WESTERN UNIVERSITY
DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING

ES1036B
PROGRAMMING FUNDAMENTALS FOR ENGINEERS
WINTER 2022

Lab03 – Methods (ET2+KB4)

Instructor:	Prepared by:	Acknowledgement:
DR. QUAZI RAHMAN	HIRA NADEEM	HAREEM NISAR

Lead TA: Hira Nadeem, hnadeem5@uwo.ca

---

1 Goal

This lab assignment will explore reading the console input with the Scanner class.

2 Resources

Lecture notes: Unit 2 Part 4: Identifiers, and Input operation
Unit 2 Part 5: Operators, Frequently used Methods, Casting and printf()
Unit 3 Part 1: Method Fundamentals

Refer to the IntelliJ video feed from 2021 "How to input data using Scanner objects (Under Unit 2)" and the one on "static methods (Under Unit 3)"
Refer to Lab 2 handout for a review of Scanner object.

3 Deliverables

One zip file containing your project folder. Name it username_Lab3.zip. where username is the beginning part of your email (e.g., Dr. Rahman's email address is qrahman3@uwo.ca, and his UWO username is: qrahman3).

Submission deadline: Submit your code by the end of your lab session. Prepare to demonstrate your understanding during the lab session.

4 Good Programming Practices

Below are several programming practices that should be followed to write and maintain quality codes.
Please note that you will be marked on all these components:

- Include meaningful comments in your program. This will help you remember what each part of your code does, especially after long breaks from your work. Your TA will also appreciate understanding your code by going over your comments.

- Choose meaningful and descriptive names for your variables. There is a balance between descriptive names and code readability but always err on the side of descriptive.

- It is recommended that you follow the naming strategy for variables, methods and class names, as outlined in your course handout (Unit 2). Since the identifiers cannot contain white-space characters (spaces or tabs), words in an identifier should be separated by uppercase letters (myNewFunction(), myNewVariable). For class names, capitalize the first letter of each word in the name (e.g. MyClass, MatrixCalculator). For any constant name, use uppercase letters, and if needed, concatenate two or more words with underscore (e.g., MINIMUM_DRIVING_AGE)

- Initialize variables when declaring them. This means giving them initial values which are easier to track in your program if logical errors are present with your output.

- Indent and properly format your code. You should write your codes so that your teaching assistant can read and understand your code easily.

- **Include a header in each of your java class files**. The header should include your full name, UWO ID number, date the code was written and a brief description of the program in that file. Use any print statement to print the Header information on the screen based on the format below:

```
/*****************************************
* Add your full name here*
* Add your student number*
* Add Date*
* Give a brief description of the task *
*****************************************/

public class MyClass
{
    public static void main(String args[])
    {
      // Your code here
    }
}
```

5 Lab Assignment Questions

Begin: Use IntelliJ IDE to create a project named ***username*_Lab03**

Question 1: Volume of a Cone (10 marks)

For this question, refer to the slides for "Unit 2.4 Identifiers, and Input operation" and the IntelliJ video from 2021 on OWL called "How to input data using Scanner objects" to get comfortable using the Scanner object. As well, refer to the slides for "Unit 2.5 Operators, Frequently used Methods, Casting and printf()" for using the printf() method and the Arithmetic Operators.

Write a program (according to the specifications below) that will ask the user to input the height and radius of a cone and will output the volume of the cone to the console.

Specifications:

- Create a package named **Q1** for your program.
- Give your java class an appropriate name.
- The program will print your commented header with a brief description of the task as shown in the sample output.
- The user must be asked to enter the **height** and **radius** for the cone they wish to compute the volume for.
- The program will then output the volume of the cone up to two decimal places using the printf() method. The volume of a cone is given by $\frac{\pi r^2 h}{3}$, where r is the radius of the base of the cone and h is the height of the cone.
  Hint: Instead of declaring a variable (or final identifier) pi and initialize it with a value 3.14159265, you can use Math.PI in the volume expression; In Math class, the identifier PI (upper case P and I) has been declared as a static final double type data field.

  Sample Output:

```
************************************************
Full Name
Student Number
Date
Brief description of the task
************************************************

Please enter the height of the cone: 3.5
Please enter the radius of the cone: 2.5

The volume of a cone with radius 2.50 cm and height 3.50 cm is: 22.91 cubic cm
```

Question 2: Course Grade Calculation (10 marks)

For this question, refer to the slides for "Unit 2.4 Identifiers, and Input operation" and the IntelliJ video from 2021 on OWL called "How to input data using Scanner objects" to get comfortable using the Scanner object. As well, refer to the slides for "Unit 2.5 Operators, Frequently used Methods, Casting and printf()" for using the printf() method and the Arithmetic Operators.

Write a program (according to the specifications outlined below) that calculates a weighted grade of a course that has four assignments.

Specifications:

- Create a package named **Q2.**
- Give your java class an appropriate name.
- The program will print your info and a message that describes its mission as shown in the sample output.
- It will prompt the user for 3 assignment marks (each out of 100), along with their grade weights
  - 1st weight: 0.2
  - 2nd weight: 0.3
  - 3rd weight: 0.5
- A final course grade will be computed based on the user inputs and will be displayed on the standard output using printf() method up to 2 decimal places, as demonstrated in the sample output. Here is the mathematical expression you need to use:
  FinalGrade = grade1 * grade1Weight + grade2 * grade2Weight + grade3 * grade3Weight
  where the '*' sign represents the multiplication operator.
- Note that the grades can be any real number between 0 and 100. We will trust that the user of this code will input the correct values.
- Note:
  - Remember to declare variables before attempting to use them in a program.
  - You are not expected to use any 'loop' or 'array' in this code.
  - Pay close attention to different variable types, and when it is appropriate to use them.

Sample Output:

```
**************************************************
Full Name
Student Number
Date
Brief description of the task
**************************************************

Input the first grade: 80
Input the first-grade weight: 0.2
Input the second grade: 70
Input the second-grade weight: 0.3
Input the third grade: 90
Input the third-grade weight: 0.5

The course grade is: 82.00
```

Question 3: (10 marks)

For this question, refer to the slides in "Unit 3.1: Method Fundamentals" and the IntelliJ video from 2021 on OWL for "static methods". Take a careful look at the examples to help you structure your code.

- Create a package named **Q3.**
- Create a driver java class named **Task** which includes the driver method main() and the commented header with your information.
- Define a method called **myHeader()**, which will accept your full name, your current lab number and the question number as arguments when called and will return nothing. The method will print out a row of stars, your full name in the next line, lab information (lab no., and question no.) in the next line after, and another row of stars in the final line. The method must have the following header:

```
public static void myHeader(datatype fullName, dataType labNum, dataType
quesNum);
```

- Define a method called **myFooter()**, which will accept a message as an argument when called, prints out that message between three asterisks on either side, and returns nothing. The method must have the following header:

```
public static void myFooter(dataType message);
```

- Define a method called **cmToInchConverter()**, which, when called, will accept a length value in centimeters (cm) scale as an argument and return the value in inch scale by using the expression: 1 inch = 2.54 cm. The method must have the following header:

```
public static dataType cmToInchConverter(dataType cm);
```

- Write the driver method (main() method) with the following specifications:
  - Call your myHeader() method with appropriate arguments.
  - Prompt the user to enter a length value in cm. Assume that the use will enter a valid value.
  - Call the cmToInchConverter() and display the return value in inches using printf() method up to two decimal places.
  - Call myFooter() method.
  - Check the sample output so that your code can print similar outputs.

Sample output

```
**************************************************
Full Name
Lab 3, Question 3
**************************************************

Enter the length in cm: 3.3
3.3 cm = 1.3 inch


*** Signing off – yourFullName ***
```

6 Practice Problems

You can try out these practice problems to help with your understanding of concepts. Please ask your TA for help if you need. These questions will not be marked.

Practice Problem 1

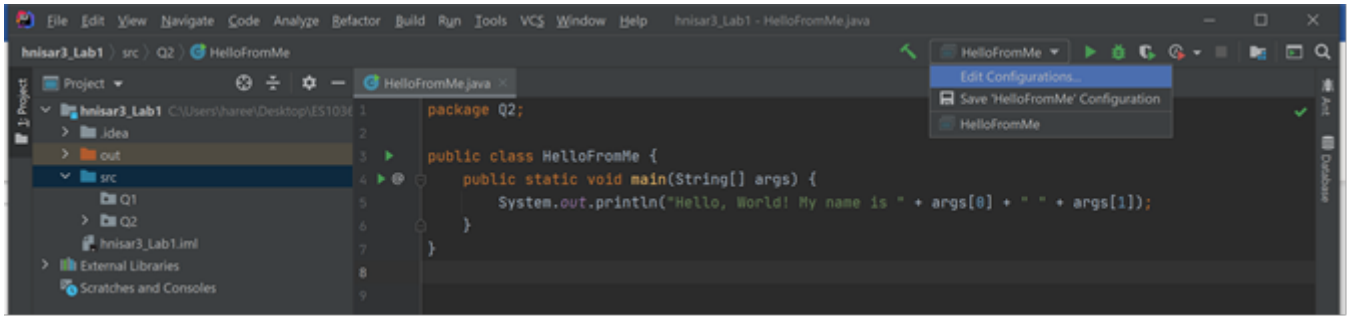
If you are wondering about the purpose of the formal parameters in the main header, this problem will give you some idea about how they work.

- Create a java class named **HelloFromMe** which includes the header with your information. Write your class code to look like this:

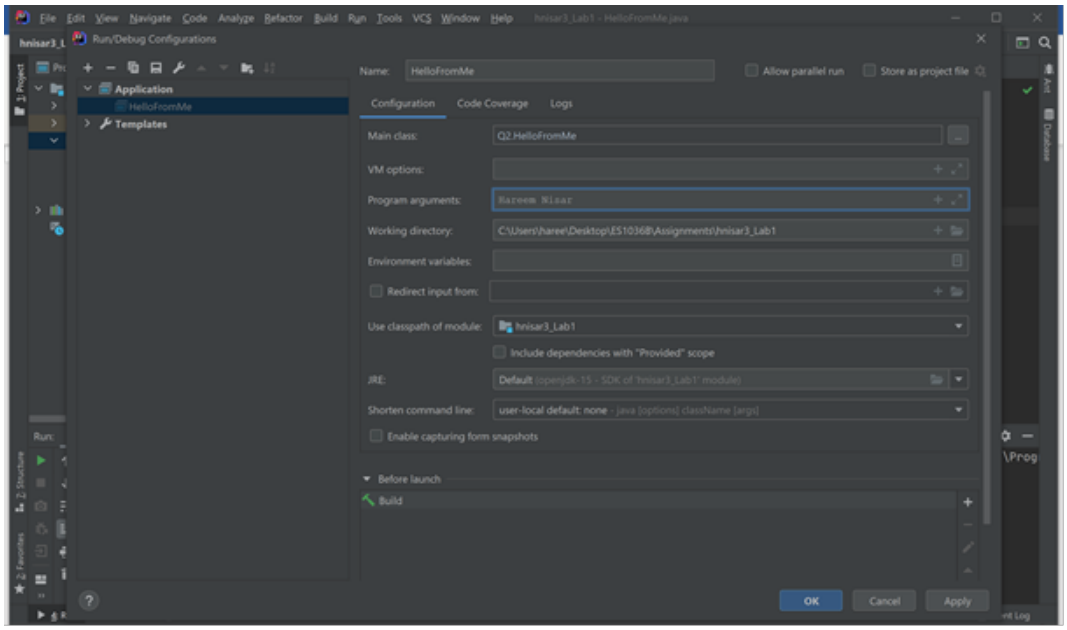
```
class HelloFromMe {
    public static void main(String[] args) {
        System.out.println("Hello, World! My name is " +
args[0] + " " + args[1]);
    }
}
```

- Run your program, you will notice that, it gives you an error "Index 0 out of bounds for length 0". This is because args[0] and args[1] do not have values. So let us fix it now.

- We will pass your first and last name to two arguments (args[0] and args[1]) needed for the program to run.
- To do so, in the righthand side of the main menu click HelloFromMe and **Edit Configuration**



- In the Program arguments filed, enter your first and last name separated by a space and then click Ok button.



- Run your program again and notice it prints the message "Hello, World! My name is *yourfirstname yourlastname*" in the IntelliJ IDE run window. Save your work. (Ctrl+S).

Practice Problem 2

Write a program according to the specifications below that asks the user to input the current year and separates this number into single years, decades, centuries, and millenniums.

Specifications:

- The program will print your info and a brief description of the task as shown in the sample output.
- Your program must ask the user to input a year that is four digits long. We will safely assume that the user will enter a four-digit long year
- Perform required arithmetic operations to separate the four-digit long year into the four separate digits.
- Print the number of years, decades, centuries, and millenniums to the console using printf() method.
- *Hint*: You may find the modulus operator helpful for this task. Please refer to the lecture unit 2

Sample Output:

```
************************************************
Full Name
Student Number
Date
Brief description of the task
************************************************

Please enter the year: 1978

The year is composed of 1 millennium(s), 9 centuries, 7 decade(s), and 8 single
year(s).
```

7 Submission

Zip the project file, name it ***username_Lab3.zip*** and submit in OWL by the end of your lab session.

If you have any questions about the Lab Handout, please contact the Lead TA, Hira Nadeem at hnadeem5@uwo.ca