

Task 1: Write a set of JUnit tests for a given class with simple mathematical operations (add, subtract, multiply, divide) using the basic @Test annotation.

```
public class MathematicalOperations {

    public int add(int a, int b) {

        return a+b;

    }

    public int subtract(int a, int b) {

        return a-b;

    }

    public int multiply(int a, int b) {

        return a*b;

    }

    public int divide(int a, int b) {

        if (b==0) {

            throw new IllegalArgumentException("Division by zero is undefined.");

        }

        return a/b;

    }

}

import org.junit.Test;

import static org.junit.Assert.*;

public class MathematicalOperationsTest{

    private MathematicalOperations mathOps = new MathematicalOperations();

    @Test

    public void testAdd() {

        assertEquals(5, mathOps.add(2, 3));

        assertEquals(-1, mathOps.add(-2, 1));

        assertEquals(0, mathOps.add(0, 0));

    }

}
```

```
@Test
public void testSubtract() {
    assertEquals(1, mathOps.subtract(3,2));
    assertEquals(-3, mathOps.subtract(0,3));
    assertEquals(5, mathOps.subtract(10,5));
}

@Test
public void testMultiply() {
    assertEquals(6, mathOps.multiply(2, 3));
    assertEquals(0, mathOps.multiply(0, 3));
    assertEquals(-15, mathOps.multiply(3, -5));
}

@Test
public void testDivide(){
    assertEquals(2, mathOps.divide(6, 3));
    assertEquals(-2, mathOps.divide(10, -5));
}

@Test(expected = IllegalArgumentException.class)
public void testDivideByZero() {
    mathOps.divide(5, 0);
}
}
```