OBject oriented programming with java

Lab manual

Aymen Khan

Aymkhan498@gmail.com

**Course Title:** Object Oriented Programming

**Course Code:**

**Course Outcomes:**

At the end of the course the student should be able to:

1. Apply Object Oriented Programming concepts to solve a given problem.
2. Apply design patterns to design a solution for a given problem.
3. Apply inheritance, polymorphism and exception handling mechanism to implement reusable, robust java programs.
4. Implement user interface java programs for a given scenario.

**List of Practical**

|  |  |  |
| --- | --- | --- |
| **Sr.#** | **Week** | **Topics** |
| 1 | One | Introduction to IDE-NetBeans, Getting Started with Java |
| 2 | Two | Java Basics (Input/output, variable declaration and initialization, strings and arrays) |
| 3 | Three | Java Basics (Selection Structure and Iterative Structure, functions) |
| 4 | Four | OOP (creating classes, objects, constructors) |
| 5 | Five | access modifiers, inheritance |
| 6 | Six | multiple/multilevel inheritance |
| 7 | Seven | Function overriding |
| 8 | Eight | Polymorphism |
| 9 | Nine | abstract classes and interfaces |
| 10 | Ten | exception handling and Java file handling |

**Introduction to NetBeans and Java**

**LAB-1**

**Java Basics-2**

**LAB-3**

**Java Conditions and If Statements**

Java supports the usual logical conditions from mathematics:

Less than: a < b

Less than or equal to: a <= b

Greater than: a > b

Greater than or equal to: a >= b

Equal to a == b

Not Equal to: a != b

You can use these conditions to perform different actions for different decisions. Java has the following conditional statements:

Use if to specify a block of code to be executed, if a specified condition is true

Use else to specify a block of code to be executed, if the same condition is false

Use else if to specify a new condition to test, if the first condition is false

Use switch to specify many alternative blocks of code to be executed

**The if Statement**

Use the if statement to specify a block of Java code to be executed if a condition is true.

**Syntax**

if (condition) {

// block of code to be executed if the condition is true

}

We test two values to find out if 20 is greater than 18. If the condition is true, print some text:

if (20 > 18) {

System.out.println("20 is greater than 18");

}

**The if else Statement**

int time = 20;

if (time < 18) {

System.out.println("Good day.");

} else {

System.out.println("Good evening.");

}

## **Java For Loop**

When you know exactly how many times you want to loop through a block of code, use the for loop.

**Syntax**

for (*statement 1*; *statement 2*; *statement 3*) {

*// code block to be executed*

}

**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.

**Example**

for (int i = 0; i < 5; i++) {

System.out.println(i);

}

**Example**

for (int i = 0; i <= 10; i = i + 2) {

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);

}

**Task-1**

Write a program using JAVA for a function that take five numbers from the user and store in array of size 5 after that display the sum and average of numbers.

**Hint:** Use scanner class and loop for taking input from user. You can use sc.nextInt() for taking integer from user and save that integer in array.

**Task-2**

Write a program that prompts the user to input a positive integer. It should then print the multiplication table of that number.

**Assignment Question**

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.