Wright College + Chapter 12

Object-Oriented Programming (OOP)
Using Classes and Objects and
(Revisit to Arrays and ArrayList)

CIS 144 Java Programming Language +
Introduction to Computer Programming

“Hands-On” Mastering Computer Logic, Design and Programming Using Java Language

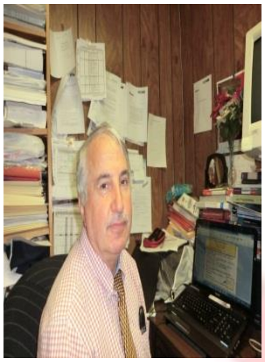
Written By:

Ogar Haji

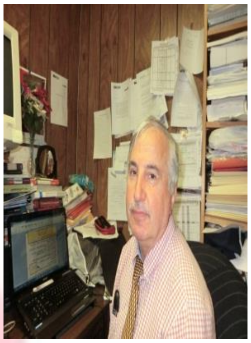
MS Computer Science

DePaul University ✚ Chicago, Illinois

Date Published: April 30, 2021



Computer Text Books Published by the Author: Ogar Haji



The Following is a List of Computer Text Books Published by the Author: Ogar Haji. He has an MS Degree in Computer Science from DePaul University, Chicago, Illinois - USA. Mr. Ogar Haji has over 30 Years of teaching experience at: The College of Office Technology, Oakton College, Washington College, Truman College, Wright College, Triton College, ITT Technical Institute, Phoenix University and East+West University in Chicago Illinois.

1) “Hands-On” Mastering Microsoft Windows 8.1 and 7
2) “Hands-On” Mastering Microsoft Excel 2016 and 2013
3) “Hands-On” Mastering Microsoft Word 2016 and 2013
4) “Hands-On” Mastering Microsoft Access 2013 and 2010
5) “Hands-On” Mastering Microsoft PowerPoint 2016 & 2013
6) “Hands-On” Mastering Microsoft Publisher 2013
7) “Hands-On” Mastering MS Visual Basic .Net Language
8) “Hands-On” Mastering Java Programming Language
9) “Hands-On” Mastering HTML5 & CSS3 Web Page Design
10) “Hands-On” Mastering JavaScript ProgrammingLanguage
11) “Hands-On” Mastering Ruby Programming Language
12) “Hands-On” Mastering Python Programming Language
13) “Hands-On” Mastering QBasic Programming Language
14) “Hands-On” Mastering DOS (Disk Operating System)
15) “Hands-On” Mastering C# Programming Language
16) “Hands-On” Mastering Java ProgrammingLanguage

Send written requests to the Author at the following Address:

Ogar Haji (Computer Instructor)
Harry S. Truman College
1145 West Wilson Avenue
Chicago, Illinois 60640
USA

“H

Computer Labs Rules

1- **No Drinks, Food, Headphones** allowed in Computer labs. And Please **Turn Off the Cell Phones**.

2- When **Lecturing** is in progress, you are Not allowed to work on the computer. Please **Pay Attention** and **Take Notes**.

3- **Attendance and Punctuality** are very **important**. If you are absent, it is your responsibility to make up for the missing work and assignments. **Attendance will be taken daily.**

4- Students should have a **USB Flash Drive** and **Save Projects** to it.

5- **Practice makes perfect**. Please keep practicing the new features or steps repeatedly until the instructor tells you to stop.

6- You must **Concentrate** on what you are doing. **Talking is Not Allowed** in the computer Lab.

7- Please **Study** the Lessons in your **Java Handout and Text Book Daily** and **review your notes before class**. **There will be a Quiz Once a Week**.

8- Please **Check Mark ☑ the Lessons** in the Handout that you **have completed.**

9- You **must do All Java works**, **Assignments and Tests** located at the End of each Chapter on Time.

CIS144 Java Programming

Instructor:

Ogar Haji

Chapter 12

Object-Oriented Programming (OOP)

Using Classes and Objects

Create a Class and Instantiate an Object

You will learn the following in Chapter 12

- **What is OOP**
- **What is a Class in Visual C# Programming Language**
- **What is an Object in Visual C# Programming Language**
- **How to Create an Object from a Class using new keyword**
- **What are get and set Accessors**
- **Create an Employee Class Lab Exercise**
- **Create a TVProject OOP Class Lab Exercise**
- **Create Vending Machine Simulator Lab Exercise**
- **Using Two Forms and display the second form**
- **Create Car Information Form Lab Exercise**
- **Create BankAccount Class and Windows Form Project to Simulate a Bank Deposit and Withdrawal Project in C# language**
- **Do Lab Assignment 12 NewHadra Pizza Restaurant Project**
- **Do Chapter 12 Homework # 12**

Input/Output

Flowchart Symbols

Processing

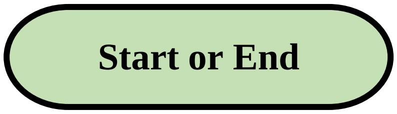
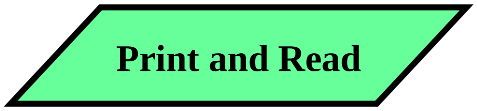
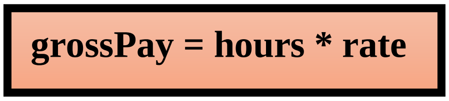
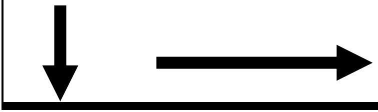
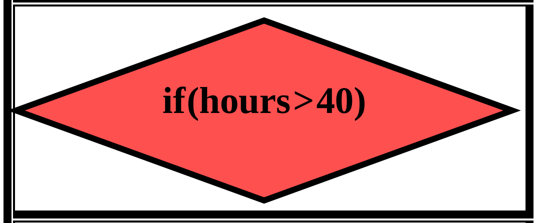
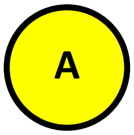
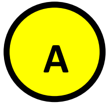
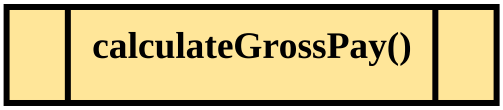
+++ Review +++

Lesson 240 Review : What are the Flowchart Symbols used in Java Language?

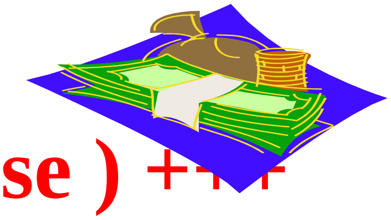
You should always **draw a Flowchart** when you Design, Code and Solve a problem in Java language.

Before you Code a program in Java Language, you have to **Draw a Flowchart** to solve the problem of the program you want to code.

The following symbols are used with Java Programming Language:

Symbol	Symbol Name	Usage
	**Oval (Beginning and Terminal)** symbol	Use **Oval (Beginning and Terminal) Symbol** at the Beginning of the Flowchart and at the End of the Flowchart. Use with Start and End statements.
	**Parallelogram (Input/Output)** Symbol	Use **Parallelogram (Input /Output)** or **I/O** symbol to Input Data, Read Input or Print Output
	**Rectangle** Symbol	Use the **Rectangle** Symbol for Calculating, Assigning Values
	**FlowLine** Symbol	Use **FlowLine** Symbols to show the Flow or Sequence of the flowchart.
	**Diamond (Decision)** Symbol	Use **Diamond (Decision)** Symbol with the If or Select statements when deciding if Hours is > 40. The Result will be either True or False.
	**Connector** Symbol	Use **Connector** Symbol to Connect the Flowchart rather than draw a long Arrow. Use 
	**Function** or **Method (Predefined Process)** Symbol.	Use **Function** or **Method (Predefined Process)** Symbol to call another Function or Method that contains coding statements.

Calculate Gross Pay of Employees Project



+++ (Review Lab Exercise) +++

Lesson 241 Ex : How to Calculate Gross Pay of Employees Project?

Problem or Project: Design and Code in Java Language the project to Calculate Gross Pay of Employees in a company.

Do the following 12 Must Steps to Design, Code and Solve a project using Java Language.

Do Steps 1 thru 7 in your Note Book or on Paper.

Step 1) Purpose of the Program: State what Program will do: (5 Points)

a) This Program will calculate Gross Pay of Employees.

b) It will ask the User to Enter Employee's Full Name:

c) It will ask the User to Enter Number of Hours Worked and

d) It will ask the User to Enter Hourly Rate.

e) The program will then calculate Gross Pay.

Gross Pay = Hours * Rate

f) Display the Gross Pay

Step 2) Input: You should know how the Input looks like: (5 Points)

```
Enter Employee's Full Name:   Ogar Haji
Enter Hours Worked:           40
Enter Hourly Rate:            10
```

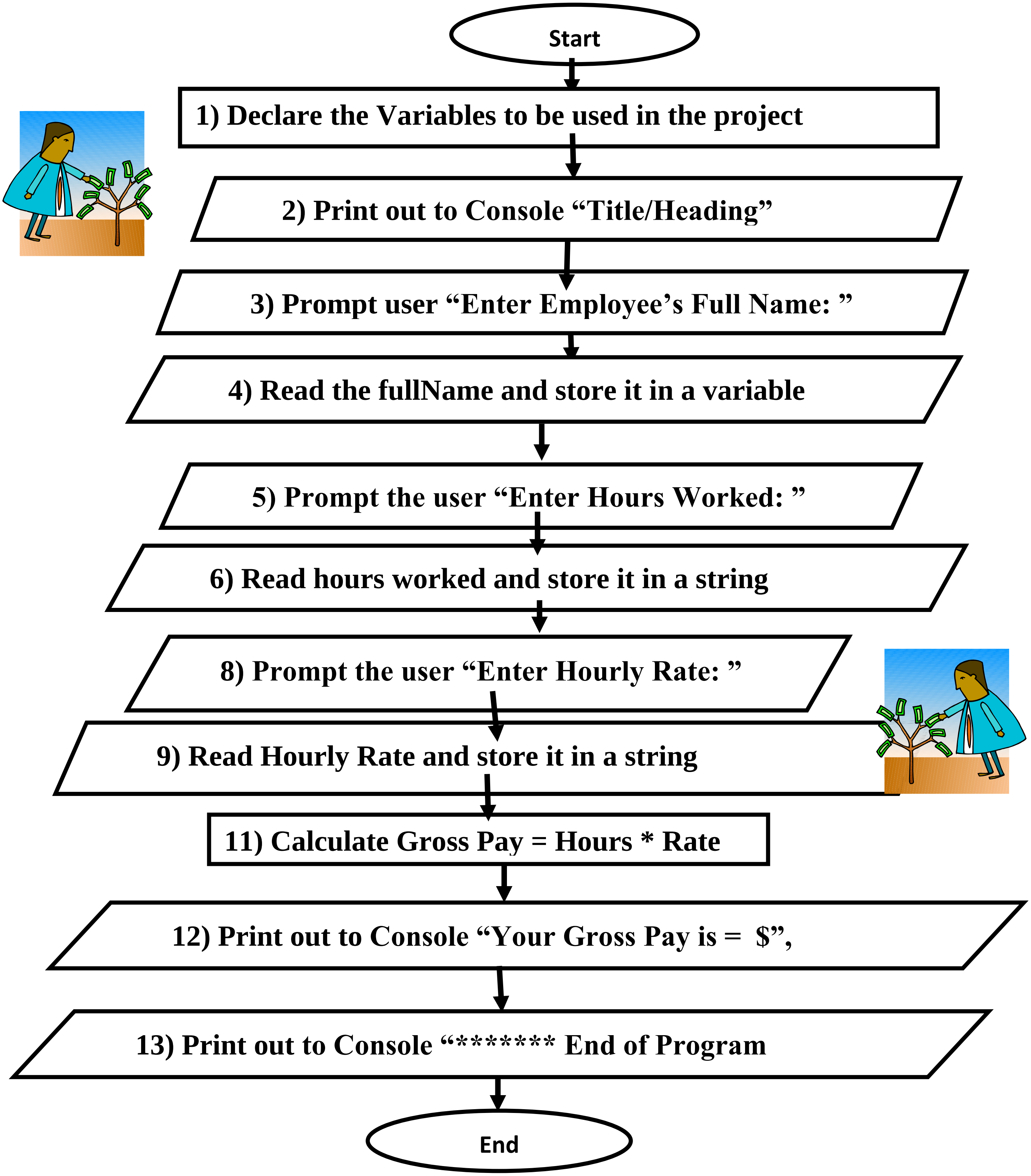
Step 3) Processing and Calculation: The program will process each record and Calculate Gross Pay: (5 Points)

Gross Pay = Hours * Rate

Step 4) Output: You should know how output should look like: (5 Points)

```
********** Calculate Gross Pay Project **********
The Employee Ogar Haji Gross Pay is = $ 400
*********** End of Program ************
```

Step 5) Flowchart: Draw a Flowchart for Gross Pay program.(5 Points)


Start
1) Declare the Variables to be used in the project
2) Print out to Console "Title/Heading"
3) Prompt user "Enter Employee's Full Name: "
4) Read the fullName and store it in a variable
5) Prompt the user "Enter Hours Worked: "
6) Read hours worked and store it in a string
8) Prompt the user "Enter Hourly Rate: "
9) Read Hourly Rate and store it in a string
11) Calculate Gross Pay = Hours * Rate
12) Print out to Console "Your Gross Pay is = $",
13) Print out to Console "******* End of Program
End

Step 6) PseudoCode: Write a PseudoCode for the Program.(5 Points)

```
1) Declare the variables to be used in the program
2) Write to Console “The Title or Heading of the program “
3) Prompt the user “Enter Employee’s Full Name: “
4) Read from Console fullName and store the string in a variable
5) Prompt the user “Enter Hours Worked: “
6) Read from console hours worked and store in a hours variable
7) Prompt the user “Enter Hourly Rate: “
8) Read from console the rate and store in a rate variable
9) Calculate Gross Pay = Hours * Rate
10) Write to Console “The Employee Gross Pay is = $“, gross_pay
11) Write to Console “******* End of Program *******”
```

Step 7) Code the Program in Java by referencing the Flowchart or Pseudocode you designed above and Save it as CalculateGrossPay to USB.

a)Type the following Java code in NetBeans IDE.

Code for the first part of “CalculateGrossPay” project follows:

```
/*
 ************************************************************
  Purpose of the Project:
  a) This Interactive Project will prompt the user to enter his/her (First Name,
Last Name, and Hours Worked and  Rate) then it will read the text entered and
store it  in string variables. It Converts the string Hours to       integer numbers
into integer and converts the Rate to Double and then Calculate the Gross Pay.
  b) Project Name: CalculateGrossPay
  c) Date: Saturday, December 28, 2013
  d) Programmer: Instructor – Ogar Haji
 ************************************************************
*/
package calculategrosspay;
```

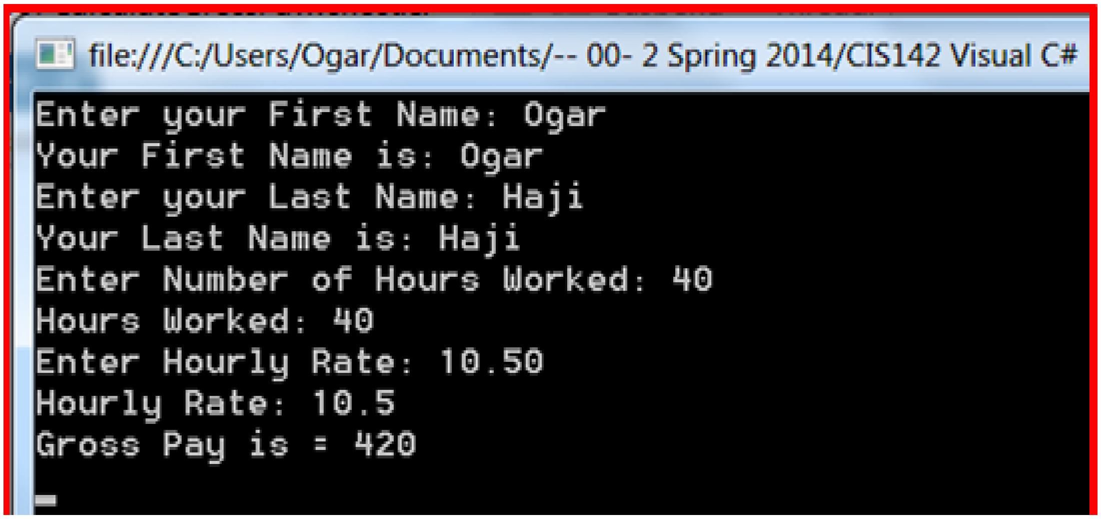
```
// import the Classes needed in this project
import java.util.Scanner;
public class CalculateGrossPay {
   public  static void main(String[] args) {
        // 1) Declare the variables to be used in the project
        String firstName, lastName;
        int hours = 0;
        double rate = 0.0;
        double grossPay = 0.0;
        // 2) Prompt the User to Enter his/her First Name
        System.out.print("Enter your First Name:  ");
        // 3) Read First Name from console and store it in a variable
        firstName = input.nextLine();
        // 4) Print First Name to the Console
        System.out.println ("Your First Name is: " + firstName);
        // 5) Prompt the User to Enter his/her Last Name
        System.out.print ("Enter your Last Name: ");
        // 6) Read his/her Last Name from console and store it in a variable
        lastName = input.nextLine();
        // 7) Print out Last Name to the Console
        System.out.println ("Your Last Name is: " + lastName);
        // 8) Prompt the User to Enter Hours Worked
        System.out.print ("Enter Number of Hours Worked: ");
        // 9) Read Hours Worked from Console and Store it in a variable
        hours = input.nextInt();
        // 10) Print out Hours Worked to the Console
        System.out.println ("Hours Worked: " + hours);
        // 11) Prompt the User to Enter Hourly Rate
        System.out.print ("Enter Hourly Rate: ");
```

```
            // 12) Read Hourly Rate from Console and Store it in a variable
            rate = input.nextDouble();
            // 13) Print out Hourly Rate to the Console
            System.out.println ("Hourly Rate: " + rate);
            // 14) Calculate Gross Pay
            grossPay = hours * rate;
            // 15) Print out grossPay to the console
            System.out.println ("Gross Pay is = " + grossPay);
    }
  }
}
```

Step 8) Click **Start** button to Start Running the program

The following output appears on the Left side of the screen with the Input you entered and the correct calculated GrossPay $400.

If any Syntax Errors Found Do Next Step 9:



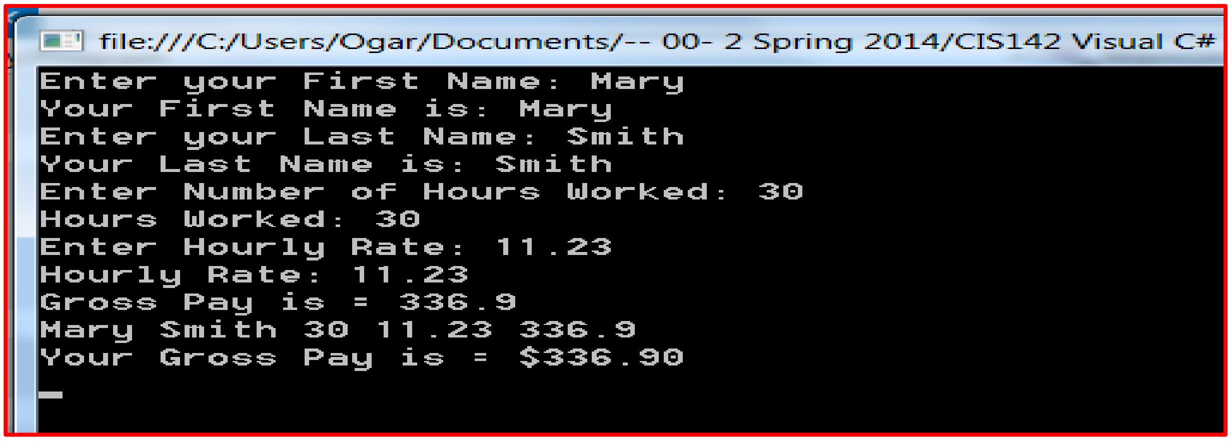
file:///C:/Users/Ogar/Documents/-- 00- 2 Spring 2014/CIS142 Visual C#

```
Enter your First Name: Ogar
Your First Name is: Ogar
Enter your Last Name: Haji
Your Last Name is: Haji
Enter Number of Hours Worked: 40
Hours Worked: 40
Enter Hourly Rate: 10.50
Hourly Rate: 10.5
Gross Pay is = 420
_
```

Step 9) Debug the Program: Debug or Correct any Syntax Errors until you have a clean Compiled program. (5 Points) (Clean compiled program means No Errors in the program).

Step 10) Test the Program: Test the Program with Test Data. (5 Points)

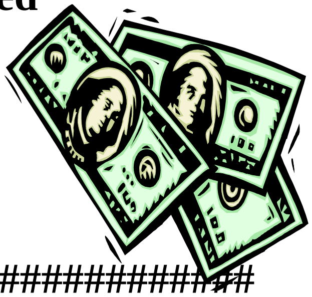
Repeat Step 10) Test the program many Times and Test the Program again and again until All conditions are tested:


file:///C:/Users/Ogar/Documents/-- 00- 2 Spring 2014/CIS142 Visual C#

```
Enter your First Name: Mary
Your First Name is: Mary
Enter your Last Name: Smith
Your Last Name is: Smith
Enter Number of Hours Worked: 30
Hours Worked: 30
Enter Hourly Rate: 11.23
Hourly Rate: 11.23
Gross Pay is = 336.9
Mary Smith 30 11.23 336.9
Your Gross Pay is = $336.90
_
```

Step 11) Documentation (5 Points): You have to add more comments to the Program (like Comments about the Purpose of the Program, Your Name and the Date the Program was written.)

```
####################################################################
# Purpose of the Program:
#  a) This Program will calculate Gross Pay.
#  b) It will ask the User to Enter Employee's Full Name:
#  c) It will ask the User to Enter Number of Hours Worked
#  d) It will ask the User to Enter Hourly Rate.
#  e) The program will calculate Gross Pay.
#       Gross Pay = Hours * Rate
#  f) Display Gross Pay
####################################################################
```



Step 12) Print a Copy of Java Code along with screen printout of the Running program. Submit to your Instructor the Print Copy and the screen Printout (Snaps) along with the following: (Which you did on Paper)

Copy the Java Code and the Output of the program and Paste it in Microsoft Word program:

1) **Purpose of the Program.**
2) **Input: how the Input looks like**
3) **Processing and Calculations**

4) Output: how the Output will look like

5) Flowchart 6) Pseudocode

7) Java Code and

8) Print out copy of Java code and Output after running the program.

Submit the Programs on Time.

Remember Points will be deducted (20%) for Programs submitted Late.

Important Note:

1) Do Steps 1 thru 7 on Paper.

2) Then Get into NetBeans IDE

3) Type the Java code.

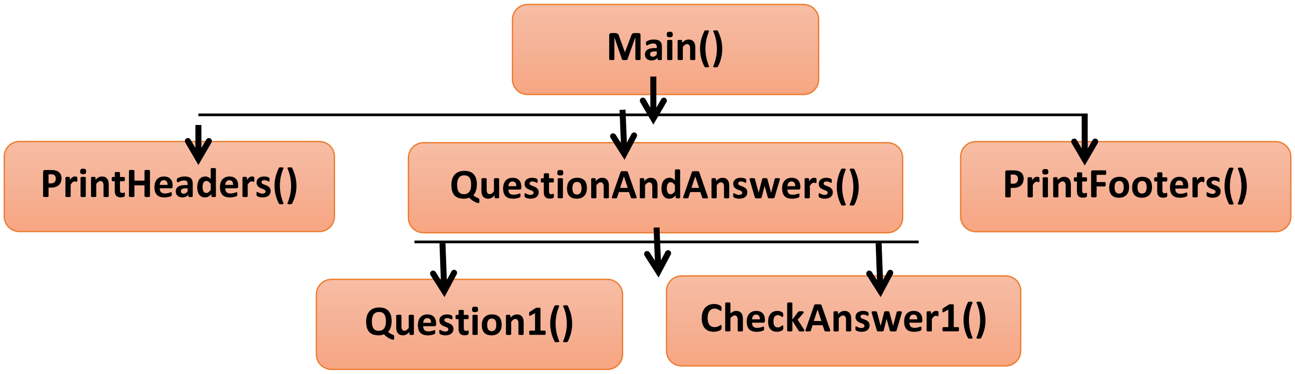
4) Save All the Files

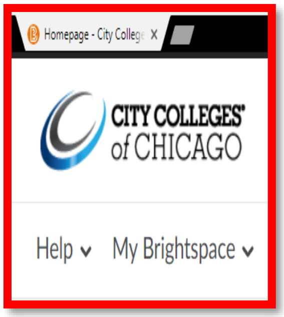
5) Run the Program and Test it with Test Data for All Conditions.

Instructor: Ogar Haji

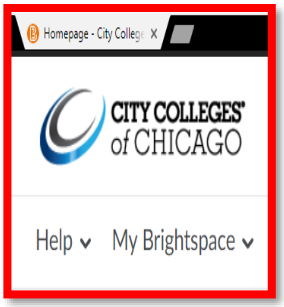
Do Only 1 Modification at a Time

Use Structure Programming:





Upload Your Weekly Assignments To Brightspace Correct Weekly Assignments Folder +++ Review +++



Lesson 242 : How to Upload Your Weekly Assignments to Brightspace Correct Weekly Assignments Folder?

You have to Upload your Weekly Assignments and Homework to Brightspace correct Weekly Assignments Folder as following:

1) Copy the Java Code from NetBeans to Word document:

1. Copy the **Java Code from** the **NetBeans IDE** and **Paste it into** the **Microsoft Word Document.**

2. In NetBeans IDE, press **Ctrl+A (select All)** to **select All** the **Java code**.

3. Press **Ctrl+C (Copy)** to **Copy** the **selected Java code** into **computer memory RAM.**

2) Paste the Java Code into Microsoft Word: W

1. Get into **Microsoft Word document then** press **Ctrl+V (paste) to Paste** the **copied Java code** from **memory** into **Word document.**

2. Press **Ctrl+Home (go to the Top of Document)** and **type** your **Full Name** at the **top of document followed by** the **Java File Name** in **size 20 and bold**.

3) Print the Screen of the Output of Java NetBeans:

1. Run the **Java project** and **make sure the program** is **running** with **correct output.**

2. Press **PrintScreen button** Prt Scr Sys Rq to **capture** the **output screen shot.**

4) Paste the Print Screen of Java output into Paint program:

1. Get into **Paint program** and press **Ctrl+V (paste) to Paste** the **screen shot** in **Paint** program.

2. Inside the **Paint program,** Click **Select** Select **icon** and then **Select only** the **Output** of the **Java project.**

3. Press **Ctrl+C (Copy)** to **Copy** the **selected output image.**

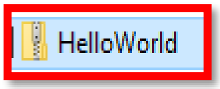
5) Get back into Microsoft Word program:

1. Go back to **Word Document,** press **Ctrl+End (End of Document)** to **go to End of document.**

2. In the **Word document,** press **Ctrl+V (Paste)** to **paste** the **Java output** there.

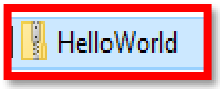
3. Save the **Word Document** as the **Name of the Java project** and in this **example (Save File as HelloWorld project)**

6) To Compress or Zip the Java project:


HelloWorld

1. Right-Click on the **Java project (HelloWorld)** that is **saved on your computer**.

2. Point to **Send to,** then **click on Compressed (Zipped) Folder** and you will have **another File** which is Compressed or Zipped.


HelloWorld

7) To Upload the 2 Files to Brightspace:

1. Log on Brightspace with your **User Name** and **Password**

2. Click on your course **CIS 144 Java** to **Select it.**

3. Click on **Assignments** ▼

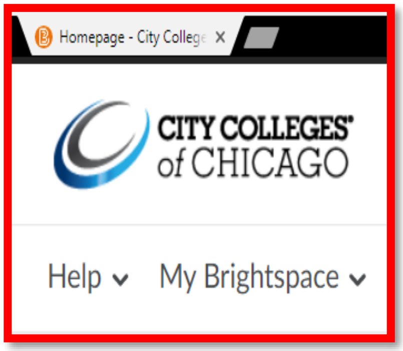
4. Click on **Assignments**

5. Click on **Week 11 Assignments Folder**

6. Click on **"Add a File"** button

7. Click on **"My Computer"**

8. Click on **Upload** button



9. Go to the **location** where you **saved the Java project "HelloWorld".**

10. Click on the **File** or **Folder (HelloWorld)**

11. Click on **Add** button and the **File** or **Folder** will be **added** to the **Week 01 Assignments Folder.**

Note: Always **Upload** to **Brightspace 2 Files** of **same Java Project:**

1) The **Microsoft Word Document** of the **Java Project Code along with the Java Output Screen shots.**

2) The **Compressed** or **Zipped File** or **Folder** of **Java Project**.

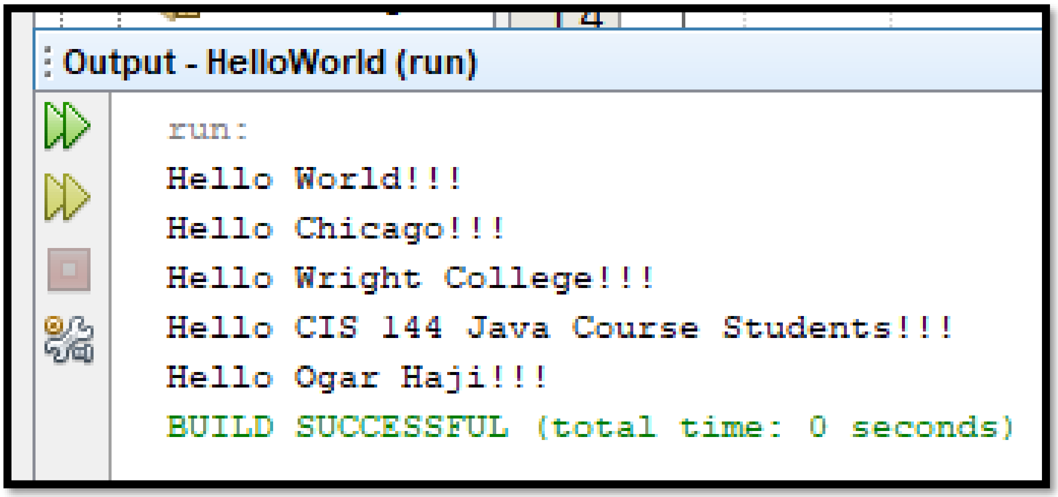
The Java project “HelloWorld” code in Word Document along with the Output Screen Shots appear as following:

Ogar Haji

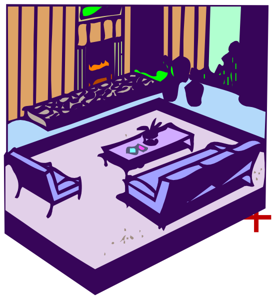
CIS 144 Java

Project Name: HelloWorld

```
/*
Project Name: HelloWorld
This Java project will print the message "Hello World" to screen
Programmer: Instructor + Ogar Haji (Type your Full Name)
Date:   June 01, 2017
*/

package helloworld;

public class HelloWorld {
   public static void main(String[] args) {
      // This project will print the Literal String “Hello World” to output screen

      System.out.println ("Hello World!!!");
      System.out.println ("Hello Chicago!!!");
      System.out.println ("Hello Wright College!!!");
      System.out.println ("Hello CIS 144 Java Course Students!!!");
      System.out.println ("Hello Ogar Haji!!!");
   }
}
```



```
run:
Hello World!!!
Hello Chicago!!!
Hello Wright College!!!
Hello CIS 144 Java Course Students!!!
Hello Ogar Haji!!!
BUILD SUCCESSFUL (total time: 0 seconds)
```



Calculate The Area of Room: Prompting User for Length and Width Of the Room Using Methods

+++(Do Lab Exercise 1) 100 Points+++

Do Lab Exercise 1

Lesson 243 Ex: How to Calculate Room Area Using Methods in Java ?

Problem or Project: Design and Code in Java Language the project to prompt user to input the Length and Width of the Room then Calculate the Area of a Room using Methods().

Follow the 12 Steps to Design, Code and Run a project in Java language.

1) In this Java project, you prompt user to enter the Room Length in feet.

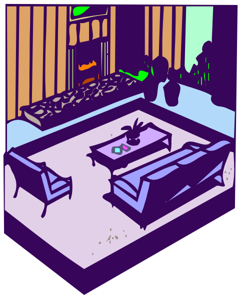
2) Then you prompt user to enter the Room Width in feet.

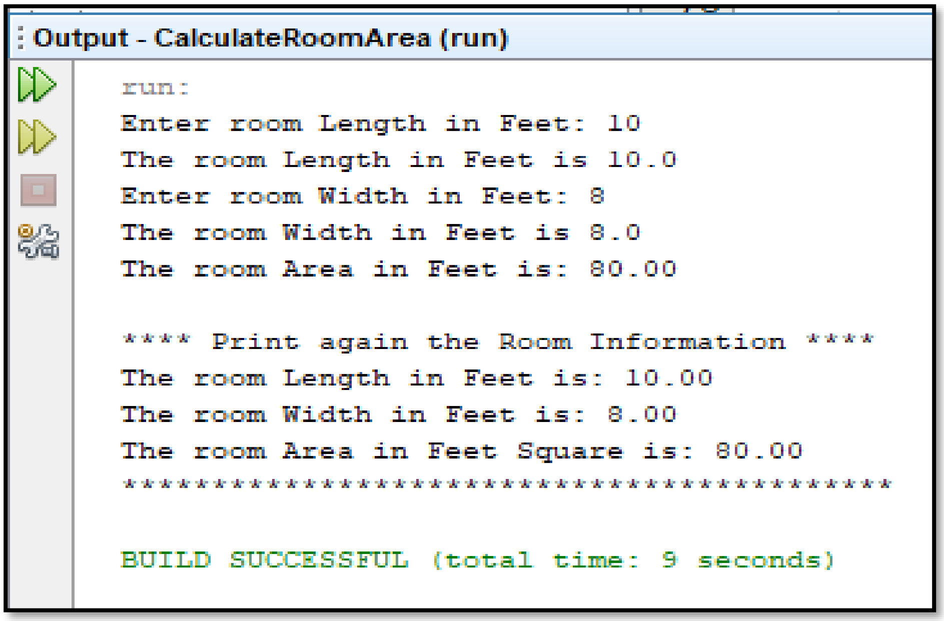
3) Then you use methods to get the Room Length.

4) Then you use methods to get the Room Width.

To Calculate the Room Area = Length * Width.

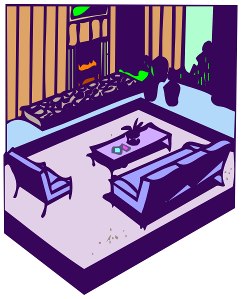
The Input/Output of the Java project looks like the following:





Output - CalculateRoomArea (run)

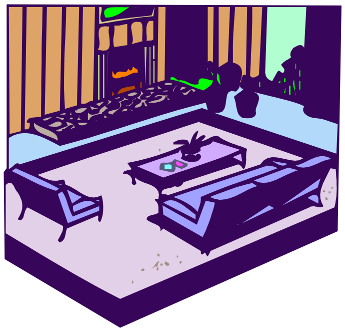
```
run:
Enter room Length in Feet: 10
The room Length in Feet is 10.0
Enter room Width in Feet: 8
The room Width in Feet is 8.0
The room Area in Feet is: 80.00

**** Print again the Room Information ****
The room Length in Feet is: 10.00
The room Width in Feet is: 8.00
The room Area in Feet Square is: 80.00
******************************************

BUILD SUCCESSFUL (total time: 9 seconds)
```



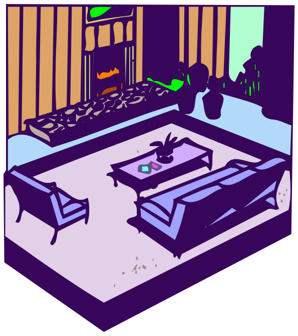
1) Ex. Type the following Java Code and
2) Save as CalculateRoomAreaUsingMethods.

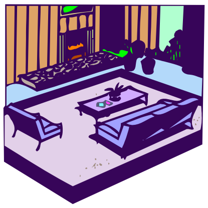
Do Lab Exercise

```
/*
   This project calculates the Area of a Room.
   It prompts the user to enter the Length and Width
   of the Room in Feet then it calculates the Room Area
   Programmer: Ogar Haji (Java Instructor)
   Date: Tuesday 05/05/2020
*/
package calculateroomareausingmethods;
// Import the Scanner Class used to read input from the keyboard
import java.util.Scanner;
public class CalculateRoomAreaUsingMethods {
// Declare Global variables to be used in the Entire project
public static Scanner input = new Scanner(System.in);
public static double length;
public static double width;
public static double area;
   // This is the main() method which is executed first
   public static void main(String[] args) {
     // 1) Call getRoomLength to get Length of the room
     length = getRoomLength();
     // 2) Call getRoomWidth to get Width of the room
     width = getRoomWidth();
     // 3) Call calculateRoomArea to Calculate Area of the room
     area = calculateRoomArea(length, width);
     // 4) Call printRoomInfo to print room Information
     printRoomInfo(length, width, area);
     } // end of main() method
```



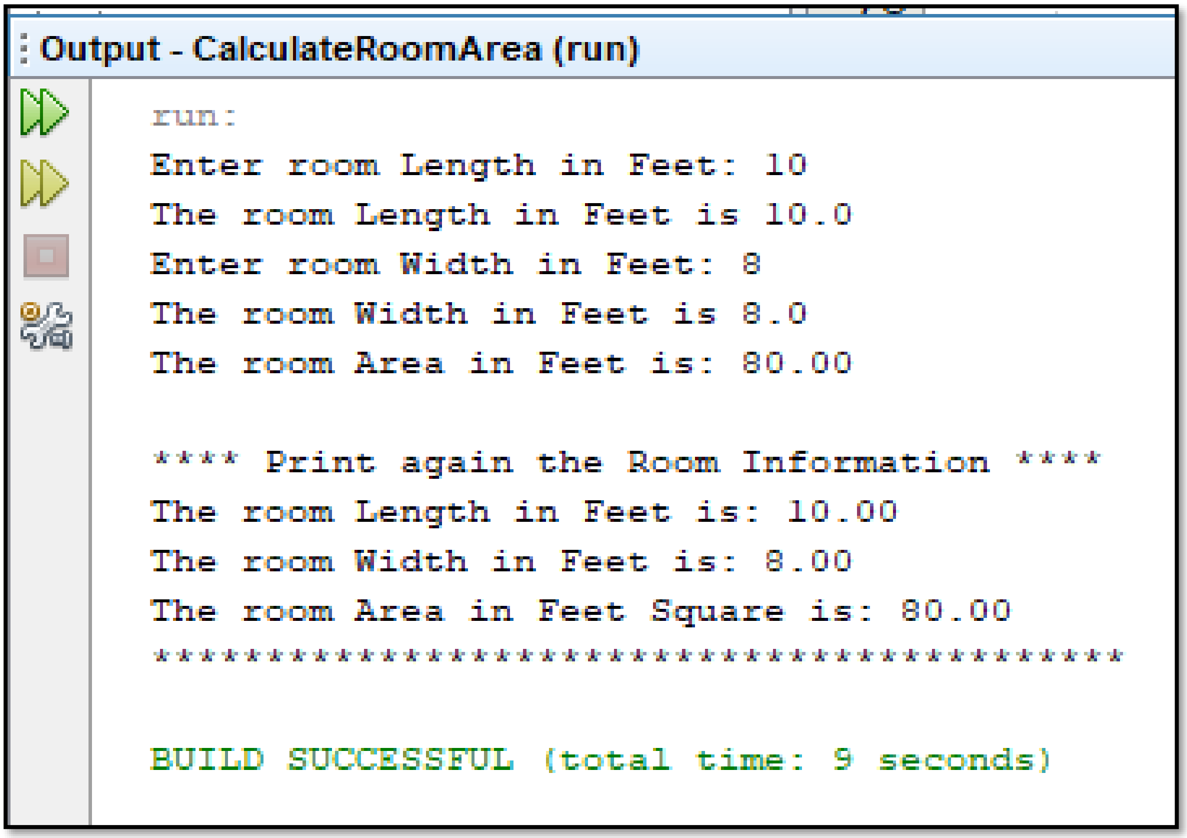
```
// 1) Define and code getRoomLength() method with double return type
 public static double getRoomLength() {
   // 1) Prompt the user to Enter room Length and get it
   System.out.print("Enter room Length in Feet: ");
   length = input.nextDouble();
     // 2) Echo back the length of the room
   System.out.println("The room Length in Feet is " + length);

   // 3) Return the length to the method it called it
   return length;
 }

 // 2) Define and code getRoomWidth() method with double return type
 public static double getRoomWidth() {
   // 1) Prompt the user to Enter room Width and get it
   System.out.print("Enter room Width in Feet: ");
   width = input.nextDouble();

   // 2) Echo back the Width of the room
   System.out.println("The room Width in Feet is " + width);

   // 3) Return the width to the method it called it
   return width;
 }

 // 3) Define and code calculateRoomArea() method with double return type
 public static double calculateRoomArea(double length, double width) {
  // 1) Calculate the area = length * width
  double area = length * width;
  // 2) Print out the Room Area using printf
  System.out.printf("The room Area in squared Feet is: %.2f %n", area);
  // 3) Return the area to the method it called it
  return area;
 }
```





```
    // 4) Define and code printRoomInfo() method with parameters received
    public static void printRoomInfo(double length, double width,
                                      double area) {
      System.out.println("\n**** Print again the Room Information ****");

      // 1) Print out again the Length, width and the area
      System.out.printf("The room Length in Feet is: %.2f %n", length);
      System.out.printf("The room Width in Feet is: %.2f %n", width);
      System.out.printf("The room Area in Squared Feet is: %.2f %n", area);
      System.out.println("*******************************************\n");
    }
}
```

The Input/Output of the Java project looks like the following:



```
run:
Enter room Length in Feet: 10
The room Length in Feet is 10.0
Enter room Width in Feet: 8
The room Width in Feet is 8.0
The room Area in Feet is: 80.00

**** Print again the Room Information ****
The room Length in Feet is: 10.00
The room Width in Feet is: 8.00
The room Area in Feet Square is: 80.00
*******************************************

BUILD SUCCESSFUL (total time: 9 seconds)
```

Modify the project to find the following:

1) **Add** the **printHeadings() method** to **print nice Headers** for the project.
2) **Add** the **printFootings() method** to **print nice Footers** for the project.
3) **Find** the **Perimeter** of the **Room** using a **Method** and **Apply formula to calculate Room Perimeter.**

Room Perimeter = (length + width) * 2 .

4) **Find** the **Amount** to **Carpet** the **Room Floor** using a **Method**. **Prompt** the **user** to **enter** the **Cost of Carpeting** for a **Squared Foot (like 8.88)**.

Then **print** the **Amount Charged** to **Cover** the **Floor with carpet = Area * Cost of Carpeting for a Squared Foot**

Amount Charged to cover Room Floor with Carpet = $ 710.40

5) **Find the Amount to Print the Walls of the Room (5 Walls) and of course you must prompt the user for the Height of the Room. Suppose it will cost $4.27 to paint a Squared Feet of wall.**

Using For statement to Loop in an Array:
int[ ] scores = {100, 77, 99, 88, 55, 82, 66};

+++(Do Lab Exercise 2) 100 Points+++

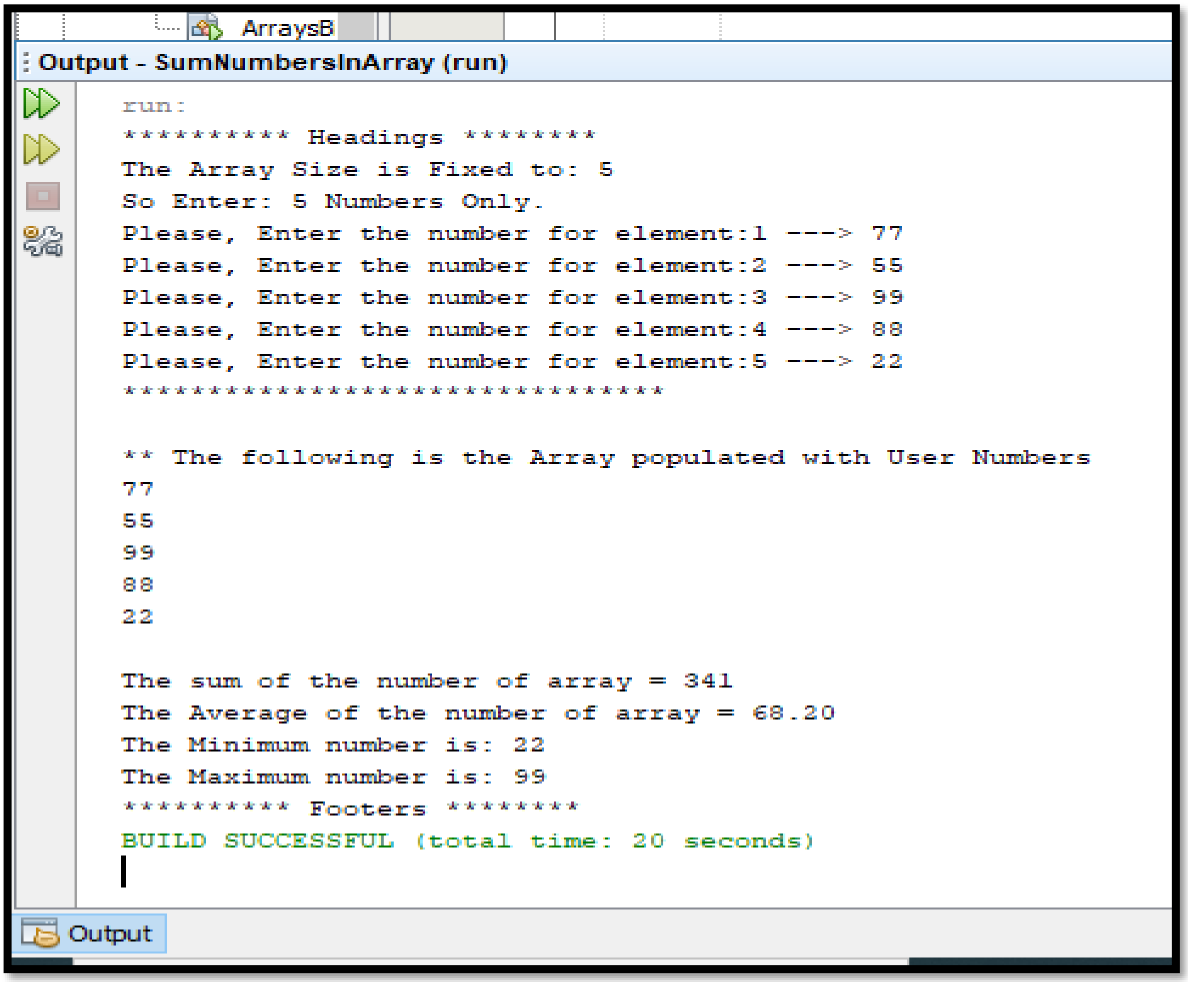
Do Lab Exercise 2

Lesson 244 Ex: How to use For statement to Traverse an Array in Java ?

Problem or Project: Design and Code in Java Language the project to Create a Fixed Array. This project will prompt the user to create an Array Fixed in Size and then will prompt the user to populate the array with numbers using Methods().

Use For statement to loop or traverse in an array.

The Input/Output of the Java project looks like the following:



```
run:
*********** Headings ********
The Array Size is Fixed to: 5
So Enter: 5 Numbers Only.
Please, Enter the number for element:1 ---> 77
Please, Enter the number for element:2 ---> 55
Please, Enter the number for element:3 ---> 99
Please, Enter the number for element:4 ---> 88
Please, Enter the number for element:5 ---> 22
*********************************

** The following is the Array populated with User Numbers
77
55
99
88
22

The sum of the number of array = 341
The Average of the number of array = 68.20
The Minimum number is: 22
The Maximum number is: 99
*********** Footers ********
BUILD SUCCESSFUL (total time: 20 seconds)
```

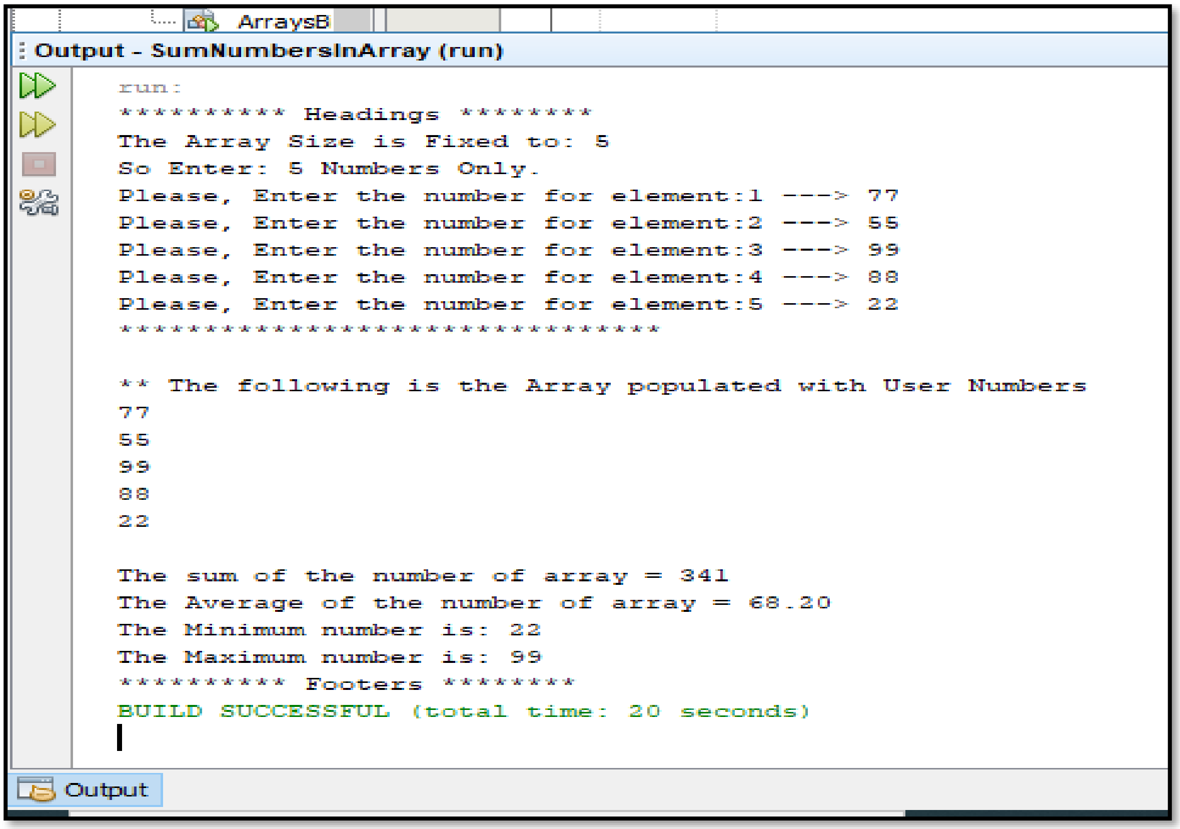
1) Ex. Type the following Java Code and

2) Save as SumNumbersInArray. Do Lab Exercise

```
/*
 * This project will prompt the user to create an Array
and populate the array with numbers.
Programmer Ogar Haji
Date: 04/30/2020.
 */
package sumnumbersinarray;
// Import the Classes needed in this Java project
import java.util.Arrays;
import java.util.Scanner;
public class SumNumbersInArray {
 public static int[] numberArray = new int[5];
// 1) Declare an object called 'input' from Scanner class
public static Scanner input = new Scanner (System.in);
// 2) Declare variables as Global so they will be used across all the methods
public static int arrayLength;
public static int number;
int sum = 0;
public static double average;
   public static void main(String[] args) {
        // 1) Call the printHeaders() method to print the headings
       printHeaders();
        // 2) Call the promptUserForInput() method to prompt the user for the input
       promptUserForInput();

        // 3) Call the printPopulatedArray() method to print the Array

       printPopulatedArray();

      // 4) Call the sumAndAverageNumbersInArray() method to find sum and average
        sumAndAverageNumbersInArray();
      // 5) Call the sortArrayAndPrintMaxandMin() method to print Max & Min numbers
      sortArrayAndPrintMaxandMin();

        // 6) Call the printFooters() method to print the Footers
       printFooters();

   } // End of main() method
// 1) Define and code the printHeaders() method
    public static void printHeaders(){
        System.out.println("********** Headings ********");
    }
// 2) Define and code the pomptUserForInput() method
   public static void promptUserForInput(){
       int arraySize = 5;

       System.out.println("The Array Size is Fixed to: " + arraySize );
       System.out.println("So Enter: " + arraySize + " Numbers Only.");

       for (int i = 0; i < arraySize; i++){
        // prompt the user to enter the number
       System.out.print("Please, Enter the number for element:" +(i+1) + " ---> ");
          number = input.nextInt();

          numberArray[i] = number;
       }

       System.out.println("********************************");

    }

// 3) Define and code the printPopulatedArray()  method
   public static void printPopulatedArray(){

       System.out.println("\n** The following is the Array populated with User
Numbers");
       for (int k = 0; k < numberArray.length; k++ ){
          System.out.println(numberArray[k]);
       }

       System.out.println();

   }
// 4) Define and code the sumAndAverageNumbersInArray() method
  public static void sumAndAverageNumbersInArray(){

   // Sum the values for the array
      int sum = 0;

       for ( int m = 0; m < numberArray.length; m++){
          sum += numberArray[m];
       }

   average = (double) sum / numberArray.length;
   System.out.println("The sum of the number of array = " + sum);
  System.out.printf("The Average of the number of array = %.2f %n", average);

  }
```

```
// 5) Define and code the sortArrayAndPrintMaxandMin() method
  public static void sortArrayAndPrintMaxandMin(){

   Arrays.sort(numberArray);

   System.out.println("The Minimum number is: " + numberArray[0]);
   System.out.println("The Maximum number is: " +
                              numberArray[numberArray.length - 1]);
  }
// 6) Define and code the printFooters() method
   public static void printFooters(){
      System.out.println("********** Footers ********");
   }
}
```

The Input/Output of the Java project looks like the following:



```
run:
********** Headings ********
The Array Size is Fixed to: 5
So Enter: 5 Numbers Only.
Please, Enter the number for element:1 ---> 77
Please, Enter the number for element:2 ---> 55
Please, Enter the number for element:3 ---> 99
Please, Enter the number for element:4 ---> 88
Please, Enter the number for element:5 ---> 22
********************************

** The following is the Array populated with User Numbers
77
55
99
88
22

The sum of the number of array = 341
The Average of the number of array = 68.20
The Minimum number is: 22
The Maximum number is: 99
********** Footers ********
BUILD SUCCESSFUL (total time: 20 seconds)
```

// 1) Modify the project to accept numbers Between 0 and 100 inclusive

// 2) Prompt the user if he/she wants to continue (Y/N)

```
/*
 * This project will prompt the user to create an Array
and populate the array with numbers.
Programmer Ogar Haji
Date: 04/30/2020.
 */
package sumnumbersinarray;
// Import the following which are needed in this project
import java.util.Arrays;
import java.util.Scanner;
public class SumNumbersInArray {
 public static int[] numberArray = new int[5];
// 1) Declare an object called 'input' from Scanner class
public static Scanner input = new Scanner (System.in);
// 2) Declare variables as Global so they will be used across all the methods
public static int arrayLength;
public static int number;
int sum = 0;
public static double average;
public static String answer = "y";

// This is the main() method which the program starts executing from here
```

```
public static void main(String[] args) {
   // 1) Call the printHeaders() method to print the headings
   printHeaders();

// 2) Prompt the user if he/she wants to continue (Y/N)
  // 2) Use while statement to ask the user if he/she wants to run the project again
   while (answer.equalsIgnoreCase("Y") ) {

  // 3) Call the promptUserForInput() method to prompt user to input numbers
      promptUserForInput();

 // 4) Call the printPopulatedArray() method to print the array populated by user
  printPopulatedArray();

    // 5) Call sumAndAverageNumbersInArray() method to find the Sum & Average
      sumAndAverageNumbersInArray();

  // 6) Call sortArrayAndPrintMaxandMin() method to print Max & Min numbers
     sortArrayAndPrintMaxandMin();

     input.nextLine(); // Read the End of line enter key

     System.out.print("Do you want to Run the program again (Y/N): ");

      answer = input.nextLine();

      System.out.println("The answer is: " + answer);

       if ( !(answer.equalsIgnoreCase("Y")) )  {

           break;

         }

   }
  // 7) ) Call the printFooters() method to print the footers
   printFooters();

} // End of main() method
```

```
// 1) Define and code the printHeaders() method
    public static void printHeaders(){
      System.out.println("*********** Headings ********\n");

   }

   // 2) Define and code the promptUserForInput() method
   public static void promptUserForInput(){
     int arraySize = 5;

     System.out.println("The Array Size is Fixed to: " + arraySize );

     System.out.println("So Enter: " + arraySize + " Numbers Only.");

   // 1) Modify the project to accept numbers Between 0 and
   // 100 inclusive
     for (int i = 0; i < arraySize; i++){

       // Prompt the user to enter the number
      System.out.print("Please, Enter the number for element:" +(i+1) + " ---> ");

        number = input.nextInt();

        // Use while statement to check for Invalid numbers
        // Invalid number is < 0 || number > 100

          while (number < 0 || number > 100){

            System.out.println("**** Invalid Number ****");

           System.out.print("Please enter a number between 0 and 100:"

                                   +(i+1) + " ---> ");

              number = input.nextInt();

          }

       numberArray[i] = number;
     }
```

```
      System.out.println("************************************");
    }

// 3) Define and code the printPopulatedArray() method
public static void printPopulatedArray(){
    System.out.println("\n The following is the Array populated with User Numbers");
      for (int k = 0; k < numberArray.length; k++ ){
         System.out.println(numberArray[k]);
      }
      System.out.println();
   }

  // 4) Define and code the sumAndAverageNumbersInArray() method
  public static void sumAndAverageNumbersInArray(){
   // Sum the number for the array
     int sum = 0;
      for ( int m = 0; m < numberArray.length; m++){
         sum += numberArray[m];
      }
   // Find the average by dividing the Sum to the number of elements in array
   // also Use (double) to cast average to double
   average = (double) sum / numberArray.length;
   System.out.println("The sum of the number of array = " + sum);
  System.out.printf("The Average of the number of array = %.2f %n", average);
  }
```

```
 // 5) Define and code the sortArrayAndPrintMaxAndMin() method
public static void sortArrayAndPrintMaxandMin(){
  // Sort the Array in Ascending Order so the smallest number will be in index 0 and
  // the highest number will be in index numberArray.length - 1
   Arrays.sort(numberArray);

   System.out.println("The Minimum number is: " + numberArray[0]);

   System.out.println("The Maximum number is: " +

                                          numberArray[numberArray.length - 1]);

  }

     // 6) Define and code the printFooters() method
    public static void printFooters(){
       System.out.println("\n********** Footers ********");
    }
}
```

// 2) Prompt the user if he/she wants to continue (Y/N)

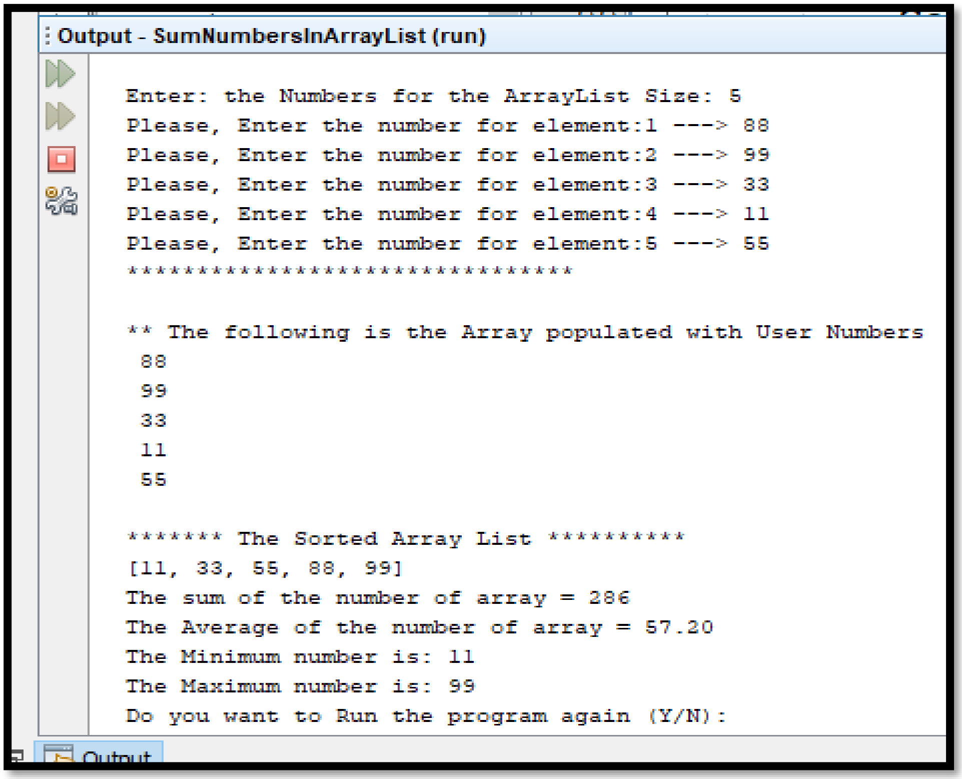
```
/*
 * This project will prompt the user to create an Array
and populate the array with numbers.
Programmer Ogar Haji
Date: 04/30/2020.
 */
// 1) Modify the project to accept numbers Between 0 and 100 inclusive
// 2) Prompt the user if he/she wants to continue (Y/N)
```

```
package sumnumbersinarray;
// Import the following which are needed in this project
import java.util.Arrays;
import java.util.Scanner;
public class SumNumbersInArray {
 public static int[] numberArray = new int[5];
// 1) Declare an object calle 'input'from Scanner class
public static Scanner input = new Scanner (System.in);
// 2) Declare variables as Global so they will be used across all the methods
public static int arrayLength;
public static int number;
int sum = 0;
public static double average;
public static String answer = "y";
   public static void main(String[] args) {
      // 1) Call the printHeaders() method to print the headings
      printHeaders();

       // 2) Use while statement to prompt the user to run the project again or not
      while (answer.equalsIgnoreCase("Y") ) {
     // 3) Call the promptUserForInput() method to prompt user to input numbers
        promptUserForInput();
       // 4) Call the printPopulatedArray() method to print the array populated by user
        printPopulatedArray();
       // 5) Call sumAndAverageNumberInArray() method to find sum & average
```

```
        sumAndAverageNumbersInArray();

       // 6) Call sortArrayAndPrintMaxandMin() method to find Max & Min numbers
        sortArrayAndPrintMaxandMin();

        input.nextLine(); // Read the End of line enter key

      System.out.print("Do you want to Run the program again (Y/N): ");

       answer = input.nextLine();

       System.out.println("The answer is: " + answer);

        // 7) Check if the user want to continue running the project
         if ( !(answer.equalsIgnoreCase("Y")) )  {

              break;

        }

       }

      printFooters();

   } // End of main() method

// 1) Define and code the printHeaders() method
   public static void printHeaders(){

      System.out.println("*********** Headings ********\n");

   }

   // 2) Define and code the promptUserForInput() method
   public static void promptUserForInput(){

      int arraySize = 5;

      System.out.println("The Array Size is Fixed to: " + arraySize );

      System.out.println("So Enter: " + arraySize + " Numbers Only.");
```

```
        for (int i = 0; i < arraySize; i++){
          // prompt the user to enter the number
         System.out.print("Please, Enter the number for element:" +(i+1) + " ---> ");
           number = input.nextInt();
           // Use while statement to check for Invalid numbers
           // Invalid number is < 0 || number > 100
           while (number < 0 || number > 100){
             System.out.println("**** Invalid Number ****");
            System.out.print("Please enter a number between 0 and 100:" +(i+1) + " ---> ");
               number = input.nextInt();
           }
          numberArray[i] = number;
        }
        System.out.println("**********************************");
     }
// 3) Define and code the printPopulatedArray() method
public static void printPopulatedArray(){
        System.out.println("\n** The following is the Array populated with User
Numbers");
        for (int k = 0; k < numberArray.length; k++ ){
           System.out.println(numberArray[k]);
        }
        System.out.println();
    }
  // 4) Define and code the sumAndAverageNumbersInArray() method
  public static void sumAndAverageNumbersInArray(){
   // Sum the values for the array
      int sum = 0;

      for ( int m = 0; m < numberArray.length; m++){

          sum += numberArray[m];

      }

   // Find the average by dividing the Sum to the number of elements in array
   // also Use (double) to cast average to double
   average = (double) sum / numberArray.length;

   System.out.println("The sum of the number of array = " + sum);

  System.out.printf("The Average of the number of array = %.2f %n", average);

  }

 // 5) Define and code the sortArrayAndPrintMaxAndMin() method
public static void sortArrayAndPrintMaxandMin(){
  // Sort the Array in Ascending Order so the smallest number will be in index 0 and
  // the highest number will be in index numberArray.length - 1
   Arrays.sort(numberArray);

   System.out.println("The Minimum number is: " + numberArray[0]);

   System.out.println("The Maximum number is: " +
                                          numberArray[numberArray.length - 1]);
  }


    // 6) Define and code the printFooters() method

    public static void printFooters(){

       System.out.println("\n********** Footers ********");

    }

}
```

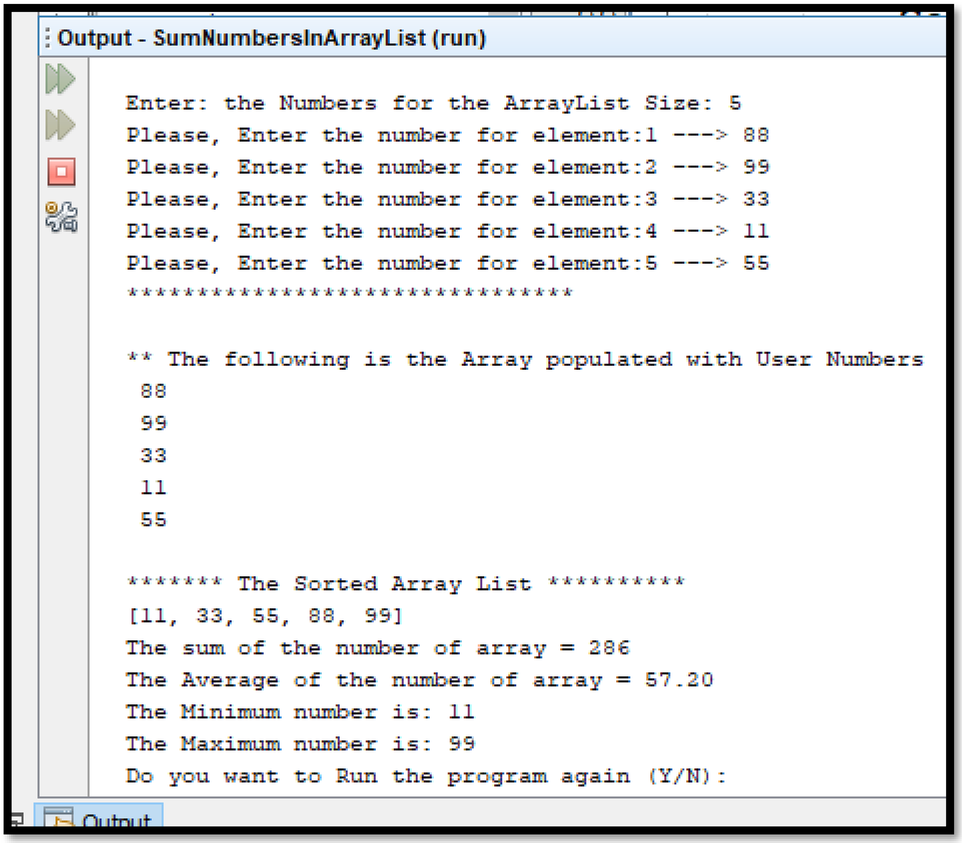
ArrayList Exercise



```
Enter: the Numbers for the ArrayList Size: 5
Please, Enter the number for element:1 ---> 88
Please, Enter the number for element:2 ---> 99
Please, Enter the number for element:3 ---> 33
Please, Enter the number for element:4 ---> 11
Please, Enter the number for element:5 ---> 55
********************************

** The following is the Array populated with User Numbers
 88
 99
 33
 11
 55

******* The Sorted Array List **********
[11, 33, 55, 88, 99]
The sum of the number of array = 286
The Average of the number of array = 57.20
The Minimum number is: 11
The Maximum number is: 99
Do you want to Run the program again (Y/N):
```

```
/*
 * This project will implement ArrayList collections
The project will prompt the user to create an ArrayList
and populate the ArrayList with numbers.
Programmer Ogar Haji
Date: 04/30/2020.
 */
// 1) Modify the project to accept numbers Between 0 and 100 inclusive
// 2) Prompt the user if he/she wants to continue (Y/N)
// 3) ARRAYLIST
```

```
/*
 * This project will prompt the user to create an Array
and populate the array with numbers.
Programmer Ogar Haji
Date: 04/30/2020.
 */
// 1) Modify the project to accept numbers Between 0 and 100 inclusive
// 2) Prompt the user if he/she wants to continue (Y/N)

package sumnumbersinarraylist;

// Import the following which are needed in this project
import java.util.Scanner;
import java.util.ArrayList;
import java.util.Collections;

public class SumNumbersInArrayList {

// Declare an ArrayList called numberArrayList of type Integer
 public static ArrayList <Integer> numberArrayList = new ArrayList <Integer>();
// 1) Declare an object called 'input' from Scanner class
public static Scanner input = new Scanner (System.in);

// 2) Declare variables as Global so they will be used across all the methods
public static int number;
int sum = 0;
public static double average;
public static String answer = "y";
public static void main(String[] args) {
   // 1) Call the printHeaders() method to print the headings
   printHeaders();

   // 2) Use while statement to check if the user want to run the project again
   while (answer.equalsIgnoreCase("Y") ) {

  // 3) Call the promptUserForInput() method to prompt user to input numbers
     promptUserForInput();

  // 4) Call the printPopulatedArray() method to print the populated array
     printPopulatedArray();

  // 5) Call sumAndAverageNumbersInArray() method to find sum & average
     sumAndAverageNumbersInArray();

  // 6) Call sortArrayAndPrintMaxandMin() method to sort the array

     sortArrayAndPrintMaxandMin();

     input.nextLine(); // Read the End of line enter key

   System.out.print("Do you want to Run the program again (Y/N): ");

      answer = input.nextLine();

      System.out.println("The answer is: " + answer);

   // 7) Check if the user wants to break from running the project

       if ( !(answer.equalsIgnoreCase("Y")) )  {

           break;

         }

    }

   // 8) Call the printFooters() method to print the Footers

   printFooters();
   } // End of main() method

// 1) Define and code the printHeaders() method
    public static void printHeaders(){
        System.out.println("*********** Headings ********\n");
    }
    // 2) Define and code the promptUserForInput() method
    public static void promptUserForInput(){
      // Declare the size of ArrayList and initialize it to 0
       int arrayListSize = 0;
        // To clear the ArrayList from the items
       numberArrayList.clear();

       System.out.println("The Array Size is Fixed to: " + arraySize );
       System.out.println("So Enter: " + arraySize + " Numbers Only.");
       for (int i = 0; i < arraySize; i++){
          // prompt the user to enter the number
        System.out.print("Please, Enter the number for element:" +(i+1) + " ---> ");
           number = input.nextInt();
           // Use while statement to check for Invalid numbers
           // Invalid number is < 0 || number > 100
           while (number < 0 || number > 100){
             System.out.println("**** Invalid Number ****");
            System.out.print("Please enter a number between 0 and 100:" +(i+1) + " ---> ");
               number = input.nextInt();
           }
        // Add the number entered by user to numberArrayList using add() method
         numberArrayList.add(number);
       }
       System.out.println("**********************************");
    }

// 3) Define and code the printPopulatedArray() method
public static void printPopulatedArray(){
       System.out.println("\n** The following is the Array populated with User
Numbers");
       for (int k = 0; k < numberArrayList.size(); k++ ){
          System.out.printf(" %d %n", numberArrayList.get(k));       }
       System.out.println();

   }

  // 4) Define and code the sumAndAverageNumbersInArray() method
  public static void sumAndAverageNumbersInArray(){
        // Sort the number ArrayList which is a Collection and print it out
       Collections.sort(numberArrayList);

        // Print out the sorted ArrayList
       System.out.println("******* The Sorted Array List **********");
     System.out.println(numberArrayList);

   // Sum the values for the array
      int sum = 0;
       for ( int m = 0; m < numberArrayList.size(); m++){
          sum += numberArrayList.get(m);
       }

   // Find the average by dividing the Sum to the number of elements in array
   // also Use (double) to cast average to double
   average = (double) sum / numberArrayList.size();

   System.out.println("The sum of the number of array = " + sum);

  System.out.printf("The Average of the number of array = %.2f %n", average);

  }

// 5) Define and code the sortArrayAndPrintMaxAndMin() method
public static void sortArrayAndPrintMaxandMin(){

  // Sort the numberArrayList in Ascending order
   Collections.sort(numberArrayList);

   // Print out the Smallest or Minimum number which is the first number index 0
   System.out.println("The Minimum number is: " + numberArrayList.get(0));

  // Find out the Last number or Maximum number which is the last number size() -1)
   int lastNumber = numberArrayList.get(numberArrayList.size()-1) ;

   System.out.println("The Maximum number is: " + lastNumber);   }

     // 6) Define and code the printFooters() method
    public static void printFooters(){

        System.out.println("\n********** Footers ********");

    }

}
```



```
Enter: the Numbers for the ArrayList Size: 5
Please, Enter the number for element:1 ---> 88
Please, Enter the number for element:2 ---> 99
Please, Enter the number for element:3 ---> 33
Please, Enter the number for element:4 ---> 11
Please, Enter the number for element:5 ---> 55
********************************

** The following is the Array populated with User Numbers
 88
 99
 33
 11
 55

******* The Sorted Array List **********
[11, 33, 55, 88, 99]
The sum of the number of array = 286
The Average of the number of array = 57.20
The Minimum number is: 11
The Maximum number is: 99
Do you want to Run the program again (Y/N):
```

Exception Handling:

Divide by Zero

+++(Do Lab Exercise 3) 100 Points+++

Do Lab Exercise 3

Lesson 245 Ex: How to use Exception Handling For Divide by Zero Exception in Java ?

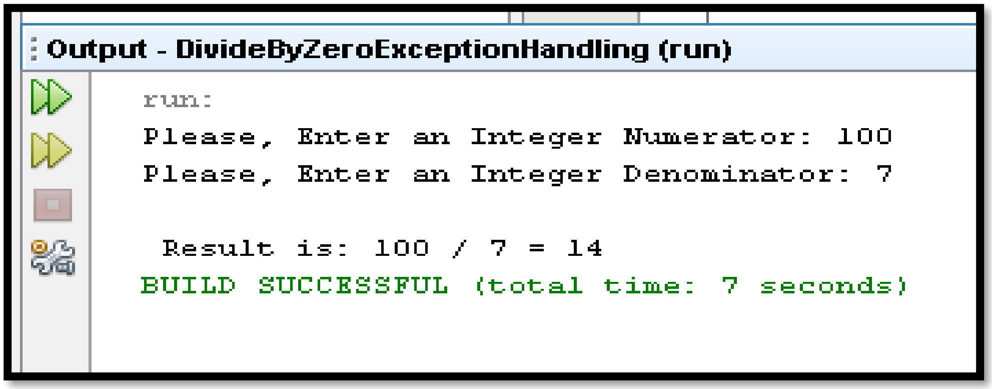
```
package dividebyzeroexceptionhandling;

// 1) Import the classes to be used in the project

import java.util.Scanner;

import java.util.InputMismatchException;

public class DivideByZeroExceptionHandling {

    public static void main(String[] args) {

        // 1) Create an object ‘input’ from class Scanner
        Scanner input = new Scanner (System.in);

        // 2) Declare all the variables to be used in the project
        int numerator, denominator, result;

        boolean continueLoop = true;

        // 3n) Use do while to loop until continue loop is false

        do {

            try {  // try is used to check for errors

                // 3) Prompt the user to enter an integer numerator
                System.out.print("Please, Enter an Integer Numerator: ");
    // 4) Get or read the interger entered by user
    numerator = input.nextInt();

    // 5) Prompt the user to enter an integer denominator
    System.out.print("Please, Enter an Integer Denominator: ");

    // 6) Get or read the integer entered by user

    denominator = input.nextInt();

    // 7) Find the quotient of the division of 2 numbers
    // by calling the method quotient()
    result = quotient (numerator, denominator);

    // 8) print out the division result
    System.out.printf("%n Result is: %d / %d = %d %n",
                      numerator, denominator, result);
    // 8n) Set contineLoop variable to false
    continueLoop = false;

}

catch (InputMismatchException ime){

    System.out.printf("%n Exception occured: %s %n", ime);

    input.nextLine(); // Discard the input so user will try again

    System.out.print("\n You Must enter an Integer for denominator");

}

catch (ArithmeticException ae) {

    System.out.printf("%n Exception: %s %n", ae);

    System.out.println("\n Zero is Invalid denominator"

                       + "Please try again");

}
```

```
            finally {
                System.out.println("Finally statement is alwyas executed");
                System.out.println("***** \"End of Project\"  ****");
            }
        } while (continueLoop == true);
    } // End of main method
    // 1) Define and code quotient() method to find the division
    public static int quotient (int n1, int n2){
        return (n1 / n2); // Possible division by zero exception
    }
}
```

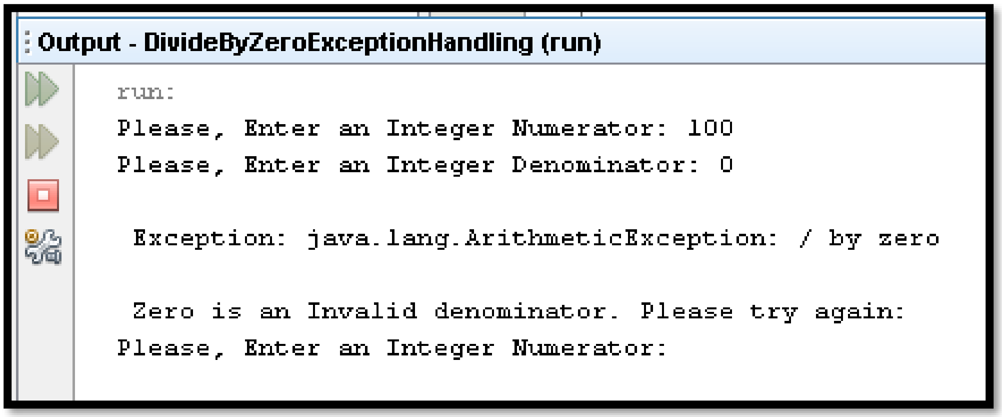
Chapter 12

Exception Handling: a Deeper Look

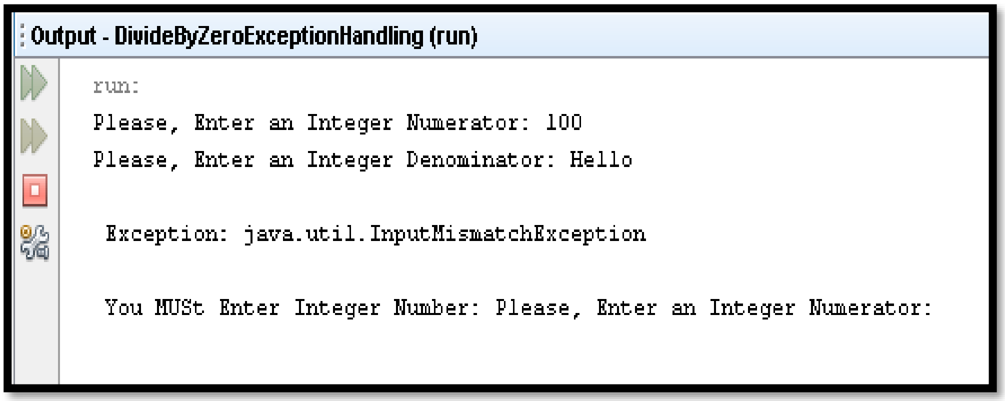
Lesson 246 Ex: How to use Exception Handling For Divide by Zero Exception in Java ?



```
run:
Please, Enter an Integer Numerator: 100
Please, Enter an Integer Denominator: 7

 Result is: 100 / 7 = 14
BUILD SUCCESSFUL (total time: 7 seconds)
```



```
run:
Please, Enter an Integer Numerator: 100
Please, Enter an Integer Denominator: 0

 Exception: java.lang.ArithmeticException: / by zero

 Zero is an Invalid denominator. Please try again:
Please, Enter an Integer Numerator:
```



```
run:
Please, Enter an Integer Numerator: 100
Please, Enter an Integer Denominator: Hello

 Exception: java.util.InputMismatchException

 You MUSt Enter Integer Number: Please, Enter an Integer Numerator:
```

1) **Save** the following Java project as **DivideByZeroExceptionHandling**

```
package dividebyzeroexceptionhandling;
```

```
/*
 * @author Ogar PC
 */

// Import Scanner Class to be used in the project
   import java.util.Scanner;

public class DivideByZeroExceptionHandling {

   public static void main(String[] args) {
      // 1) Create an object input from class Scanner
      Scanner input = new Scanner (System.in);

      // 2) Declare the variables to be used in the project
      int numerator, denominator, result;

      // 3) Prompt the user to enter an integer
      System.out.print ("Please, Enter an Integer Numerator: ");

      // 4) Get or Read the number entered
      numerator = input.nextInt();

      // 5) Prompt the user to enter an integer Denominator
      System.out.print ("Please, Enter an Integer Denominator: ");

      // 6) Get or Read the Denominator number entered
      denominator = input.nextInt();
```

```
        // 7) Find the quotient of the division of 2 numbers
        // by calling quotient() method
        result = quotient (numerator, denominator);


        // 8) Print out the result
        System.out.printf("%n Result is: %d / %d = %d %n",
                          numerator, denominator, result);
    }

    // 1) Define and code quotient() method to find the quotient
    public static int quotient (int numerator, int denominator){
        // 2) Find the quotient and return it to calling method
        return (numerator / denominator ); // possible division by zero
    }
}
```

Modify the project by adding Exception Handling:

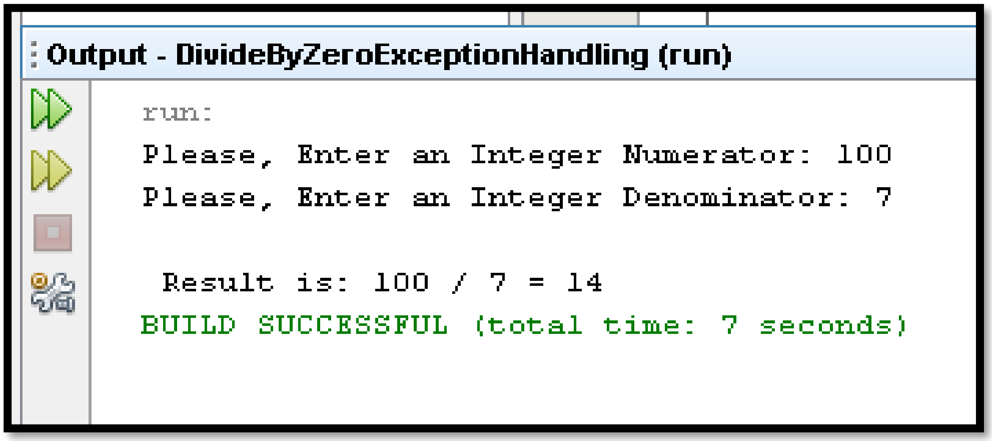
```
package dividebyzeroexceptionhandling;


/*
 * @author Ogar PC
 */


// Import Scanner Class to be used in the project
    import java.util.Scanner;
```

```
// 2) import the input mismatch exception
   import java.util.InputMismatchException;

public class DivideByZeroExceptionHandling {

   public static void main(String[] args) {
      // 1) Create an object input from class Scanner
      Scanner input = new Scanner (System.in);

      // 2) Declare the variables to be used in the project
      int numerator, denominator, result;
      boolean continueLoop = true;  // declare the variable to
                                    // continue looping

       // 3n) Use do … while to loop until continueLoop is false
       do {
          try {  // Use try method to check for errors
           // 3) Prompt the user to enter an integer
           System.out.print ("Please, Enter an Integer Numerator: ");

           // 4) Get or Read the number entered
           numerator = input.nextInt();

           // 5) Prompt the user to enter an integer
           System.out.print ("Please, Enter an Integer Denominator: ");
```

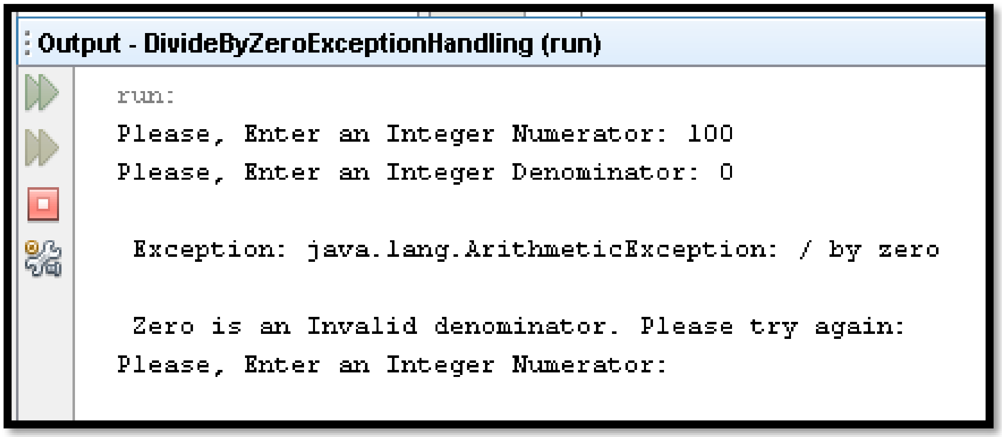
```
        // 6) Get or Read the number entered
        denominator = input.nextInt();

        // 7) Find the quotient of the division of 2 numbers
        // by calling quotient() method
        result = quotient (numerator, denominator);

        // 8) Print out the result
        System.out.printf("%n Result is: %d / %d = %d %n",
                          numerator, denominator, result);

          // 8n) Set continueLoop variable to false
          continueLoop = false;
      }
      catch (InputMismatchException   ime ) {
        System.out.printf("%n Exception: %s %n", ime);
       input.nextLine(); // Discard the input so user will try again
      System.out.print("\n You Must Enter Integer Number: ");
      }

      catch (ArithmeticException  ae ) {
         System.out.printf("%n Exception: %s %n", ae);
        System.out.println("\n Zero is an Invalid denominator. "
                 + "Please try again: ");
```

```
            }
             finally {
               System.out.println("Finally statement will always execute.");
            }
        } while (continueLoop == true);
     }

   // 1) Define and code quotient() method to find the quotient
   public static int quotient (int numerator, int denominator){
      // 2) Find the quotient and return it to calling method
      return (numerator / denominator ); // possible division by zero
   }
}
```

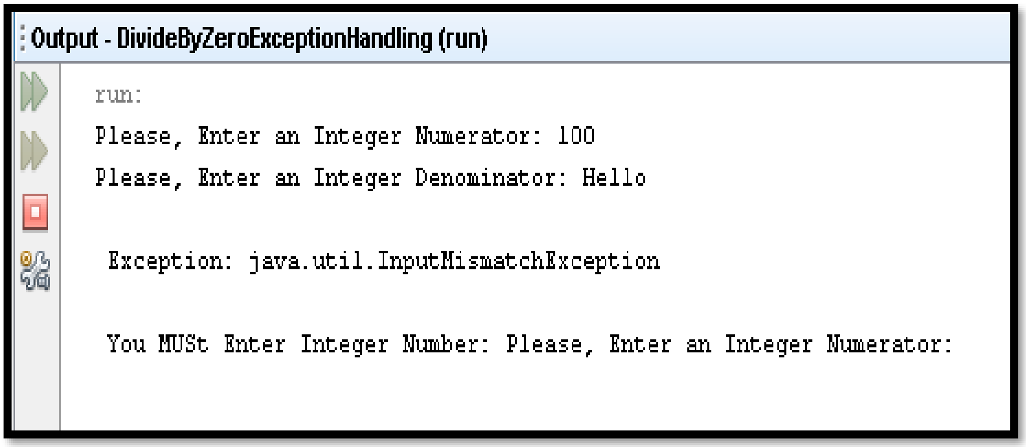
The Output will look like the following:



```
run:
Please, Enter an Integer Numerator: 100
Please, Enter an Integer Denominator: 7

 Result is: 100 / 7 = 14
BUILD SUCCESSFUL (total time: 7 seconds)
```



```
run:
Please, Enter an Integer Numerator: 100
Please, Enter an Integer Denominator: 0

 Exception: java.lang.ArithmeticException: / by zero

 Zero is an Invalid denominator. Please try again:
Please, Enter an Integer Numerator:
```

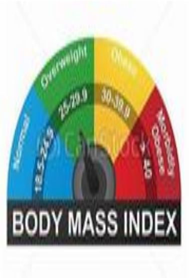


```
run:
Please, Enter an Integer Numerator: 100
Please, Enter an Integer Denominator: Hello

 Exception: java.util.InputMismatchException

 You MUSt Enter Integer Number: Please, Enter an Integer Numerator:
```

CIS 144 Java Language Instructor:

Ogar Haji

Please, Read, Study and Practice the Lessons in the Java Handout



Create a JFrame Form to: Calculate Body Mass Index (BMI) of a Person Java Project

To calculate BMI = weight * 703 / (height * height);

+++(Do Lab Exercise 4) 100 Points+++

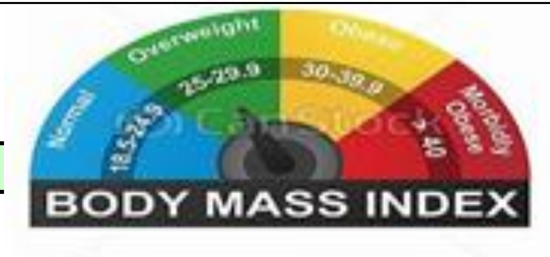
Do Lab Exercise 4

Lesson 247 - Ex-How to Calculate the BMI (Body Mass Index) in a person using Java Language?

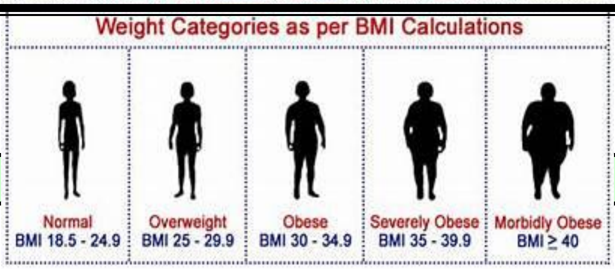
1) **Do the 12 Must Steps to Design, Code and Solve a problem using Java Language. You may look in the Instructor's Java Handout**

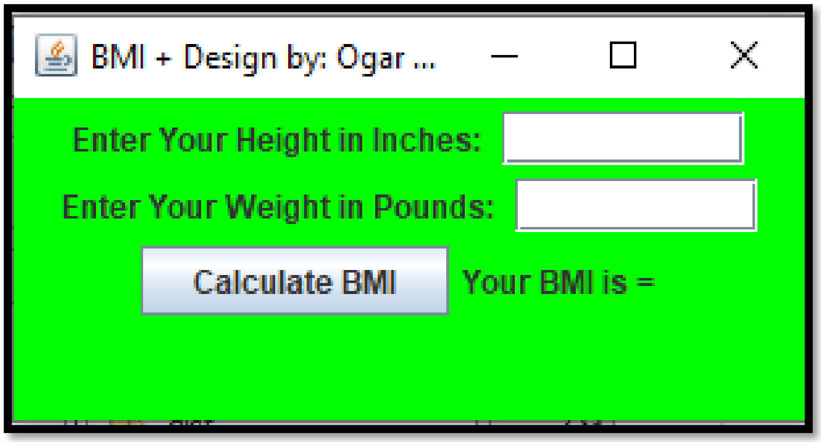
Do Steps 1 thru 7 in your Note Book or on Paper.

```
/*
This project calculates BMI (Body Mass Index)
The program prompts the user to input the First and Last Name.
It also prompts the user to input the Weight in Pounds and Height in
Feet and then Height again in Inches. It calculates the BMI and
prints out whether the person is Normal or Overweight.
To calculate BMI = weight * 703 / (height * height);
If the calculated BMI is < 25, then the person is Normal and if the
calculated BMI is > 25, then the person is Overweight.
Programmer: Ogar Haji - Java Instructor
Date-Written: 01/25/2019
*/
```



ing Language.

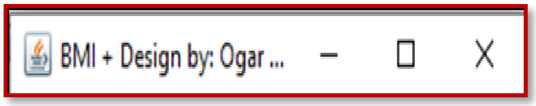



BMI + Design by: Ogar ...
Enter Your Height in Inches:
Enter Your Weight in Pounds:
Calculate BMI
Your BMI is =

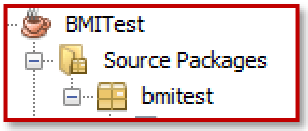
1) Get into **NetBeans** type the following Java project and

2) Save this Java project as **BMITest**. **(Create Project BMITest without the main() method)**

```
package bmitest;

/**
 * @author Ogar's Laptop
 */
/*************************************************************
//  BMITest Sets up a GUI to calculate body mass index.
//**********************************************************/
import javax.swing.JFrame;

public class BMITest {
   //  Creates and displays the BMI JFrame Form.
   public static void main (String[] args)  {
```



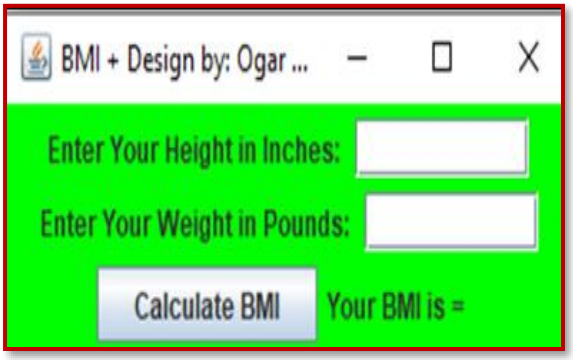
```
      JFrame frame = new JFrame("BMI + Design by: Ogar Haji");
      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      BMIPanel panel = new BMIPanel();
      frame.getContentPane().add(panel);
      frame.pack();
      frame.setVisible(true);

   }

}
```

1) RC on  **"bmitest" under Source Package**

2) Point to New
3) Click on JPanelForm to display the JPanel Form
4) Click on Source tab to display the code
5) Type the following Java code as shown below:

```
package bmitest;

//************************************************************

//  BMIPanel.java

//  Computes body mass index in a GUI.

//************************************************************

import java.awt.*;

import java.awt.event.*;

import javax.swing.*;

public class BMIPanel extends JPanel  {

   // Declare the Global Variables to be used in this project

   private final int WIDTH = 300;

   private final int HEIGHT = 120;

   private final JLabel heightLabel;
```

```
 private final JLabel weightLabel;
 private final JLabel BMILabel;
 private final JLabel resultLabel;
 private final JTextField heightTextField;
 private final JTextField weightTextField;
 private final JButton calculateButton;

//  Sets up the GUI JForm.
public BMIPanel() {
   //create labels for the height and weight textfields
   heightLabel = new JLabel ("Enter Your Height in Inches: ");
   weightLabel = new JLabel ("Enter Your Weight in Pounds: ");
```


BMI + Design by: Ogar ...
Enter Your Height in Inches:
Enter Your Weight in Pounds:
Calculate BMI
Your BMI is =

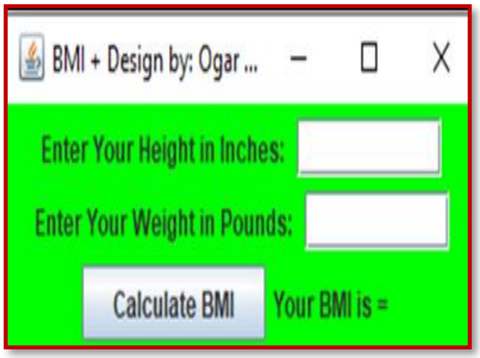
```
   //create a Label with the Text  "Your BMI is = "
   BMILabel = new JLabel ("Your BMI is = ");

   //create a result label to hold the BMI value
   resultLabel = new JLabel ();
```

BMI + Design by: Ogar ...
Enter Your Height in Inches:
Enter Your Weight in Pounds:
Calculate BMI
Your BMI is =

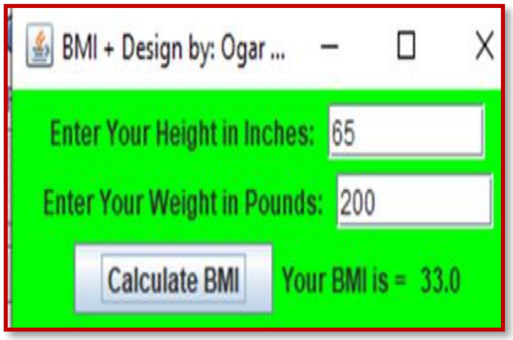
```
   //create a JTextField to hold the person's height in inches
   heightTextField = new JTextField (8);

   //create a JTextField to hold the person's weight in pounds
   weightTextField = new JTextField (8);

   //create a button to press to calculate BMI
   calculateButton = new JButton(" Calculate BMI ");
```

BMI + Design by: Ogar ...
Enter Your Height in Inches:
Enter Your Weight in Pounds:
Calculate BMI
Your BMI is =

```
   //create a BMI Listener and make it listen for button to be pressed
   calculateButton.addActionListener (new BMIButtonListener());

   //add the height label and height text field to the panel
   add (heightLabel);
   add (heightTextField);

   //add the weight label and weight text field to the panel
   add (weightLabel);
   add (weightTextField);

   //add the Calculate button to the panel
   add (calculateButton);

   //add the BMI label to the panel
   add (BMILabel);

   //add the result label that holds the result to the panel
   add (resultLabel);

   //set the size of the panel to the WIDTH and HEIGHT constants
   setPreferredSize (new Dimension(WIDTH, HEIGHT));

   //set the color of the panel to green Color
   setBackground (Color.green);
}
//  Represents an action listener for the calculate button.
private class BMIButtonListener implements ActionListener {
```


BMI + Design by: Ogar ...
Enter Your Height in Inches:
Enter Your Weight in Pounds:
Calculate BMI
Your BMI is =

```
    //  Compute the BMI when the button is pressed
    @Override
    public void actionPerformed (ActionEvent event) {
      String heightText, weightText;
      int heightValue, weightValue;
      double bmi;

     //get the text from the height and weight text fields
      heightText = heightTextField.getText();
      weightText = weightTextField.getText();
```

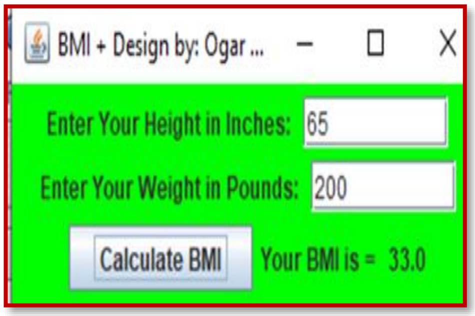


Enter Your Height in Inches: 65

Enter Your Weight in Pounds: 200

Your BMI is = 33.0

```
      //Use Integer.parseInt to convert the text to integer values
      heightValue = Integer.parseInt(heightText);
      weightValue = Integer.parseInt(weightText);

      //Calculate  bmi = 703 * weight in pounds / (height in inches)^2
    bmi = (weightValue * 703 )  / (heightValue * heightValue) ;

 //Put result in result label, use Double.toString to convert double to string.
      String result = Double.toString(bmi);
      resultLabel.setText (result);
    }
  }
}
```

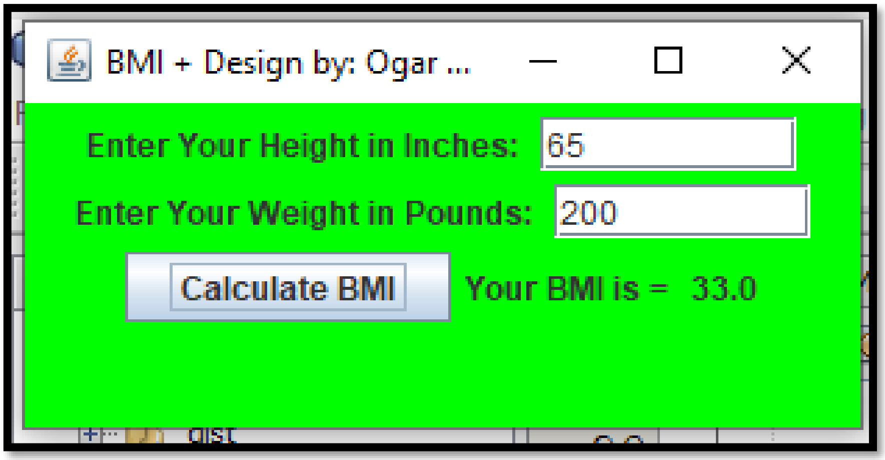


Enter Your Height in Inches: 65

Enter Your Weight in Pounds: 200

Your BMI is = 33.0

1)Run the project in NetBeans to display the following Java Form



BMI + Design by: Ogar ...

Enter Your Height in Inches: 65

Enter Your Weight in Pounds: 200

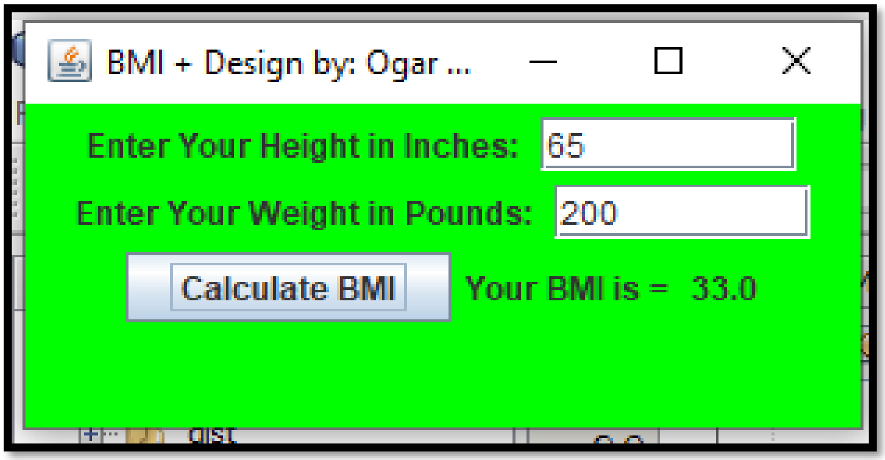
Your BMI is = 33.0

2)Type 65 or your height in Inches in the Height Text Field

3)Type 200 or your weight in the Weight Text Field

4)Click on Calculate BMI Button to display the Result 33.0 in the Result Label

<u>The output will appear like the following:</u>



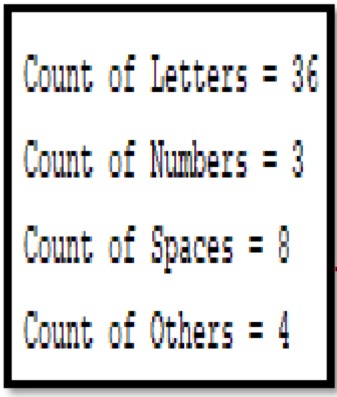
BMI + Design by: Ogar ...

Enter Your Height in Inches: 65

Enter Your Weight in Pounds: 200

Your BMI is = 33.0

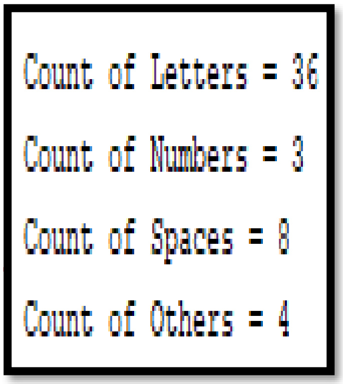
Please, Read, Study and Practice the Lessons in the Java Handout



```
Count of Letters = 36
Count of Numbers = 3
Count of Spaces = 8
Count of Others = 4
```

Count the Number of Letters, Numbers and Others in a String Text +(Do Lab Exercise 5) 100 Points

Do Lab Exercise 6



```
Count of Letters = 36
Count of Numbers = 3
Count of Spaces = 8
Count of Others = 4
```

Lesson 248 Ex: How to Count the Number of Letters, Numbers and Others in a Text String in Java?

This Java project will Count the number of Letters, Numbers and Others in a Text String.

1)The method **Character.isLetter(i)** is used to check if the character at **position** ‘**i**’ is a **Letter** (**a thru z**) either Uppercase or lowercase, **then increment lettersCount** by **1**.

```
if (Character.isLetter (ch[i]) ) {
            lettersCount ++ ;
```

2)The method **Character.isDigit(i)** is used to check if the character at **position ‘i’** is a **Digit (0 thru 9),** then **increment numbersCount** by **1**.

```
if (Character.isDigit (ch[i]) ) {
            numberssCount ++ ;
```

3)The method **Character.isWhitespace(i)** is used to check if the character at **position ‘i’** is a **Whitespace**(**space**), then **increment spacessCount** by **1**.

```
if (Character.isWhitespace (ch[i]) ) {
            spacesCount ++ ;
```

The steps to follow to solve the Java Project Count Letters, Numbers, Spaces and Others:

1) Declare a Fixed **text String** and initialize it as shown below:

```
String text = "Count number of Letters, Numbers, and Others 123: ?";
```

2) Call the **countLettersNumbersOthers(text)** and pass the 'text' parameter to it.

```
countLettersNumbersOthers (text);
```

3) Declare an **Array** called **'ch'** of type char and store the String '**Text'** in the Array ch

```
char[] ch = text.toCharArray();
```

4) Initialize the Letters, Numbers, Spaces and Others **counters to 0**

```
int lettersCount = 0;
```

5) Use **for** statement to **traverse** the length of the entire **array 'ch'** which contains the String text in it.

```
for(int i = 0; i < ch.length; i++) {

    if (Character.isLetter (ch[i]) ) {

            lettersCount ++ ;

    }
```

6) At last **print out** the various counters **(lettersCount, numbersCount** and so on).

```
System.out.println ("Count of Letters = " + lettersCount);
```

1) Type the following Java Project in NetBeans IDE and

Do Lab Exercise

2) Save the Java Project As CountLettersNumbersAndSpaces

```
package countlettersnumbersandspaces;
public class CountLettersNumbersAndSpaces {
public static void main(String[] args) {
   String text = "Count number of Letters, Numbers, and Others 123: ?";
   countLettersNumbersOthers (text);
} // End of main() method
public static void countLettersNumbersOthers(String text) {
    char[] ch = text.toCharArray(); // Declare an Array 'ch' of type char
    int lettersCount = 0;
    int numbersCount = 0;
    int spacesCount = 0;
    int othersCount = 0;
    for(int i = 0; i < text.length(); i++) {
       if (Character.isLetter (ch[i]) ){
              lettersCount ++ ;
       }
       else if (Character.isDigit (ch[i]) ) {
              numbersCount ++ ;
       }
       else if (Character.isWhitespace (ch[i]) ) {
              spacesCount ++ ;
```

```
        }
        else {
              othersCount ++;
        }
     }

     System.out.printf ("The Text String is: %s %n %n", text);

     System.out.println ("Count of Letters = " + lettersCount);
     System.out.println ("Count of Numbers = " + numbersCount);
     System.out.println ("Count of Spaces = " + spacesCount);
     System.out.println ("Count of Others = " + othersCount);
   }
}
```

As you see, I did Not use a good programming practice to code the above Java program. Now, I want you to Modify the project so it will look a Well Coded Java program and very documented.

1) **Use while statement to prompt the User "Do you Want to Continue Running the program: (Y/N)".**
2) **Add Comments to the project.**
3) **Add the following methods to this projects:**
   a. **A method to print the Headings**
   b. **A method to Prompt the User to Input the String Text**
   c. **A method to Count the Letters, Numbers, Spaces and Others**
   d. **A method to print the Footers**

Do Only 1 Modification at a Time

3 6 9

12 15

18 21

Print Numbers Between 1 to 100 which are Divisible by 3, by 5 and by both 3 and 5 by using % operator +++(Do Lab Exercise 6) 100 Points+++

5 10

15 20

25 30

Do Lab Exercise 6

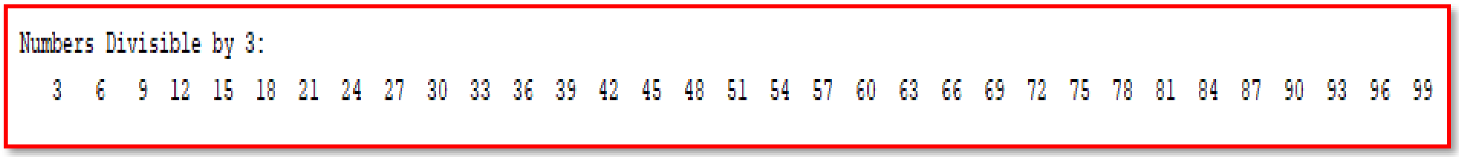
Lesson 249 Ex: How to Print Numbers Between 1 and 100 which are Divisible by 3 by 5 and by Both 3 and 5 ?

Modulus % operator finds the **Remainder** of **Division of 2 Numbers. Modulus % operator also is used to find if a Number is Divisible by 3, 5, 7 or any number if** the **Remainder** of **Division is Zero means that the number is divisible.**

You can use the **% Modulus operator** to **find** if a **Number** is **Odd** or **Even.**
If Remainder of the **Modulus % division by 2** is **0,** then the **Number** is **Even.**
If Remainder of the **Modulus % division by 2** is **1,** then the **Number** is **Odd.**

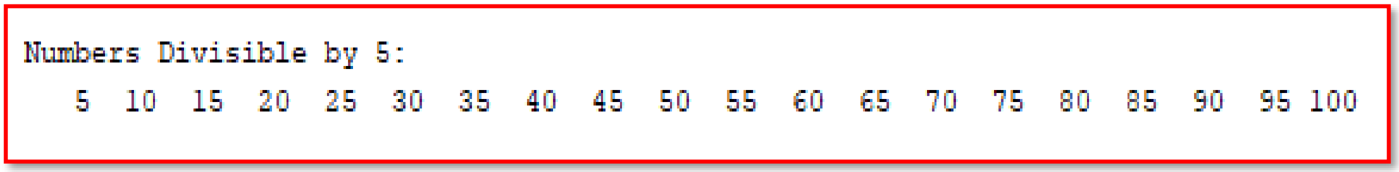
This Java program will print Numbers between 1 to 100 which are Divisible by 3, 5 and by both (3 and 5).

Numbers Divisible by 3 are: 3 % 3 equals 1 and Reminder = 0



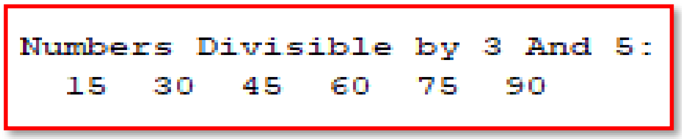
```
Numbers Divisible by 3:
   3   6   9  12  15  18  21  24  27  30  33  36  39  42  45  48  51  54  57  60  63  66  69  72  75  78  81  84  87  90  93  96  99
```

Numbers Divisible by 5 are: 10 % 5 equals 2 and Reminder = 0



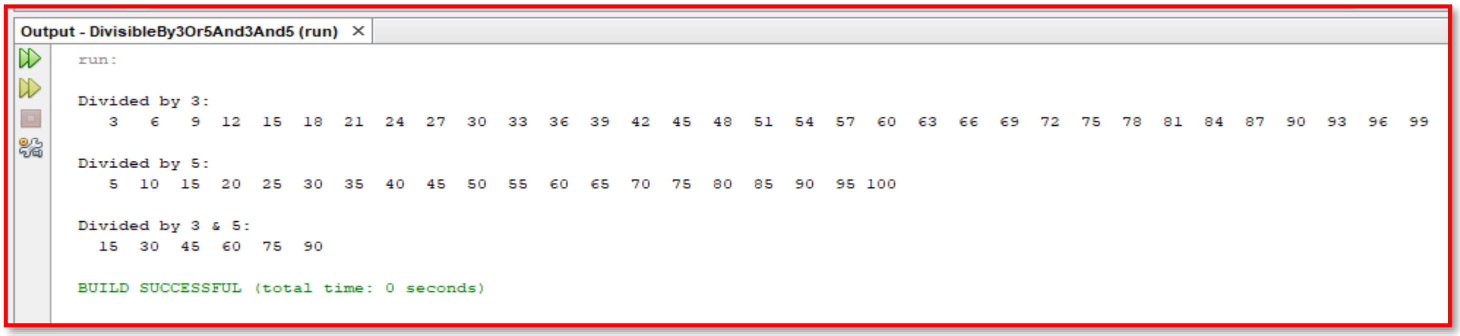
```
Numbers Divisible by 5:
   5  10  15  20  25  30  35  40  45  50  55  60  65  70  75  80  85  90  95 100
```

Numbers Divisible by 3 And 5 are: 15 % 3 equals 5 and Reminder = 0, And 15 % 5 equals 3 and Reminder = 0 And



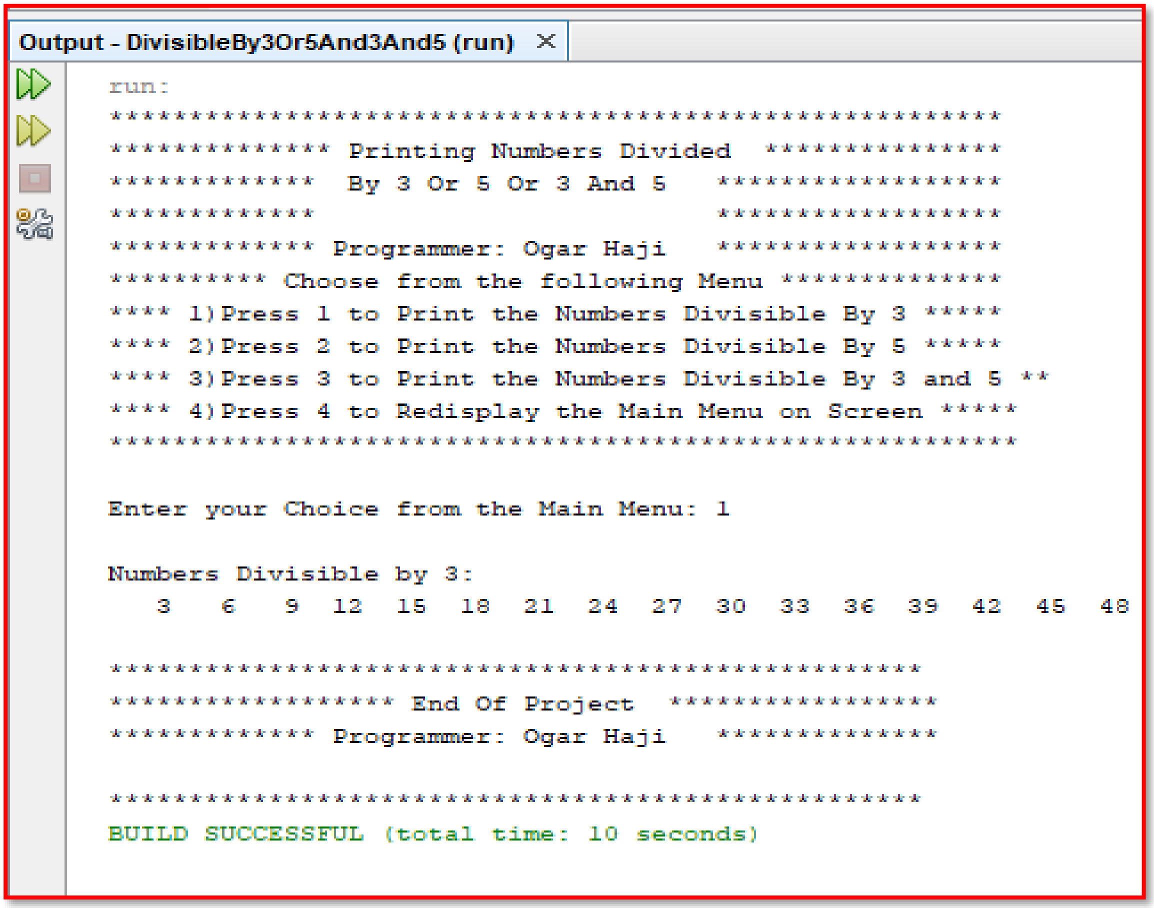
```
Numbers Divisible by 3 And 5:
   15   30   45   60   75   90
```

The Complete Output of the project is shown below:



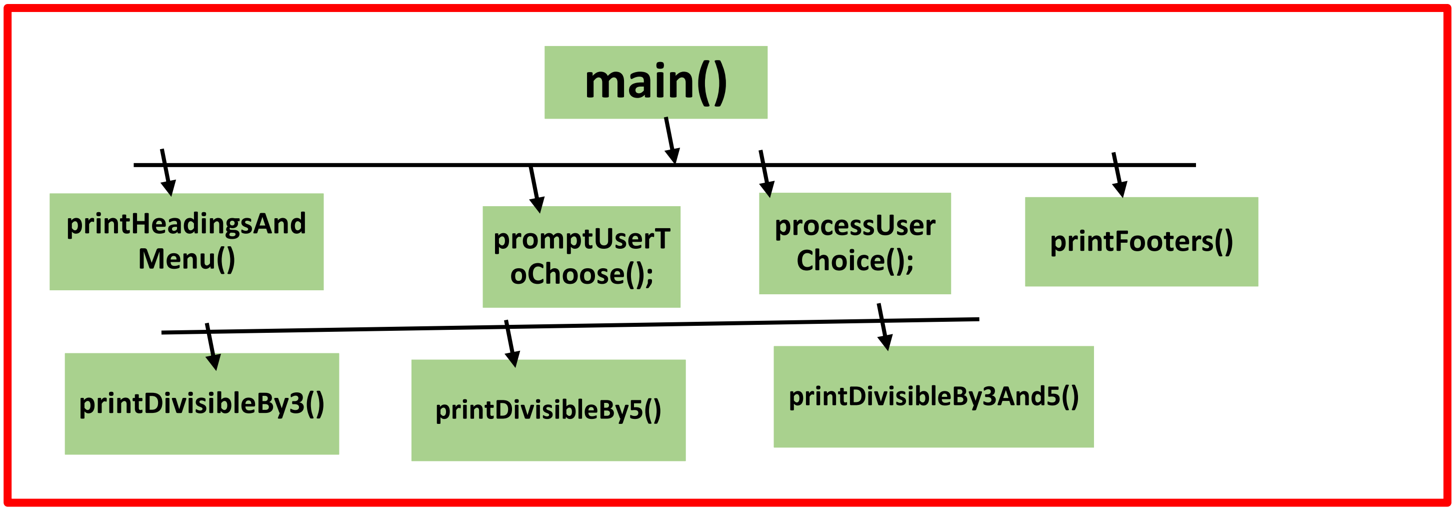
```
run:

Divided by 3:
   3   6   9  12  15  18  21  24  27  30  33  36  39  42  45  48  51  54  57  60  63  66  69  72  75  78  81  84  87  90  93  96  99

Divided by 5:
   5  10  15  20  25  30  35  40  45  50  55  60  65  70  75  80  85  90  95 100

Divided by 3 & 5:
  15  30  45  60  75  90

BUILD SUCCESSFUL (total time: 0 seconds)
```

The Output of the project with choice of 1 (Divisible by 3) is shown below:



```
run:
*********************************************************
************** Printing Numbers Divided  ***************
*************  By 3 Or 5 Or 3 And 5   ******************
*************                         ******************
************* Programmer: Ogar Haji   ******************
********** Choose from the following Menu **************
**** 1)Press 1 to Print the Numbers Divisible By 3 *****
**** 2)Press 2 to Print the Numbers Divisible By 5 *****
**** 3)Press 3 to Print the Numbers Divisible By 3 and 5 **
**** 4)Press 4 to Redisplay the Main Menu on Screen *****
**********************************************************

Enter your Choice from the Main Menu: 1

Numbers Divisible by 3:
   3   6   9  12  15  18  21  24  27  30  33  36  39  42  45  48

****************************************************
****************** End Of Project  *****************
************* Programmer: Ogar Haji   **************

****************************************************
BUILD SUCCESSFUL (total time: 10 seconds)
```

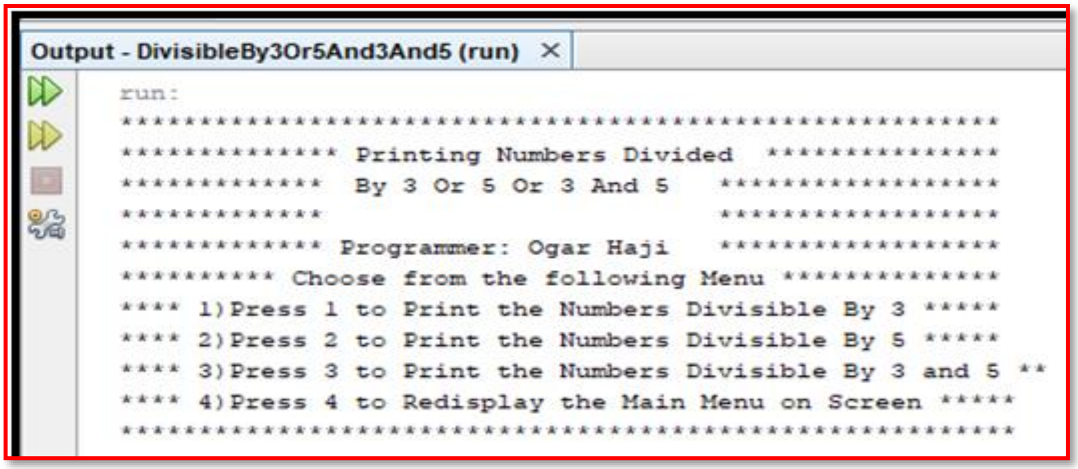
The Structure Flowchart of the Methods used in this Java Project:



The Java Pseudocode for the Java Project:

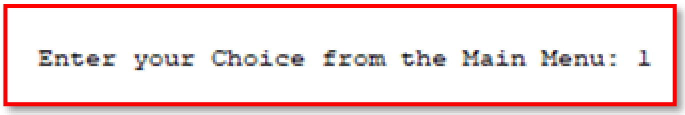
From the main() method you will call the following methods:

1) Call the printHeadingsAndMenu() method to print the Headings and Main Menu as shown below:



```
run:
********************************************************
************** Printing Numbers Divided  ***************
*************  By 3 Or 5 Or 3 And 5   ******************
*************                         ******************
************* Programmer: Ogar Haji   ******************
********** Choose from the following Menu **************
**** 1)Press 1 to Print the Numbers Divisible By 3 *****
**** 2)Press 2 to Print the Numbers Divisible By 5 *****
**** 3)Press 3 to Print the Numbers Divisible By 3 and 5 **
**** 4)Press 4 to Redisplay the Main Menu on Screen *****
*********************************************************
```

2) Call the promptUserToChoose() method to prompt the user to Enter a Choice (1, 2, 3, or 4) as shown below:



```
Enter your Choice from the Main Menu: 1
```

// Prompt user to Enter a Choice and Read and store it in choice variable

```
System.out.print("Enter your Choice from the Main Menu: ");
choice = input.nextInt();
```

Then use while statement to check if the User Choice is Valid (1, 2, 3, or 4) and Invalid, prompt again for the user choice.

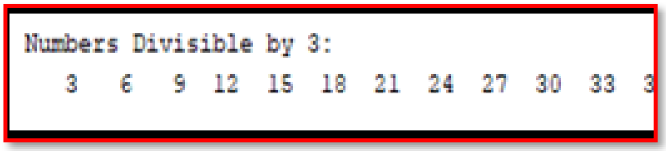
```
// Use while to check the user entered correct choices: 1, 2, 3 and 4
while ((choice != 1) && (choice != 2)
      && (choice != 3) && (choice != 4)) {
   System.out.println("** Invalid Choice + Please Choose options 1, 2, or 3");
   printHeadingsAndMenu();      // Call printHeadingsAndMenu() method

   // Prompt user to Enter a Choice again and Read and store it in choice variable
   System.out.print("Enter your Choice from the Main Menu: ");
   choice = input.nextInt();
```

3) **Call the processUserChice() method to process the user choice using switch….case statement as shown below:**

```
 switch (choice) {
     case 1:
          printDivisibleBy3();  // a) Call printDivisibleBy3()method
          break;
```

And the following is the printDivisibleBy3() method:

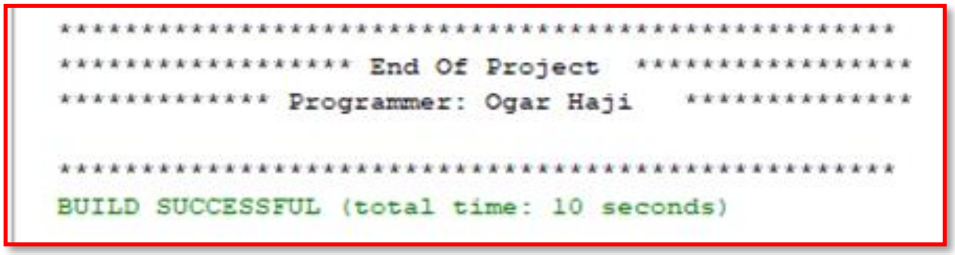
```
public static void printDivisibleBy3() {
   System.out.println("\nNumbers Divisible by 3: ");

   for (int i = 1; i <= 100; i++) {
       if (i % 3 == 0) {
           System.out.printf("%4d", i);
       }
   }
}
```



```
Numbers Divisible by 3:
   3   6   9  12  15  18  21  24  27  30  33  3
```

And the same thing with choice case 2 and case 3.

4) **Call the printFooters() method to print the Footers and End of Project as shown below:**



```
***************************************************
******************  End Of Project  *****************
*************  Programmer: Ogar Haji    **************

***************************************************
BUILD SUCCESSFUL (total time: 10 seconds)
```

5) **Get** into **NetBeans** Java Code Editor

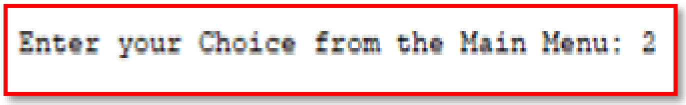
Do Lab Exercise

6) **Type** the following Java project, **save as** **DivisibleBy3Or5And3And5**

```
/*
This Java project will print the Numbers from 1 to 100 that
are divisible by 3 or 5 or 3 And 5.
Programmer: Ogar Haji (Java Instructor)
Date: 04/01/2020
*/
package divisibleby3or5and3and5;

// import the Classes needed in the project
import java.util.Scanner;

public class DivisibleBy3Or5And3And5 {

   // Declare Global variables to be used in the Entire project
   public static int choice;

   public static void main(String args[]) {

      // 1) Call the printHeadings and Menu() method
      printHeadingsAndMenu();
```

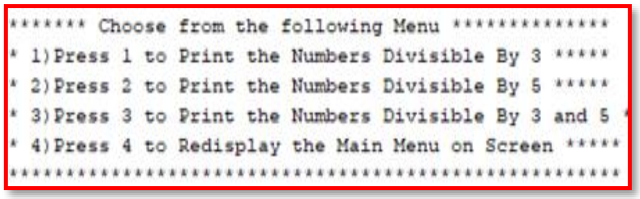
```
     // 2) Call the prompt User To Choose() method and receive choice
     choice = promptUserToChoose();

     // 3) Call the process UserChoic() method passing choice as parameters
     processUserChoice(choice);

      // 4) Call the printFooters() method
     printFooters();

 } // End of main()method

  // 1) Define and code printHeadingsAndMenu() method
  public static void printHeadingsAndMenu() {
     System.out.println("***********************************");
     System.out.println("** Printing Numbers Divided  ********");
     System.out.println("***  By 3 Or 5 Or 3 And 5   **********");
     System.out.println("*****                       **************");
     System.out.println("***** Programmer: Ogar Haji   *******");
     System.out.println("** Choose from the following Menu ****");
     System.out.println("** 1)Press 1 to Print the Numbers Divisible By 3 **");
     System.out.println("** 2)Press 2 to Print the Numbers Divisible By 5 **");
     System.out.println("** 3)Press 3 to Print the Numbers Divisible By 3 and 5 *");
     System.out.println("** 4)Press 4 to Redisplay the Main Menu on Screen**");
     System.out.println("***************************************\n");
}
```

```
// 2) Define and code promptUserToChoose() method
public static int promptUserToChoose() {
    // Create an object 'input' from the Scanner class
   Scanner input = new Scanner(System.in);

  // Prompt the user to Enter a Choice and Read and store it in choice variable
   System.out.print("Enter your Choice from the Main Menu: ");
   choice = input.nextInt();

   // Use while to check the user entered correct choices: 1, 2, 3 and 4
   while ((choice != 1) && (choice != 2)
          && (choice != 3) && (choice != 4)) {
      System.out.println("** Invalid Choice + Please Choose options 1, 2, or 3");
      printHeadingsAndMenu();      // Call printHeadingsAndMenu() method
      // Prompt user to Enter a Choice again and Read and store it in choice variable
      System.out.print("Enter your Choice from the Main Menu: ");
      choice = input.nextInt();
   }
  return choice;
}
```



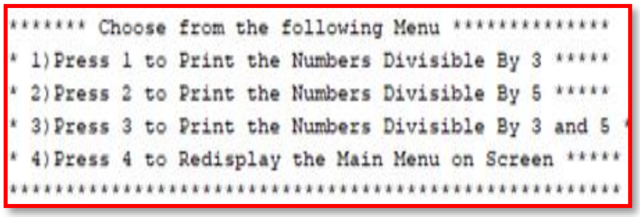
```
Enter your Choice from the Main Menu: 2
```

```
// 3) Define and code processUserChoice() method with argument choice
public static void processUserChoice(int choice) {
  // Check the choice entered by user using switch…case statements
   switch (choice) {

      case 1:

         printDivisibleBy3();  // a) Call printDivisibleBy3()method

         break;
```



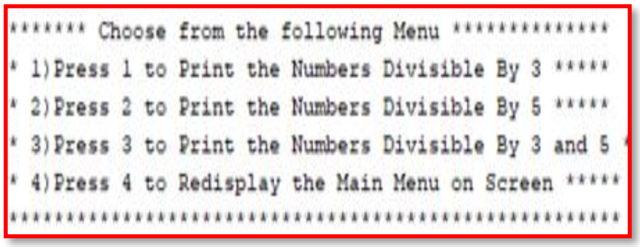
```
******* Choose from the following Menu **************
* 1)Press 1 to Print the Numbers Divisible By 3 *****
* 2)Press 2 to Print the Numbers Divisible By 5 *****
* 3)Press 3 to Print the Numbers Divisible By 3 and 5 *
* 4)Press 4 to Redisplay the Main Menu on Screen *****
*****************************************************
```

```
    case 2:
      printDivisibleBy5();  // b) Call printDivisibleBy5()method
      break;
```



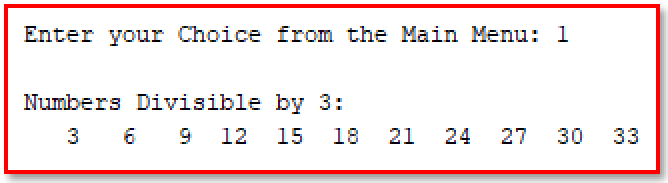
```
******* Choose from the following Menu **************
* 1)Press 1 to Print the Numbers Divisible By 3 *****
* 2)Press 2 to Print the Numbers Divisible By 5 *****
* 3)Press 3 to Print the Numbers Divisible By 3 and 5 *
* 4)Press 4 to Redisplay the Main Menu on Screen *****
*****************************************************
```

```
    case 3:
      printDivisibleBy3And5();  // c) Call printDivisibleBy3And5()method
      break;

    case 4:
      printHeadingsAndMenu(); // d) Call printHeadingsAndMenu()method
      break;

    default:
   System.out.println("** Invalid Choice+ Please Choose options 1, 2, or 3");
      break;
   }
}
```



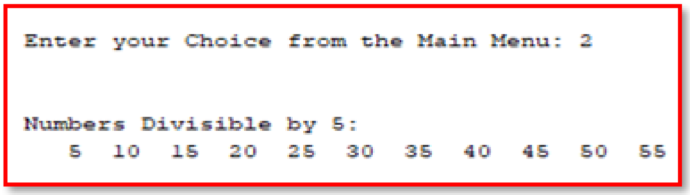
```
******* Choose from the following Menu **************
* 1)Press 1 to Print the Numbers Divisible By 3 *****
* 2)Press 2 to Print the Numbers Divisible By 5 *****
* 3)Press 3 to Print the Numbers Divisible By 3 and 5 *
* 4)Press 4 to Redisplay the Main Menu on Screen *****
*****************************************************
```

```
// a) Define and code printDivisibleBy3()method
public static void printDivisibleBy3() {
  System.out.println("\nNumbers Divisible by 3: ");

  for (int i = 1; i <= 100; i++) {
     if (i % 3 == 0) {
        System.out.printf("%4d", i);
     }
  }
}
```



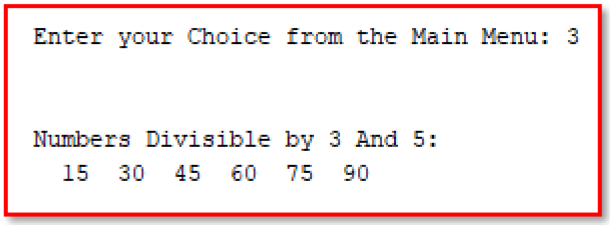
```
Enter your Choice from the Main Menu: 1

Numbers Divisible by 3:
   3   6   9  12  15  18  21  24  27  30  33
```

```
// b) Define and code printDivisibleBy5()method
public static void printDivisibleBy5() {
   System.out.println("\n\nNumbers Divisible by 5: ");

   for (int i = 1; i <= 100; i++) {
       if (i % 5 == 0) {
          System.out.printf("%4d", i);
       }
   }
}
```



```
Enter your Choice from the Main Menu: 2


Numbers Divisible by 5:
   5  10  15  20  25  30  35  40  45  50  55
```

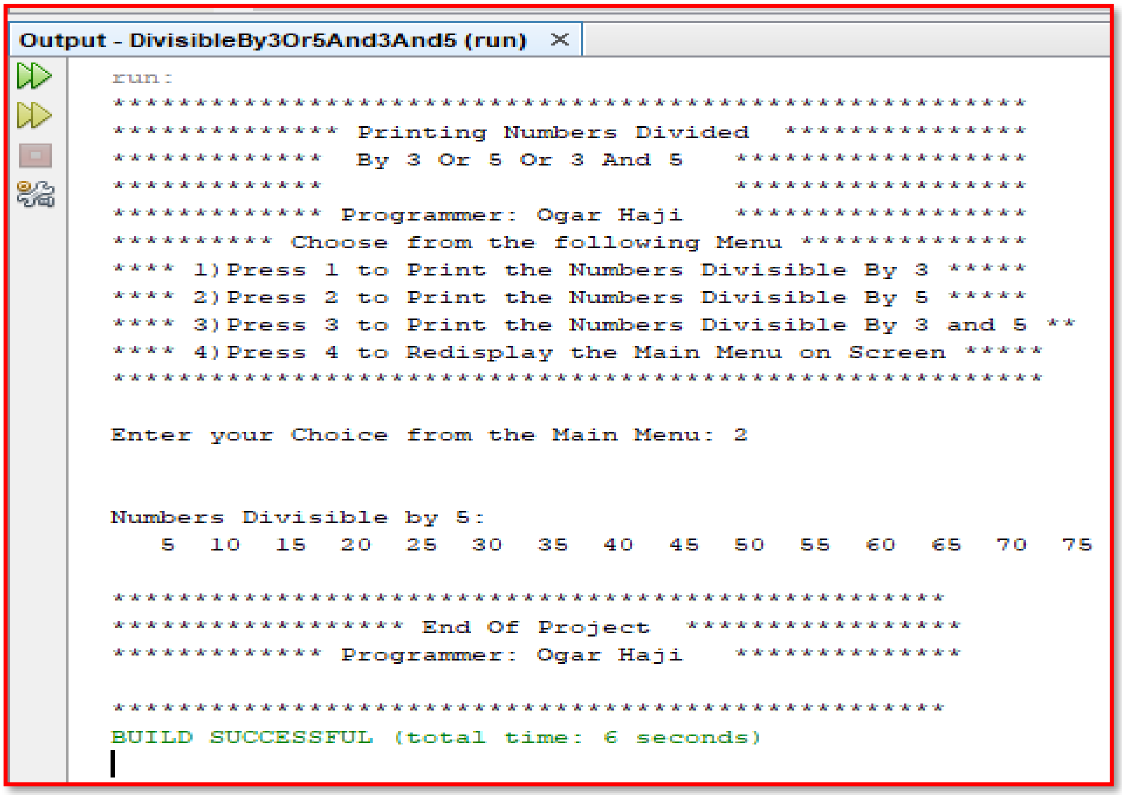
```
// c) Define and code printDivisibleBy3And5() method
public static void printDivisibleBy3And5() {
   System.out.println("\n\nNumbers Divisible by 3 And 5: ");
   for (int i = 1; i <= 100; i++) {

       if (i % 3 == 0 && i % 5 == 0) {
          System.out.printf("%4d", i);
       }

   }
    System.out.println("\n");
}
```



```
Enter your Choice from the Main Menu: 3


Numbers Divisible by 3 And 5:
  15  30  45  60  75  90
```

```
// 4) Define and code printFooters() method
public static void printFooters() {
   System.out.println("\n***************************************");
   System.out.println("************ End Of Project  *************");
   System.out.println("******** Programmer: Ogar Haji   *********");
   System.out.println("***************************************\n");
  }
}
```

Output of project with choice of 2 (Divisible by 5) is shown below:



```
run:
********************************************************
************** Printing Numbers Divided  ***************
*************  By 3 Or 5 Or 3 And 5   ******************
*************                         ******************
************* Programmer: Ogar Haji   ******************
********** Choose from the following Menu **************
**** 1)Press 1 to Print the Numbers Divisible By 3 *****
**** 2)Press 2 to Print the Numbers Divisible By 5 *****
**** 3)Press 3 to Print the Numbers Divisible By 3 and 5 **
**** 4)Press 4 to Redisplay the Main Menu on Screen *****
*********************************************************

Enter your Choice from the Main Menu: 2


Numbers Divisible by 5:
    5   10   15   20   25   30   35   40   45   50   55   60   65   70   75

***************************************************
****************** End Of Project  ****************
************* Programmer: Ogar Haji   **************

***************************************************
BUILD SUCCESSFUL (total time: 6 seconds)
```

Note: Always **Upload** to **Brightspace 2 Files** of **same Java Project:**
1) The **Microsoft Word Document** of the **Java Project Code along with the Java Output Screen shots.**
2) The **Compressed** or **Zipped File** or **Folder** of **Java Project**.

Do Only 1 Modification at a Time

Working with Files:
Write Students Info to a Text File

+++(Do Lab Exercise)100 Points+++

Do Lab Exercise

Lesson 250 Ex: How to Write Students Information to Students Text File in Java ?

Use For statement to loop or traverse in an array.

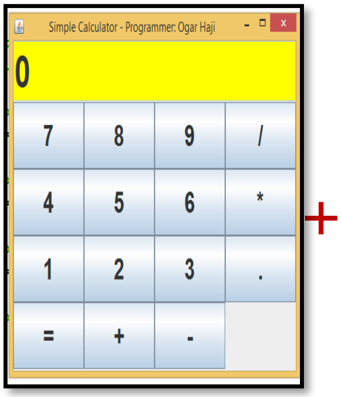
```
/*
 This project will create a File called students.txt to write text to it
 It will prompt the user to enter Student First Name, first test and second test.
 It is also using try.... catch statement and if No Invalid data,
 the project will write the student First Name, Last Name, First test and Second
 test to the file students.txt
programmer: Ogar Haji
Date Written: May 4, 2020
*/
package createstudentsfile;

import java.io.FileNotFoundException;
import java.util.Formatter;
import java.util.FormatterClosedException;
import java.util.NoSuchElementException;
import java.lang.SecurityException;
import java.util.Scanner;

public class CreateStudentsFile {

public static void main(String[] args) throws FileNotFoundException {
```

```
// 1) Declare the variables to be used in the project
String firstName, lastName;
int firstTest, secondTest;
String moreNames = "Y";

// 2) Open students file and output or write data to file
try (Formatter output = new Formatter ("students.txt")) {
   Scanner input = new Scanner (System.in);

   // 3) Use while statement to check if there are more students
   while (moreNames.equals("Y")) {

      // 4) Use try and catch to check for exceptions
      try {

         // 5) Prompt user to enter student First Name and get it
         System.out.print ("Enter student First Name: ");
         firstName = input.nextLine();

         // 6) Prompt user to enter student Last Name and get it
         System.out.print ("Enter student Last Name: ");
         lastName = input.nextLine();

         // 7) Prompt user to enter student First Test and get it
         System.out.print ("Enter First Test: ");
         firstTest = input.nextInt();

         // 8) Prompt user to enter student Second Test and get it
         System.out.print ("Enter Second Test: ");
         secondTest = input.nextInt();

         // 9) Use output.format to write student info to output file students.txt
        output.format ("%s %s %d %d %n", firstName, lastName, firstTest,
                                                    secondTest);
         input.nextLine();

         // 10) Prompt the user if more students to enter and get it
         System.out.print ("Do you have More Students to Enter (Y/N): ");
         moreNames = input.nextLine();
```

```
            // 11) Convert to Upper Case
            moreNames = moreNames.toUpperCase();

        } // end of second try after while

        // 12) Use catch statement to catch the Invalid Input
        catch (NoSuchElementException e){
            System.err.println("Invalid Input, please Try Again");
            input.nextLine();
        }
    } // end of while statement
 } // end of the First try after while

        // 13) Use a second catch statement to catch File Not Found Exception
        catch (SecurityException | FileNotFoundException
                | FormatterClosedException se){
                se.printStackTrace();
  }
 } // end of main method
} // end of the Class CreateStudentsFile
```

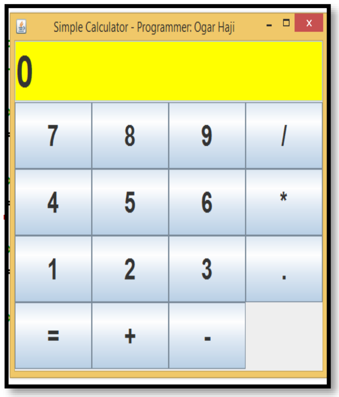
Chapter 12 + Java Lab Assignment #12 (Due Next Week) 200 Points

Name:___________________________ **CIS144 Java language + Wright College**

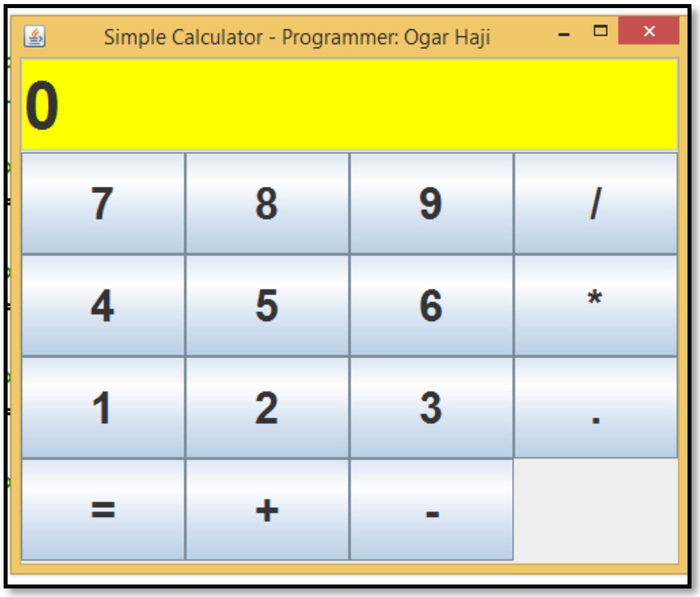


Design a Simple Calculator + Using Arrays + Java Project + (Do Lab Assignment + Lab 12)

Do Lab Exercise + Lab 12

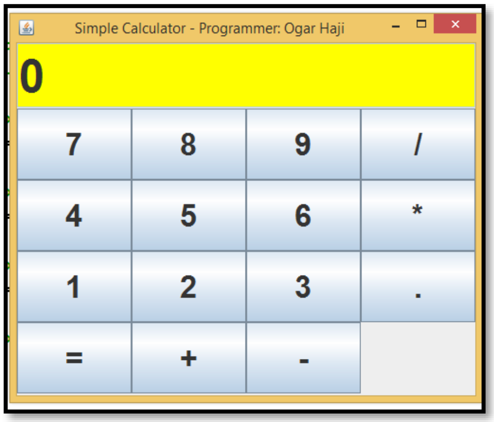


Design a Simple Calculator in Java

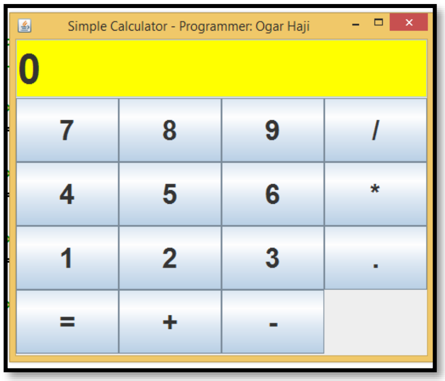


```
/*
    This is a program that simulate a Simple Calculator.
    It uses a for statement to Add All the Buttons (0 thru 0)
    along with Math operator to the JFrame
    Date: December 10, 2017
    Programmer: Ogar Haji
```

```
*/
package calculatorusingfor;

// Import the classes needed for this project
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
```

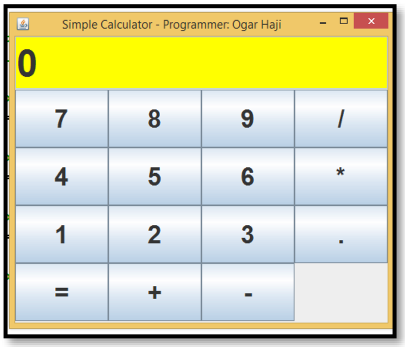


```
public class CalculatorUsingFor extends JPanel implements ActionListener {
   // Declare the objects and variables to be used
   private JTextField displayJTextField = new JTextField("0");
   private String buttons = "789/456*123.=+-";
   private double result = 0;
   private String operator = "=";
   private boolean calculating = true;
   private boolean decimal = false;
```

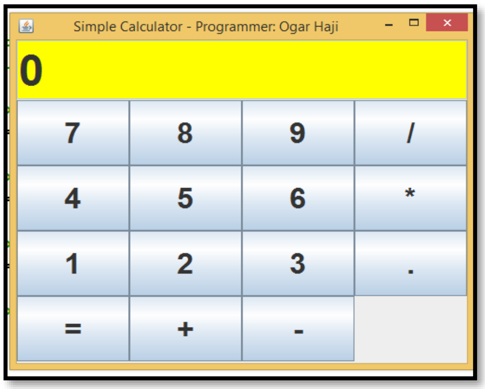


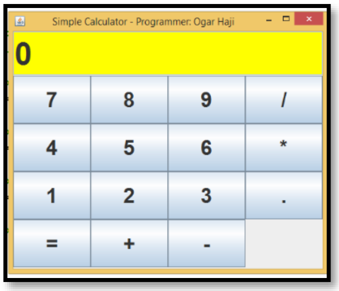
```
   // The main() method will create an object frame of the class JFrame
   public static void main(String[] args) {
      JFrame frame = new JFrame();
      frame.setTitle ("Simple Calculator - Programmer: Ogar Haji");
      frame.setBounds (450,200,200,200);
      frame.setSize (500,400);

      // Add windows Listener to the object frame and code
      // window Closing method
```

```
    frame.addWindowListener(new WindowAdapter() {
     public void windowClosing(WindowEvent e) {
          System.exit(0);  // the program exited successfully
       }
    });
```

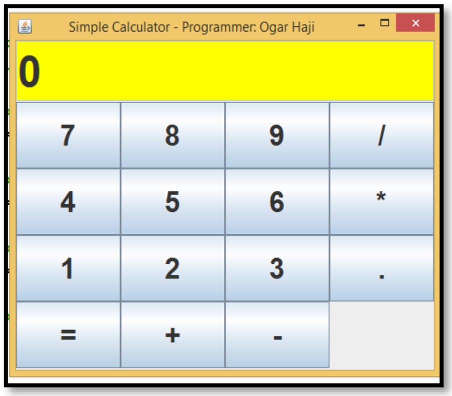


```
    // Declare contentPane object to get the frame's contentPane
    Container contentPane = frame.getContentPane();

    // Add the object CalculateorUsingFor to the conentPane and show it
    contentPane.add(new CalculatorUsingFor());
    frame.show();
  }

public CalculatorUsingFor() {
  // SetLayout to BorderLayout (North, East, Sought, West and center)
  setLayout(new BorderLayout());

  // Set the content of display Text Field as Read Only
  displayJTextField.setEditable(false);

  //Add display at North of layout
  add(displayJTextField, "North");

  JPanel p = new JPanel();    // Create a new object p of JPanel
    // Set the Layout of the grid to 4 Rows and 4 Columns
  p.setLayout (new GridLayout (4,4));
```

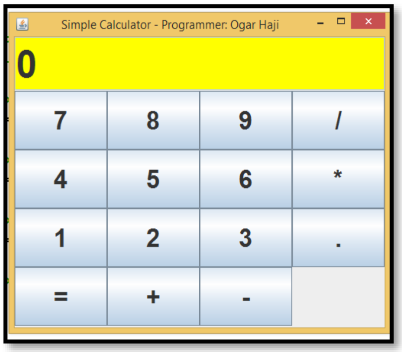




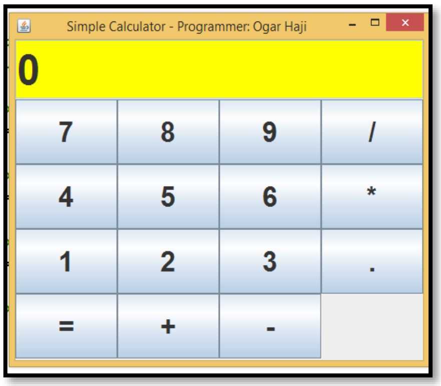
```
      // Use for statement to add all Buttons b to the Panel
     for (int i = 0; i < buttons.length(); i++) {
        JButton b = new JButton(buttons.substring(i, i + 1));
        p.add(b); // Add button to the panel
        b.addActionListener(this); // Add Action Listiner to the button
        b.setFont(new Font("Times",Font.BOLD,32)); // set Font Size to size 32
     }

     // set Font Size of display Text Field to size 48, Bold and Yellow color
      displayJTextField.setFont(new Font("Times",Font.BOLD,48));
      displayJTextField.setBackground(Color.YELLOW);

     add (p, "Center");   // add the panel to the center position
 }
```



```
// Declare and code the actionPerformed method
public void actionPerformed(ActionEvent evt) {
   // Get the action Command performed by the user
   String actionCommand = evt.getActionCommand();

   // Check if the user clicked numbers o thru 9 or the decimal period .
   if ('0' <= actionCommand.charAt(0) && actionCommand.charAt(0) <= '9'
         || actionCommand.equals("."))    {
     if (calculating)  {  // if calculating is true
        displayJTextField.setText(actionCommand);
     }
     else {
         displayJTextField.setText(displayJTextField.getText() + actionCommand);
     }
     calculating = false;
   }
   else {
     if (calculating) {
         if (actionCommand.equals("-"))
         calculating = false;
     }
     else {
         try {
             double x = Double.parseDouble(displayJTextField.getText());
             calculate(x);
             operator = actionCommand;
             calculating = true;
             }
         catch(NumberFormatException e)
         {
             displayJTextField.setText("Illegal Number Format");
             calculating = false;
         }
     calculating = true;
     }
   }
} // End of Action performed method
```

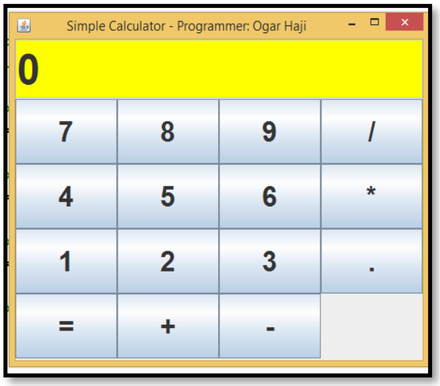

Simple Calculator - Programmer: Ogar Haji

0


Simple Calculator - Programmer: Ogar Haji

0

```
private void calculate(double n) {
  if(operator.equals("+"))
    result += n;

  else if (operator.equals("-"))
     result -= n;

  else if (operator.equals("/"))
    result /= n;

    else if (operator.equals("*"))
  result *= n;


    else if(operator.equals("="))
  {
    result = n;
  }

  displayJTextField.setText("" + result);
 }
}
```



Please, Read, Study and Practice the Lessons in the Java Handout

Modify the Program:

Calculate Area of a Room

Note: When Modifying a Program, Do One Modification at a time.

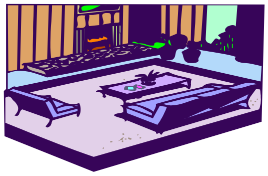
Modify calculate_rectangle_area.py program to **do the following:**

1) Also **Calculate** the **Perimeter** of the **Rectangle**.

2) Login on ITT Virtual Library and **find the Formula to Calculate the Perimeter of a Rectangle.**

3) Also Calculate Amount Charged to cover the Room Floor with Carpet if the company charges $7.00 per Square Foot.

The Output will look like the following:



```
The Area of the Rectangle in Square Feet = 80
The Perimeter of the Rectangle in Feet = 36
Amount Charged to cover Room Floor with Carpet = $ 560.0
```

Remember: Just write the steps and statements of Modifying the program in Python Language in your Notebook, then apply them to the actual program.

Note: Submit this **Modified program** to your **Instructor.**