**package com.niit;**

**public** **class** Calculation {

    //method that returns maximum number

**public** **static** **int** findMax(**int** arr[]){

**int** max=0;

**for**(**int** i=1;i<arr.length;i++){

**if**(max<arr[i])

                max=arr[i];

        }

**return** max;

    }

    //method that returns cube of the given number

**public** **static** **int** cube(**int** n){

**return** n\*n\*n;

    }

    //method that returns reverse words

**public** **static** String reverseWord(String str){

        StringBuilder result=**new** StringBuilder();

        StringTokenizer tokenizer=**new** StringTokenizer(str," ");

**while**(tokenizer.hasMoreTokens()){

        StringBuilder sb=**new** StringBuilder();

        sb.append(tokenizer.nextToken());

        sb.reverse();

        result.append(sb);

        result.append(" ");

        }

**return** result.toString();

    }

}

**import** **static** org.junit.Assert.assertEquals;

**import** org.junit.After;

**import** org.junit.AfterClass;

**import** org.junit.Before;

**import** org.junit.BeforeClass;

**import** org.junit.Test;

**import** com.niit.\*;

**public** **class** TestCase2 {

    @BeforeClass

**public** **static** **void** setUpBeforeClass() **throws** Exception {

        System.out.println("before class");

    }

    @Before

**public** **void** setUp() **throws** Exception {

        System.out.println("before");

    }

    @Test

**public** **void** testFindMax(){

        System.out.println("test case find max");

        assertEquals(4,Calculation.findMax(**new** **int**[]{1,3,4,2}));

        assertEquals(-2,Calculation.findMax(**new** **int**[]{-12,-3,-4,-2}));

    }

    @Test

**public** **void** testCube(){

        System.out.println("test case cube");

        assertEquals(27,Calculation.cube(3));

    }

    @Test

**public** **void** testReverseWord(){

        System.out.println("test case reverse word");

        assertEquals("ym eman si nahk",Calculation.reverseWord("my name is khan");

    }

    @After

**public** **void** tearDown() **throws** Exception {

        System.out.println("after");

    }

    @AfterClass

**public** **static** **void** tearDownAfterClass() **throws** Exception {

        System.out.println("after class");

    }

}