



Computer Science and Software Engineering Department

CSC103– Object Oriented Programming

Spring 2025

Lab #2: Java conditional statements and loops

Objective:

To understand Java programming basics. In this laboratory, students are expected to learn, java conditional statements and loops.

Name of Student	
Student ID	
Date of Lab Conducted	
Marks Obtained	
Remarks	
Signature	

UIT University
Computer Science and Software Engineering Department
CSC103– Object Oriented Programming
Fall 2024

1. Objective

To understand Java programming basics. In this laboratory, students are expected to learn java syntax, java variables, java conditional statements, loops and little bit of introduction about object oriented paradigm of java.

2. Instructions

State the instruction that student needs to follow for performing the example and exercises. Some sample instructions are given below which can be altered as needed

E.g.

- Come to the lab in time. Students who are late more than 15 minutes, will not be allowed to attend the lab.
- Students have to perform the examples and exercises by themselves.
- Raise your hand if you face any difficulty in understanding and solving the examples or exercises.
- Lab work must be submitted on or before the submission date.

3. How to Submit

State the procedure that how to submit the lab work by the student. This section contains the instructions related to how the student will go through the lab's content and how will s/he submit the task file. Some sample instructions are given below which can be altered as needed. (*Font size 11, Arial*)

E.g.

- Submit lab work in a single pdf/docx/xlsx/cpp/py file on MS Team/Google Classroom.
- Submit the work as per format given in this manual (No other format will be accepted)
- Lab work (Exercises) file name should be saved with your roll number and course code (e.g. 19B-001-CS_CSxxx_LWxx.pdf where CSxxx is course code and LWxx is Lab number)
- If required attach your programming/coding file as well.

Java Programming fundamentals:

Variable:

A variable is the name of a reserved area allocated in memory. In other words, it is a name of the memory location. It is a combination of "vary + able" which means its value can be changed

There are three types of variables in java:

- local variable
- instance variable
- static variable

Example:

```
public class JavaApplication1 {  
  
    public static void main(String[] args) {  
  
        int a=10;  
  
        int b=10;  
  
        int c=a+b;  
  
        System.out.println(c);  
  
    }  
  
}
```

JAVA CONTROL STATEMENTS | CONTROL FLOW IN JAVA

Java compiler executes the code from top to bottom. The statements in the code are executed according to the order in which they appear. However, java provides statements that can be used to control the flow of Java code. Such statements are called control flow statements. It is one of the fundamental features of Java, which provides a smooth flow of program.

Java provides three types of control flow statements.

1. Decision Making statements
 - if statements
 - switch statement
2. Loop statements
 - do while loop
 - while loop
 - for loop
 - for-each loop
3. Jump statements
 - break statement
 - continue statement

Conditional statement Example:

If else statement is a condition statement that is used in the execution of a computer program in pre-defined rules. The if-else statement helps you to run a specific block of a program if the condition is true or else, it will check other conditions. It is used to control the flow or to determine the rules in a program.

```
If (condition 1) {  
  
statement 1; //executes when condition 1 is true  
  
}  
  
else if (condition 2) {  
  
statement 2; //executes when condition 2 is true  
  
}  
  
else {  
  
statement 2; //executes when all the conditions are false  
  
}
```

Switch example:

Instead of writing many if..else statements, you can use the switch statement.

The switch statement selects one of many code blocks to be executed:

This is how it works:

- The switch expression is evaluated once.
- The value of the expression is compared with the values of each case.
- If there is a match, the associated block of code is executed.

```
public class Main {
    public static void main(String[] args) {
        int day = 4;
        switch (day) {
            case 1:
                System.out.println("Monday");
                break;
            case 2:
                System.out.println("Tuesday");
                break;
            case 3:
                System.out.println("Wednesday");
                break;
            case 4:
                System.out.println("Thursday");
                break;
            case 5:
                System.out.println("Friday");
                break;
            case 6:
                System.out.println("Saturday");
                break;
            case 7:
                System.out.println("Sunday");
                break;
        }
    }
}
// Outputs "Thursday" (day 4)
```

Loops:

Loops are used to repeat a block of code as long as a specified condition is reached.

Loops are handy because they save time, reduce errors, and they make code more readable.

For Loop:

Statement 1 is executed (one time) before the execution of the code block.

Statement 2 defines the condition for executing the code block.

Statement 3 is executed (every time) after the code block has been executed.

The example below will print the numbers 0 to 4:

For loop Example

```
for (int i = 0; i < 5; i++) {  
    System.out.println(i);  
}
```

for-Each Loop

There is also a "**for-each**" loop, which is used exclusively to loop through elements in an array:

Syntax

```
for (type variableName : arrayName) {  
    // code block to be executed  
}
```

The following example outputs all elements in the **cars** array, using a "**for-each**" loop:

Example

```
String[] cars = {"Volvo", "BMW", "Ford", "Mazda"};  
for (String i : cars) {  
    System.out.println(i);  
}
```

Java While Loop

The while loop loops through a block of code as long as a specified condition is true:

Syntax

```
while (condition) {  
    // code block to be executed  
}
```

In the example below, the code in the loop will run, over and over again, as long as a variable (i) is less than 5:

Example

```
int i = 0;
```

```
while (i < 5) {
    System.out.println(i);
    i++;
}
```

JAVA DO-WHILE LOOP

The Java *do-while loop* is used to iterate a part of the program repeatedly, until the specified condition is true. If the number of iteration is not fixed and you must have to execute the loop at least once, it is recommended to use a do-while loop.

Java do-while loop is called an **exit control loop**. Therefore, unlike while loop and for loop, the do-while check the condition at the end of loop body. The Java *do-while loop* is executed at least once because condition is checked after loop body.

Syntax:

```
do{

    //code to be executed / loop body

    //update statement

}while (condition);
```

Example:

```
public class DoWhileExample {
    public static void main(String[] args) {
        int i=1;
        do {
            System.out.println(i);
            i++;
        } while(i<=10);
    }
}
```

4. Exercise(s)

Perform the following tasks in java language using NetBeans/Eclipse IDE.

■

Task 1

.....

Write a Java program which takes marks of 5 subjects (maths, English, islamiat, computer science and science) as an input out of hundred and calculates its average. After calculating its average, it determines whether the student falls in A, B or C grade. If average is above 88, he/she has A grade. If above 70 and below 88, then B and below 70 is C.

Task 2

.....

Write a program that prompts the user to input an integer and then outputs the number with the digits reversed. For example, if the input is 12345, the output should be 54321.

Hint: Use While loop

Task 3:

A PALINDROME is a word which has SAME SPELLING whether it is read from Left to Right or from Right to Left. Example: MOM, DAD, DEED, PEEP and NOON. Other words which are not PALINDROME are HELLO, DOOR and FEET. Write a program that can read a String as user input in Capital Letters and then Print YES as Output if the Input is a PALINDROME otherwise NO.

Task 4:

Write a program in java to produce following outputs:

```
*
* *
* * *
* * * *
* * * * *
```

```
    *
  * *
* * *
```


* * * *