# Department of Computing

**CS 212: Object Oriented Programming**

**Class: BESE-11AB**

**Lab03: Flow Control [Loops, Decisions & Arrays]**

**Date: 24th March, 2021 Time: 2:00pm- 4:50pm, 09:00am-11:50 am**

**Instructor: Ms. Hania Aslam**

## Learning Objectives

### The objective of this lab is to introduce the concepts of relational operators, if conditions, loops and arrays. In particular, the students are assumed to have basic understanding of these concepts. However, more focus is put on implementing these constructs in the Java language.

**Activity #1.**

Your first activity will be to test the if statements, relational and equality operators in Java. Test the following program:

// Compare integers using if statements, relational operators

// and equality operators.

import java.util.Scanner; // program uses class Scanner

public class Comparison {

// main method begins execution of Java application public static void main( String[] args )

{

// create Scanner to obtain input from command line Scanner input = new Scanner( System.in );

int number1; // first number to compare int number2; // second number to compare

System.out.print( "Enter first integer: " ); // prompt number1 = input.nextInt(); // read first number from user

System.out.print( "Enter second integer: " ); // prompt number2 = input.nextInt(); // read second number from user

if ( number1 == number2 )

System.out.printf( "%d == %d\n", number1, number2 );

if ( number1 != number2 )

System.out.printf( "%d != %d\n", number1, number2 );

if ( number1 < number2 )

System.out.printf( "%d < %d\n", number1, number2 );

if ( number1 > number2 )

System.out.printf( "%d > %d\n", number1, number2 );

if ( number1 <= number2 )

System.out.printf( "%d <= %d\n", number1, number2 );

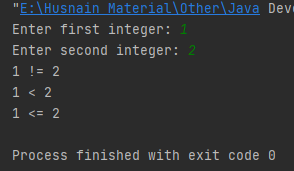
if ( number1 >= number2 )

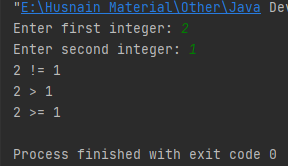
System.out.printf( "%d >= %d\n", number1, number2 );

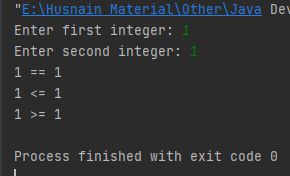
} // end method main

} // end class Comparison

**Output Screenshot:**







## Activity #2.

### Do as directed in the comments at the start of Calculate class.

public class Calculate {

/\*\*

* There is a mysterious error in this program. Identify this error.
* Is this a syntax error? or a logical one.

\*/

public static void main(String[] args) { int sum;

int x;

x = 1; // initialize x to 1 for counting sum = 0; // initialize sum to 0 for totaling

System.out.printf( "Going to calculate the sum" ); while ( x <= 10 ) // while x is less than or equal to 10

{

sum += x; // add x to sum

} // end while

System.out.printf( "The sum is: %d\n", sum );

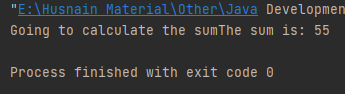
}

}

**Error:**

The only problem with this program Is that in the while loop the value of ‘x’ is not updated. Hence, the while loop becomes an infinite loop and the program keeps on running forever. It is a **logical error** as the program runs fine but our required output is not produced. This bug can be removed by just putting an increment operator **‘x++’** at the end of while loop inside the while loop so that the value of x gets updated in each loop.

**Output Screenshot:**



## Activity #3.

### What does the following program print? Predict the output, before you actually try this program.

public class Mystery

{

public static void main( String[] args )

{

int y;

int x = 1; int total = 0;

while ( x <= 10 )

{

y = x \* x; System.out.println( y ); total += y;

++x;

} // end while

System.out.printf( "Total is %d\n", total );

} // end main

} // end class Mystery

**Predicted Output:**

1

4

9

16

25

36

49

64

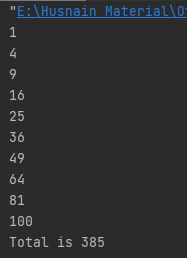
81

100

Total is

385

## Real Output:



## Activity #4.

### Try the following program and pay special attention to how arrays are created/used in Java. What is the output of this program? How else can you use this program? Think!!

public class Cards {

public static void main(String[] args) {

String [] suit = { "Clubs", "Diamonds", "Hearts", "Spades"}; String [] rank=

{

"2", "3", "4", "5", "6", "7", "8", "9", "10", "Jack", "Queen", "King", "Ace"

};

String [] deck = new String [suit.length \* rank.length];

for (int i = 0; i < suit.length; i++)

for (int j = 0; j < rank.length; j++)

deck [rank.length \* i + j] = rank [j] + " of " + suit[i];

for (int k = 0; k < (suit.length \* rank.length); k++) System.out.println(deck[k]);

}

}

### Describe the outcome of the above code?

**Output:**

2 of Clubs

3 of Clubs

4 of Clubs

5 of Clubs

6 of Clubs

7 of Clubs

8 of Clubs

9 of Clubs

10 of Clubs

Jack of Clubs

Queen of Clubs

King of Clubs

Ace of Clubs

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

2 of Clubs

3 of Clubs

4 of Clubs

5 of Clubs

6 of Clubs

7 of Clubs

8 of Clubs

9 of Clubs

10 of Clubs

Jack of Clubs

Queen of Clubs

King of Clubs

Ace of Clubs

2 of Diamonds

3 of Diamonds

4 of Diamonds

5 of Diamonds

6 of Diamonds

7 of Diamonds

8 of Diamonds

9 of Diamonds

10 of Diamonds

Jack of Diamonds

Queen of Diamonds

King of Diamonds

Ace of Diamonds

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

null

2 of Clubs

3 of Clubs

4 of Clubs

5 of Clubs

6 of Clubs

7 of Clubs

8 of Clubs

9 of Clubs

10 of Clubs

Jack of Clubs

Queen of Clubs

King of Clubs

Ace of Clubs

2 of Diamonds

3 of Diamonds

4 of Diamonds

5 of Diamonds

6 of Diamonds

7 of Diamonds

8 of Diamonds

9 of Diamonds

10 of Diamonds

Jack of Diamonds

Queen of Diamonds

King of Diamonds

Ace of Diamonds

2 of Hearts

3 of Hearts

4 of Hearts

5 of Hearts

6 of Hearts

7 of Hearts

8 of Hearts

9 of Hearts

10 of Hearts

Jack of Hearts

Queen of Hearts

King of Hearts

Ace of Hearts

null

null

null

null

null

null

null

null

null

null

null

null

null

2 of Clubs

3 of Clubs

4 of Clubs

5 of Clubs

6 of Clubs

7 of Clubs

8 of Clubs

9 of Clubs

10 of Clubs

Jack of Clubs

Queen of Clubs

King of Clubs

Ace of Clubs

2 of Diamonds

3 of Diamonds

4 of Diamonds

5 of Diamonds

6 of Diamonds

7 of Diamonds

8 of Diamonds

9 of Diamonds

10 of Diamonds

Jack of Diamonds

Queen of Diamonds

King of Diamonds

Ace of Diamonds

2 of Hearts

3 of Hearts

4 of Hearts

5 of Hearts

6 of Hearts

7 of Hearts

8 of Hearts

9 of Hearts

10 of Hearts

Jack of Hearts

Queen of Hearts

King of Hearts

Ace of Hearts

2 of Spades

3 of Spades

4 of Spades

5 of Spades

6 of Spades

7 of Spades

8 of Spades

9 of Spades

10 of Spades

Jack of Spades

Queen of Spades

King of Spades

Ace of Spades

**Task #1:**

Body Mass Index Calculator: The formula for calculating BMI is

BMI = Weight

Height ∗Height

### Create a BMI calculator that reads the user’s weight in kilograms and height in meters, then calculates and displays the user’s body mass index. Also, display the following information, using control structures, so that the user can evaluate his/her BMI.

**BMI VALUES:**

**Underweight**: less than 18.5

**Normal**: between 18.5 and 24.9

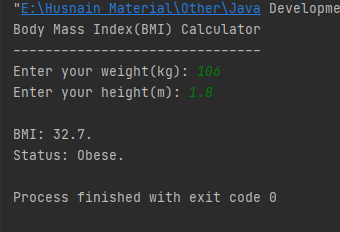
**Overweight**: between 25 and 29.9

**Obese**: 30 or greater

**Code:**

package com.company;  
import java.util.Scanner;  
  
public class Task01 {  
  
 public static void main(String[] args) {  
  
 //Declaring variables  
 float weight;  
 float height;  
 float bmi;  
  
 Scanner sc = new Scanner(System.*in*);  
  
 //Title  
 System.*out*.println("Body Mass Index(BMI) Calculator");  
 System.*out*.println("-------------------------------");  
  
 //Taking Inputs  
 System.*out*.print("Enter your weight(kg): ");  
 weight = sc.nextFloat();  
  
 System.*out*.print("Enter your height(m): ");  
 height = sc.nextFloat();  
  
 //Calculating BMI  
 bmi = weight / (height \* height);  
  
 System.*out*.printf("\nBMI: %.1f.\n", bmi);  
  
 //Checking for the status of the subject  
 if (bmi < 18.5){  
 System.*out*.println("Status: Underweight.");  
 }  
 else if (18.5 <= bmi && bmi <= 24.9){  
 System.*out*.println("Status: Normal.");  
 }  
 else if (bmi <= 29.9 && bmi >= 25){  
 System.*out*.println("Status: Overweight.");  
 }  
 else if (bmi >= 30){  
 System.*out*.println("Status: Obese.");  
 }  
 sc.close();  
 }  
}

**Output Screenshot:**



## Task #2:

### Drivers are concerned with the mileage their automobiles get. One driver has kept track of several trips by recording the kilometers driven and liters used for each tankful.

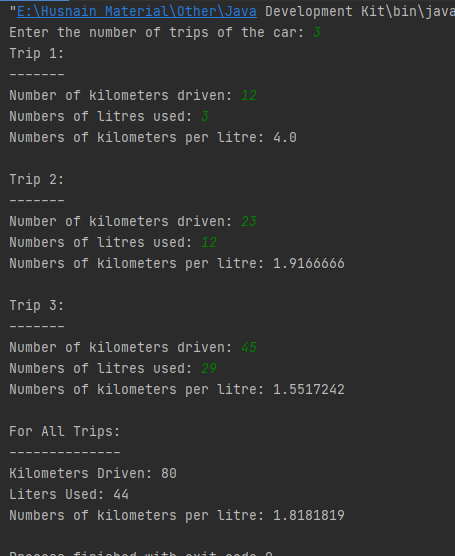
Develop a Java application that will input the kilometers driven and liters used (both as integers) for each trip. The program should calculate and display the kilometers per liter obtained for each trip and print the combined kilometers per liter obtained for all trips up to this point, as well.

All averaging calculations should produce floating-point results. Use class Scanner and sentinel- controlled repetition to obtain the data from the user.

**Code:**

package com.company;  
import java.util.Scanner;  
  
public class Task02 {  
  
 public static void main(String[] args) {  
  
 //Declaring variables  
 byte trips;  
 int kilometers\_driven;  
 int litres\_used;  
 float km\_per\_L;  
 int total\_kilometers\_driven;  
 int total\_litres\_used;  
 float total\_km\_per\_L;  
  
 //Making instance of Scanner class  
 Scanner sc = new Scanner(System.*in*);  
  
 //Taking input the number of trips of the car  
 System.*out*.print("Enter the number of trips of the car: ");  
 trips = sc.nextByte();  
  
 total\_kilometers\_driven = 0;  
 total\_litres\_used = 0;  
 //This loop calculates the number of kilometers driven per litre  
 for (int i = 1; i <= trips; i++){  
  
 System.*out*.printf("Trip %d:\n-------\n", i);  
  
 System.*out*.print("Number of kilometers driven: ");  
 kilometers\_driven = sc.nextInt();  
  
 System.*out*.print("Numbers of litres used: ");  
 litres\_used = sc.nextInt();  
  
 km\_per\_L = (float) kilometers\_driven / litres\_used;  
 System.*out*.println("Numbers of kilometers per litre: " + km\_per\_L);  
 System.*out*.println();  
  
 total\_kilometers\_driven += kilometers\_driven;  
 total\_litres\_used += litres\_used;  
  
 }  
 //Calculating data for all trips  
 System.*out*.println("For All Trips:\n--------------");  
 System.*out*.printf("Kilometers Driven: %d\nLiters Used: %d\n", total\_kilometers\_driven, total\_litres\_used);  
 total\_km\_per\_L = (float) total\_kilometers\_driven / total\_litres\_used;  
 System.*out*.println("Numbers of kilometers per litre: " + total\_km\_per\_L);  
  
 sc.close();  
 }  
}

**Output Screenshot:**



**Hand in**

Hand in the source code from this lab at the appropriate location on the LMS system.

**To Receive Credit**

1. Comment your program heavily. Intelligent comments and a clean, readable formatting of your code account for 20% of your grade.
2. The lab time is not intended as free time for working on other assignments.