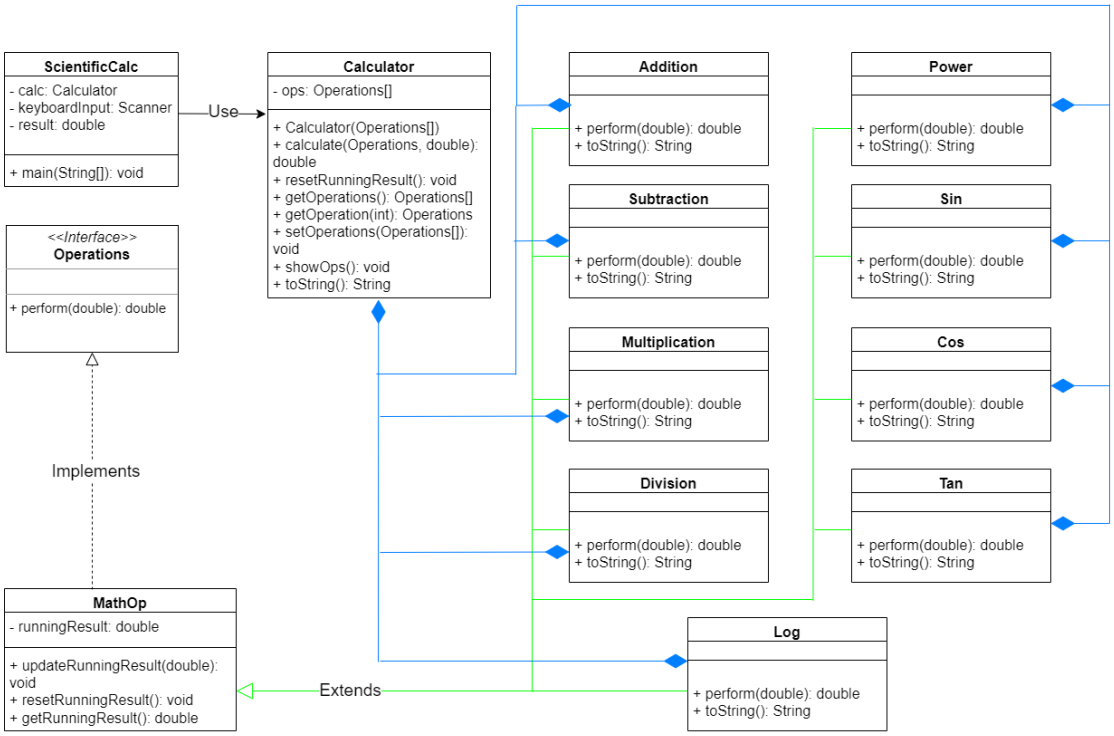
Project 2
CS 501 - 001
Kartikeya Arvind Yadav (kay54)

Scientific Calculator

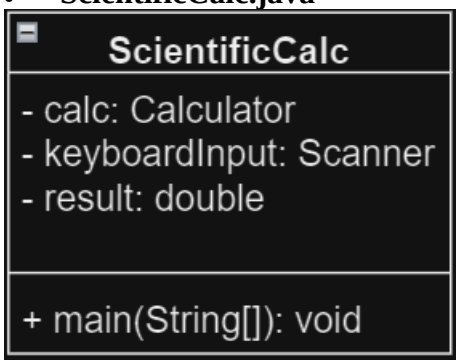
1. UML Diagram of the project

- The green line indicates the extends relationship.
- The blue line indicated the composition relationship.

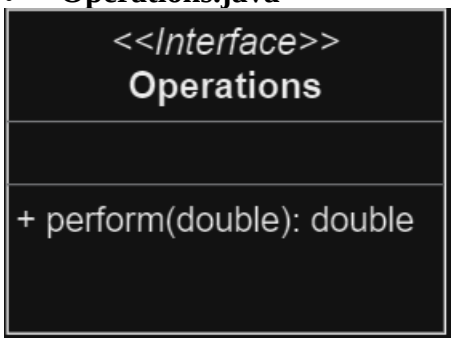


2. UML diagram of each class

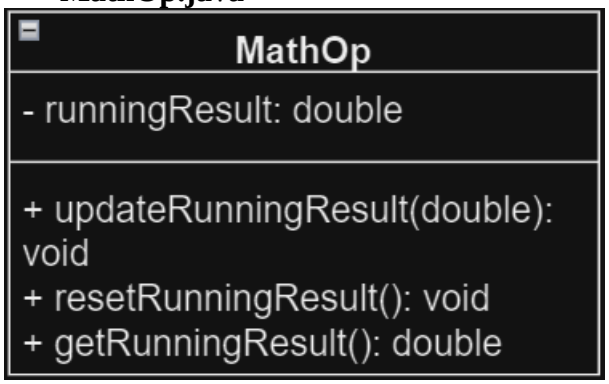
- **ScientificCalc.java**



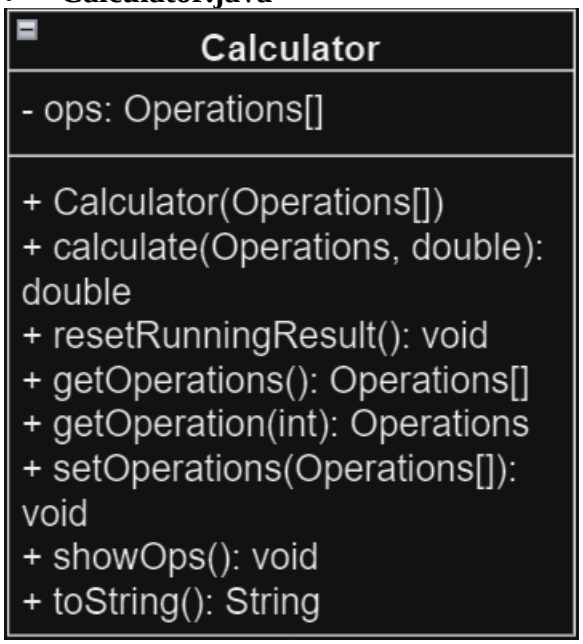
- **Operations.java**



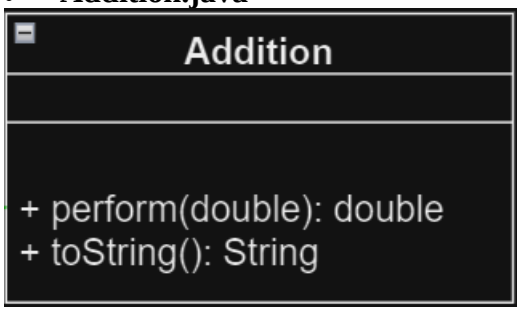
- **MathOp.java**



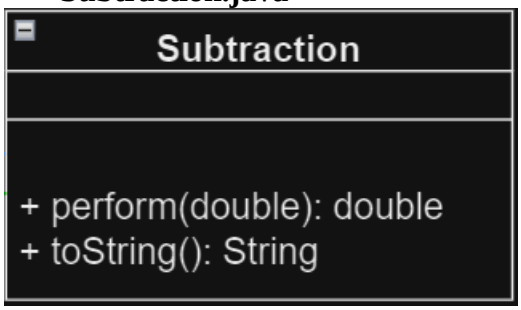
- **Calculator.java**



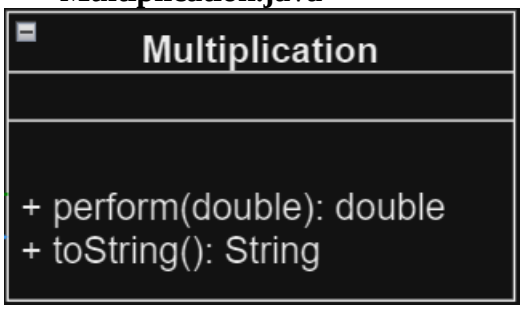
- **Addition.java**



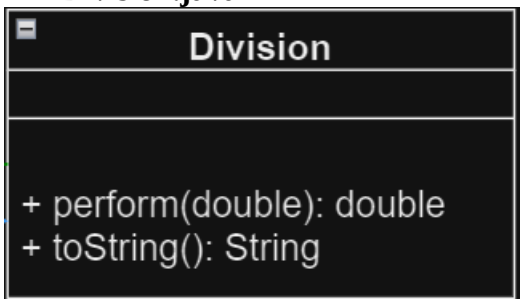
- **Subtraction.java**



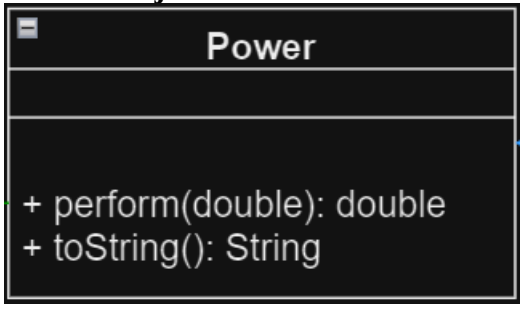
- **Multiplication.java**



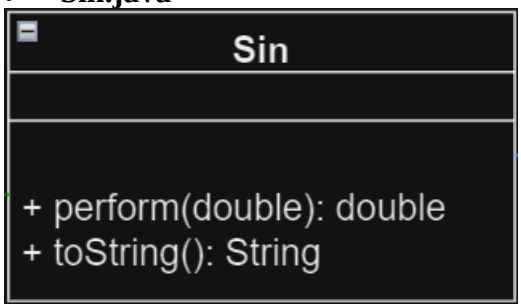
- **Division.java**



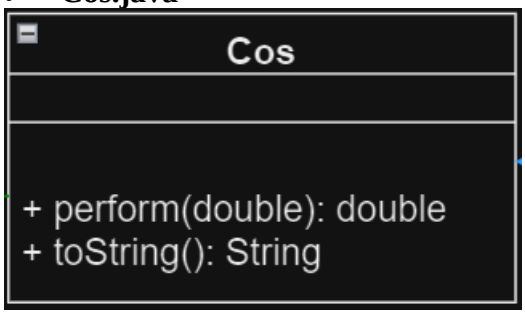
- **Power.java**



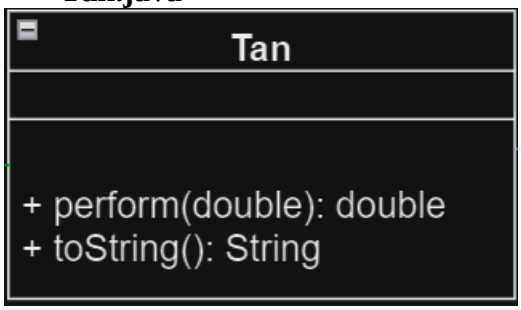
- **Sin.java**



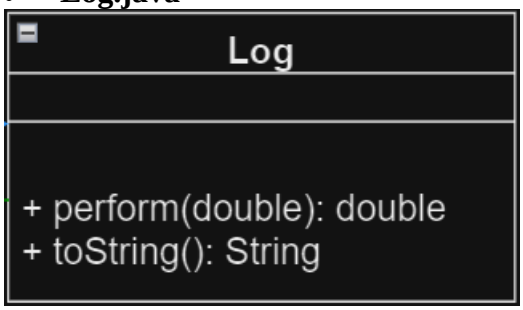
- **Cos.java**



- **Tan.java**



- **Log.java**



3. IS-A and HAS-A relationship in this project

IS-A relationship (Inheritance)

- The **MathOp** is an abstract class that serves as a common base for specific mathematical operations.
- **MathOp** extends **Operations**, indicating that it **IS-A** type of mathematical operation.
- The subclasses **Addition**, **Subtraction**, **Multiplication**, **Division**, **Power**, **Sin**, **Cos**, **Tan**, and **Log** inherit from **MathOp** and, by extension, implement the **Operations** interface.

- All these sub-classes (**Addition**, **Subtraction**, **Multiplication**, **Division**, **Power**, **Sin**, **Cos**, **Tan**, and **Log**) have an **IS-A** relationship with **MathOp**.
- We can say that **Addition (Subtraction**, **Multiplication**, **Division**, **Power**, **Sin**, **Cos**, **Tan**, and **Log**) IS-A **MathOp**.

HAS-A relationship (Composition)

- The **Calculator HAS-A** composition relationship with an array of **Operations**.
- It contains and manages instances of different mathematical operations.
- **Calculator** has methods to get, set, and perform operations on the array of **Operations**.
- **ScientificCalc HAS- AN** instance of **Calculator** (calc). It contains and utilizes the functionalities provided by the **Calculator** class.