JAVA PROGRAMMING COURSE (SWE2023)
FALL SEMESTER 2022

INSTRUCTOR: Prof. TAMER ABUHMED
COLLEGE OF SOFTWARE

Assignment 2

This assignment consists of 3 tasks. Guidelines for submission format are given at the end of the assignment file.

Note: The green numbers and green words in the console are the user's inputs (Used IDEA: IntelliJ IDEA 2021.1).

Task 1

(Employee Class) Create a class called Employee that includes four instance variables: a name (string), salary (double), tax (double), and insurance (double). Provide a constructor that initializes the two instance variables: name and salary. Provide a set and a get method for **all** instance variables. First, when the user runs the program, the program should give information about employees: their name, salary, tax, and insurance.

- Initialize two employee objects as shown in the example below
- Calculate tax (assume that tax is **8.8%** of salary) and set the variable in the constructor.
- Calculate insurance (assume that insurance is **6.5%** of salary) and the variable in the constructor.
- Show information about each employee

- Ask the user to enter the percentage for salary increase.
- Calculate the new salary and update the salary, tax, and insurance for each employee
- Show information about each employee after the salary increase

Main.java Employee.java

```java
package com.company;

import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        double salaryIncreasePercentage = 0; // variable for storing increasing percentage
        Scanner scanner = new Scanner(System.in); // creating Scanner object for getting user inputs

        Employee firstEmployee = new Employee( name: "John",  salary: 600000); // First employee object
        Employee secondEmployee = new Employee( name: "Mark",  salary: 900000); // Second employee object

        // getInformation() method is for getting information about the employee
        firstEmployee.getInformation(); // getting info about first employee: name, salary, tax amount and insurance amount
        secondEmployee.getInformation(); // getting info about second employee: name, salary, tax amount and insurance amount

        System.out.println("==================================");
        System.out.print("Increase salary (in percentage): ");
        salaryIncreasePercentage = scanner.nextDouble(); // Getting user input for percentage of increasing salary

        // increaseSalary() method increases the salary by the given percentage and update tax and insurance amount
        firstEmployee.increaseSalary(salaryIncreasePercentage); // increasing salary for first employee
        secondEmployee.increaseSalary(salaryIncreasePercentage); // increasing salary for second employee

        System.out.println("After Salary Increase");
        firstEmployee.getInformation(); // getting info about first employee after salary increase
        secondEmployee.getInformation(); // getting info about second employee after salary increase
    }
}
```

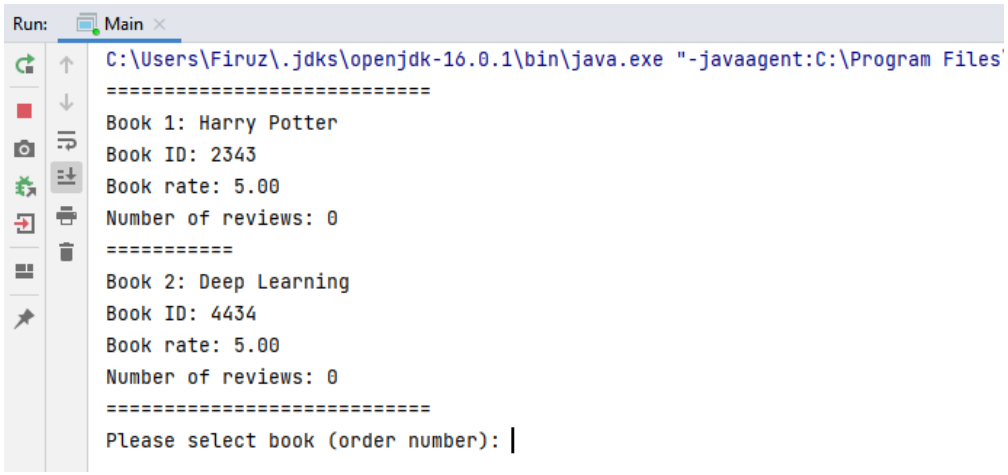
Expected output:

Task 2

(Book Review) The SKKU library has many books, so it needs a book review program for recommending them to students. To get student reviews for the SKKU library, you need to develop a Java program.

Your program should work as follow:

1. The program prints the list of books with information and order number (Book 1, Book 2, etc). Then, the program asks the student to enter the book order number to select a book to give a review.

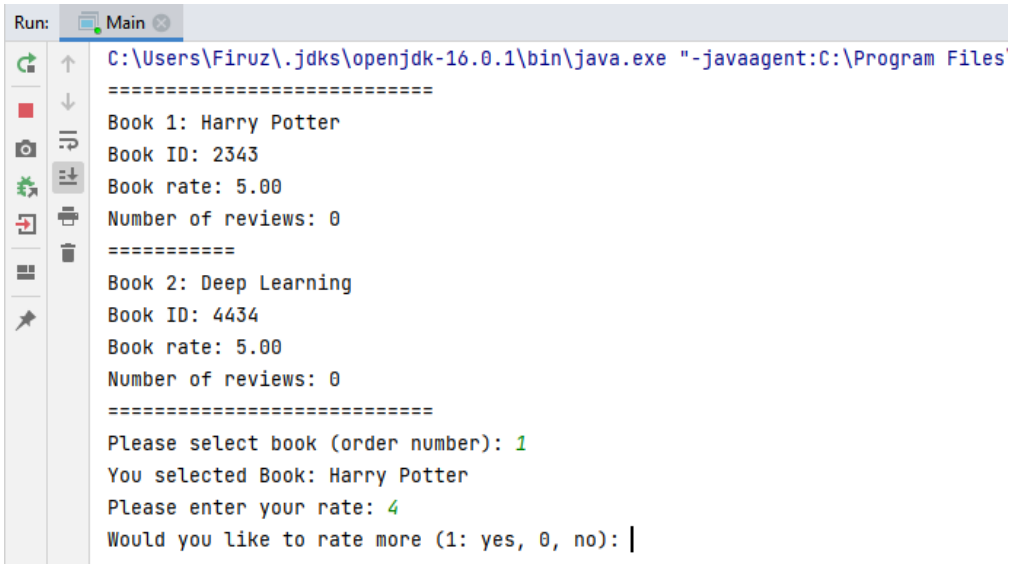


Note: by default, the book rate is 5.0 (with zero reviews).

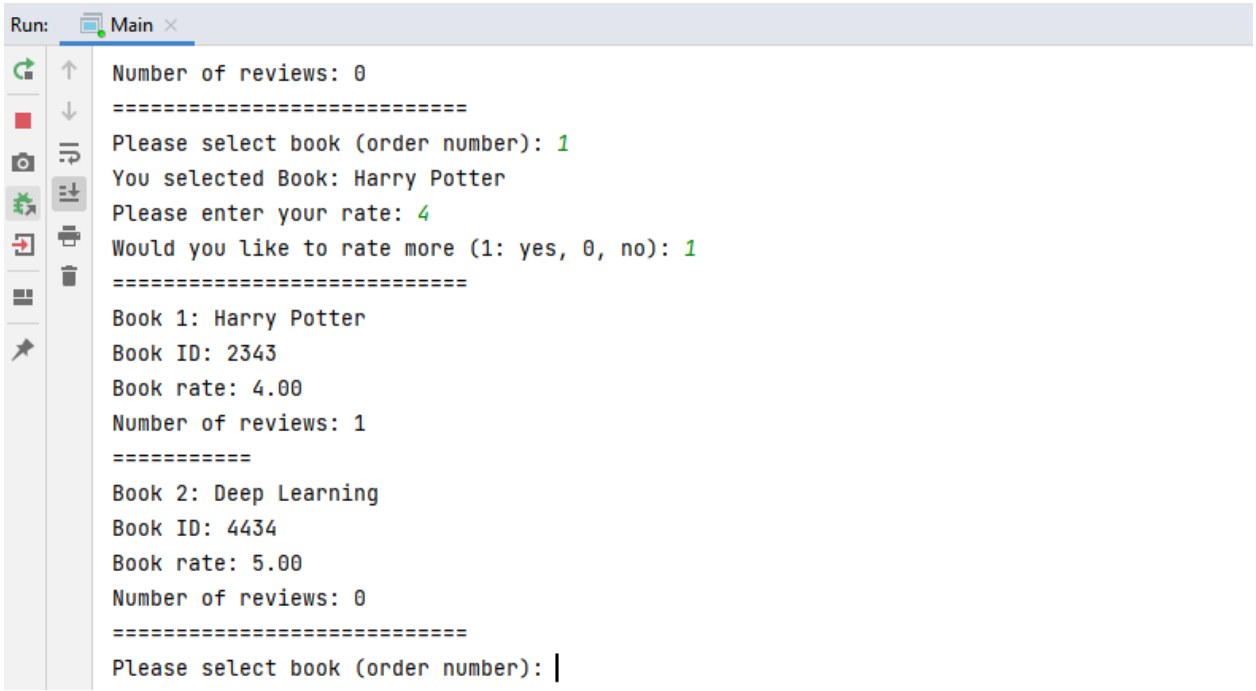
Note: For this assignment, the number of books is only 2, you do not need to use Array List. Just create 2 book objects and work on them:

```
Book firstBook = new Book( bookID: 2343,  bookTitle: "Harry Potter",  reviewCount: 0);
Book secondBook = new Book( bookID: 4434,  bookTitle: "Deep Learning",  reviewCount: 0);
```

2. After the student enters the book order number (in our example, it is 1), the program confirms the selection by printing the book name (ex: "You selected Book: Harry Potter") and asks the student to enter the rate (max: 5). After the student enters the rate (ex: 4), the program asks "Would you like to rate more (1: yes, 0: no)".



3. If the student enters 1 and wants to enter more reviews, the program prints the list of books again:



Note: you can see that this time the reviews are updated. For Book 1, in the beginning, the rate was 5.0 by default. After entering rate 4 for this book, now the book rate is 4.0 (number of reviews: 1).

4. Let's select book 1 and this time let's give a rate of 5 and let's see how it is changed:

```
Please select book (order number): 1
You selected Book: Harry Potter
Please enter your rate: 5
Would you like to rate more (1: yes, 0, no): 1
============================
Book 1: Harry Potter
Book ID: 2343
Book rate: 4.50
Number of reviews: 2
==========
Book 2: Deep Learning
Book ID: 4434
Book rate: 5.00
Number of reviews: 0
```

Note: We can see that the average rate of the book "Harry Potter" is 4.5 now with 2 reviews.

5. When the student enters 0 when the program asks "Would you like to rate more (1: yes, 0: no)", the program will finish:

```
Would you like to rate more (1: yes, 0, no): 0

Process finished with exit code 0
```

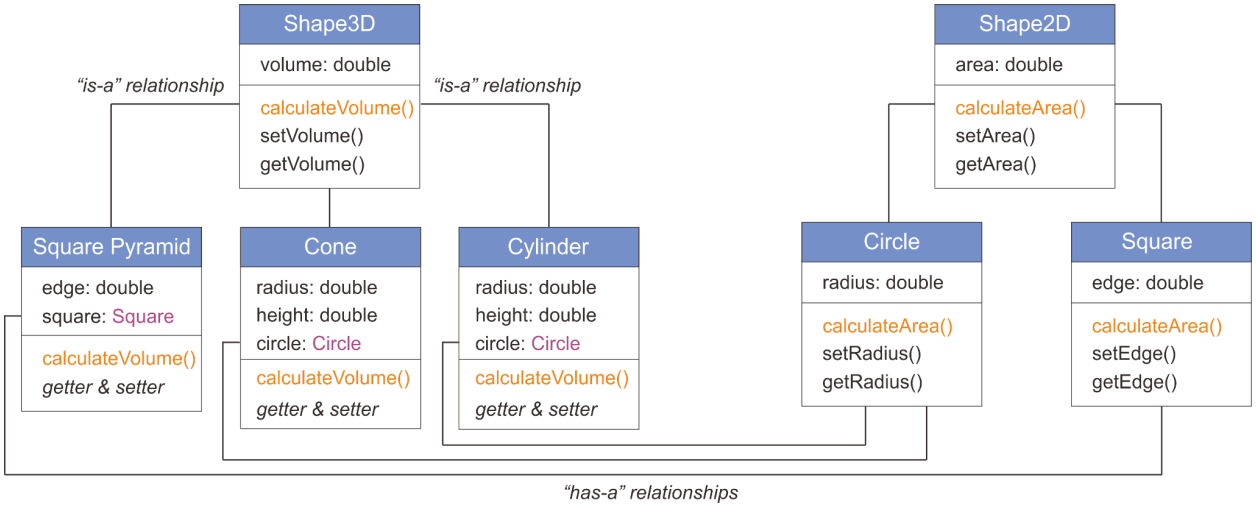
Create a Book class as follow:

Book
bookID bookTitle reviewCount averageRate
updateRate() updateReviewCount() *+ getter & setter*

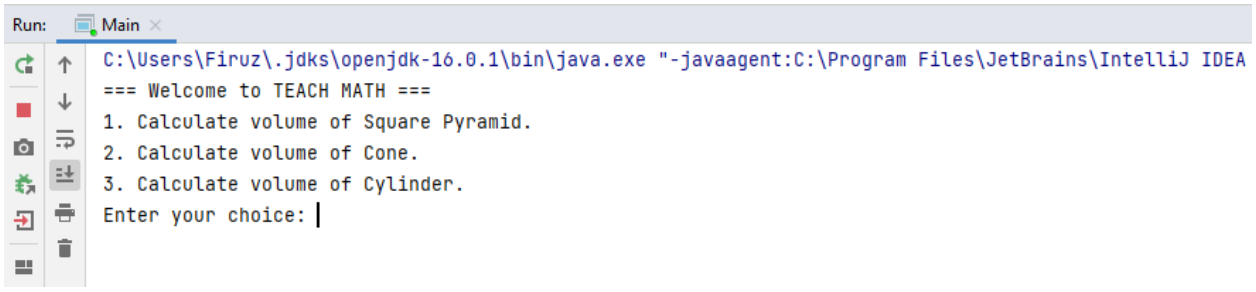
Note: We will check with expected inputs, handling exceptions is totally optional.

Task 3

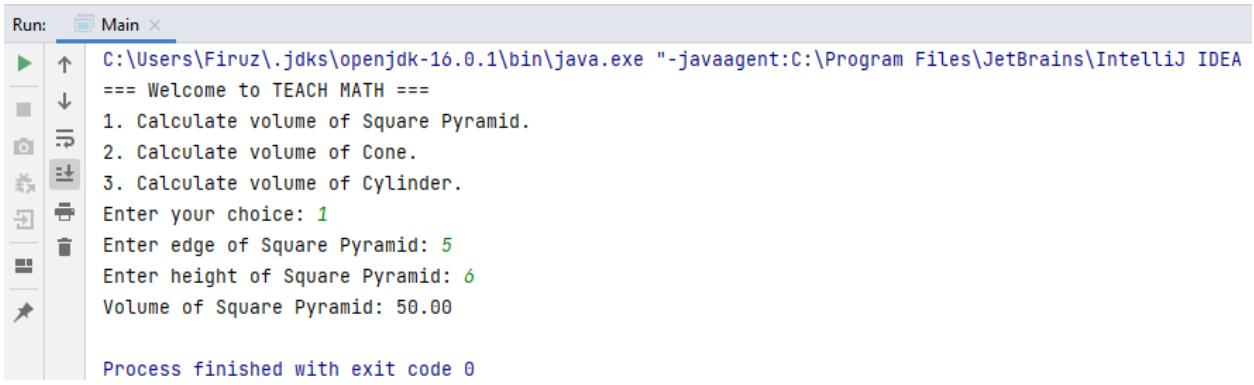
(Teach Math) Create a Java program to help children to learn some geometry. Create the following classes: Shape2D Circle, Square, Shape3D, SqaurePyramid, Cone, and Cylinder. Provide constructors for each class that initializes the initial instances except for area and volume. Provide a set and a get method for **all** instance variables. Follow the following diagram:



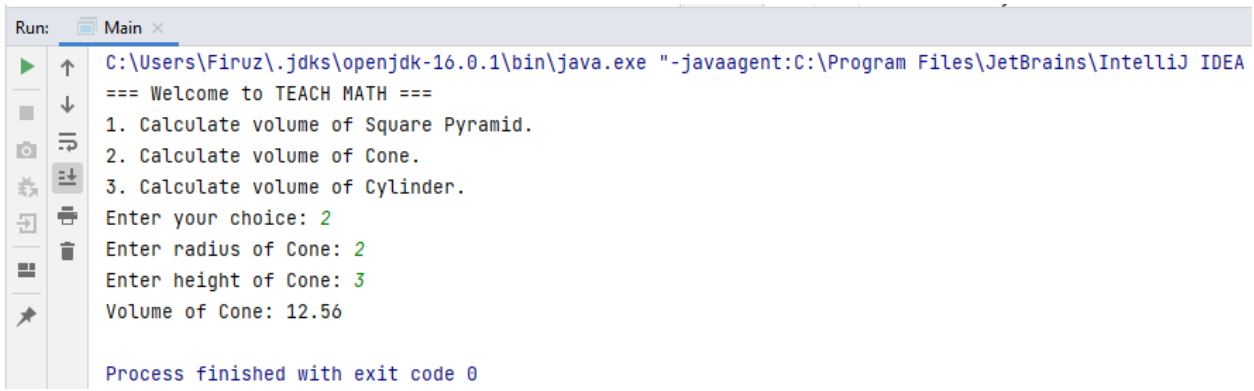
First, provide a menu that asks the user's input to calculate which type of geometric figure.



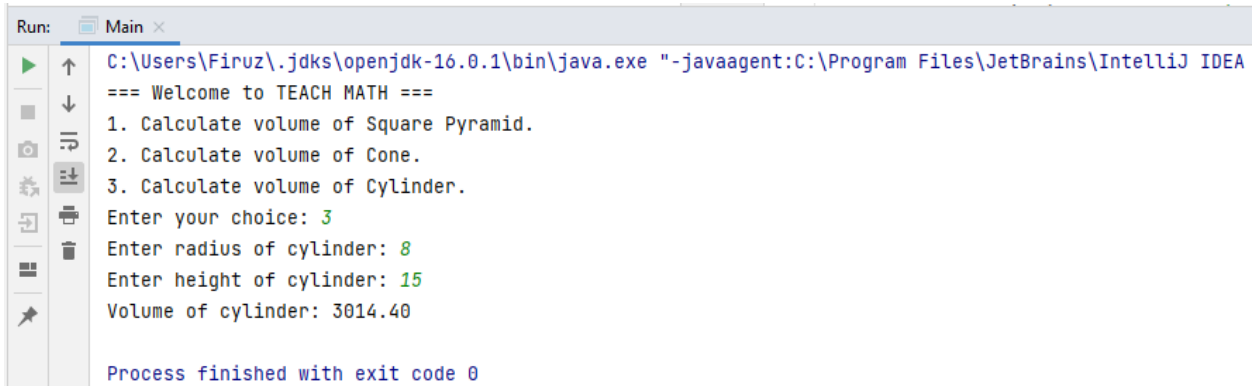
If the user selects the first option (1), ask edge and height of the square pyramid, then print the volume of the square pyramid and the program ends:



If the user selects the second option (2), ask for the radius and height of the Cone, then print the volume of the cone and end the program:

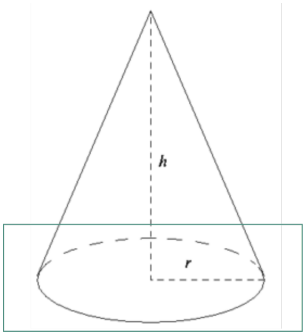


If the user selects the first option (3), ask height and radius of the cylinder, then print the volume of the cylinder and end the program:



***Hints**:*

- For calculating the volume of the cone, first, calculate the area of a circle and multiply it by (height/3) (figure below):

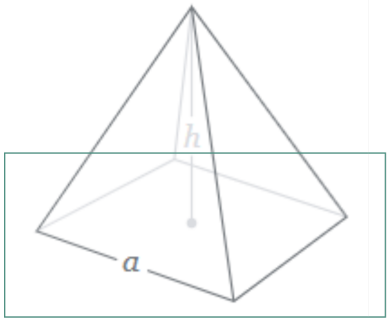


Area of Circle = $\pi * radius^2$

Volume of Cone:

Volume = (circle area) $* \frac{height}{3}$

- Same with Square Pyramid and Cylinder. For Square Pyramid use the area of the square and multiply it by height;



a = edge

Area of Square = a^2

Volume of Square Pyramid:

Volume = (square area) $* \frac{height}{3}$

Grading:

- Correctness of code (should be run without errors)
- Comments (for important lines)
- Coding style (meaningful naming variables and methods)
- Number of classes (as asked in the task description)
- Usage of Inheritance and Composition

Submission format: Submit **twelve separate files (only .java files, not the whole project folder)**. Two .java files for task 1: Main.java and Employee.java, two for task 2: Main.java and Book.java, and eight .java files for task 3: Main.java, Shape3D.java, Cone.java, Cylinder.java, SquarePyramid.java, Shape2D.java, Circle.java, and Square.java. Files must include the implementation code of each task and comments for important lines of code to explain the purpose. All the files should be submitted as a **zip** file.

Name of zip file: {student ID}_{Student name}_assignment2.zip

Example: 2020712837_Frank_Thomas_assignment2.zip

Important: Plagiarism is strictly prohibited. If there is any plagiarism found in the code, you will be given an "F" for the assignment evaluation.

If you have any questions about the assignment, you can ask them in the discussion section of the week or contact the TAs directly.

Good luck!