Code: BCA 2004T (For Theory) BCA 2004P (For Practical)	SEC-II	Object Oriented Programming Using Java	3L+T: 4P	5 Credits (45 hours theory and 60 hours practical)

Max Marks; Theory: 100 (Int: 25; Ext: 75); Practical: 100

Course Outcomes: Upon competition of the course, students will be able to

CO1: Understand the fundamental concepts of Object-Oriented Programming (OOP) and Java language structure.

CO2: Apply decision-making, branching, looping, and operator usage in Java programming.

CO3: Implement concepts of classes, objects, inheritance, polymorphism, arrays, and strings in Java programs.

CO4: Develop Java programs using packages and exception handling mechanisms.

Unit	Topics	Purposed lectures
I	**Fundamentals of Object-Oriented Programming** – Basic Concepts of Object-Oriented Programming (OOP), Benefits and Applications of OOP. **Java Evolution** – Java Features, Difference between Java, C and C++, Java and Internet, Java Environment.	11
	Overview of Java Language – Setting up Java Development Environment (JDK, IDEs), Introduction to Simple Java Program, Use of Comments and Math functions, Application of two classes, Java Program Structure, Java Tokens and Statements, Implementing Java Program and JVM, Command Line Arguments.	
II	**Constants, Variables and Data Types** – Constants, Variables, Primitive & Non-Primitive Data Types, Declaration of Variables, Giving values to Variables, Symbolic Constants, Typecasting. **Operators & Expressions** – Arithmetic operators, Relational operators, Logical operators, Assignment operators, Increment & Decrement operators, Conditional operators, Bitwise operators, Arithmetic Expressions, Evaluation of Expressions, Type Conversions in Expressions, Operator Precedence & Associativity. **Decision Making, Branching & Looping** – Decision Making with Control Statements (if-else, switch-case), Looping Statements (for, while, do-while), Jump in loops, Labeled loops.	11
III	**Classes, Objects and Methods** – Java Keywords, Defining Class, Instance Variables & Methods, Creating Objects, Methods Declaration, Constructors, this keyword, Static Members (Variables & Methods). **Arrays, Strings and Vectors** – 1D Arrays, Creating an Array, 2D Arrays, Strings, Vectors, Wrapper Classes, Enumerated Types. **Inheritance** – Defining Classes & Objects, Access Modifiers, Extending Classes and Implementing Interfaces, Multiple Inheritance using Interfaces and Polymorphism (Method Overloading & Overriding Methods).	11
IV	**Packages** – Basics of Packages, System Packages, Creating and Accessing Packages, Creating User-Defined Packages, Adding Class to a Package. **Exception Handling** – Using the main keywords of exception handling: try, catch, throw, throws and finally; Nested try, Multiple catch statements, Creating User-Defined Exceptions.	12
Lab Programs	1. Write a program to read two numbers from the user and print their product. 2. Write a program to print the square of a number passed through command line arguments. 3. Write a program to send the name and surname of a student through command line arguments and print a welcome message for the student. 4. Write a Java program to find the largest number out of *n* natural numbers. 5. Write a Java program to find the Fibonacci series & Factorial of a number using recursive and non-recursive functions. 6. Write a Java program to multiply two given matrices. 7. Write a Java program for sorting a given list of names in ascending order. 8. Write a Java program that checks whether a given string is a palindrome or not (e.g., MADAM is a palindrome). 9. Write a Java program to read *n* number of values in an array and display them in reverse order. 10. Write a Java program to perform mathematical operations. Create a class called AddSub with methods to add and subtract. Create another class called MulDiv that extends from AddSub class to use the member data of the superclass. MulDiv should have methods to multiply and divide. A main function should access the methods and perform the mathematical operations. 11. Create a JAVA class called Student with the following details as variables within it: USN, NAME, BRANCH, PHONE, PERCENTAGE. Write a JAVA program to create *n* Student objects and print the USN, Name, Branch, Phone, and Percentage of these objects with suitable headings. 12. Write a Java program that displays the number of characters, lines, and words in a text. 13. Write a Java program to create a class called Shape with methods called getPerimeter() and getArea(). Create a subclass called Circle that overrides the getPerimeter() and getArea() methods to calculate the area and perimeter of a circle.	

14. Write a Java program to create a class Employee with a method called calculateSalary(). Create two subclasses Manager and Programmer. In each subclass, override the calculateSalary() method to calculate and return the salary based on their specific roles.
15. Write a Java program using an interface called 'Bank' having a function 'rate_of_interest()'. Implement this interface to create two separate bank classes SBI and PNB to print different rates of interest. Include additional member variables and constructors in classes SBI and PNB.
16. Write a Java package program for the class Book and then import the data from the package and display the result.
17. Write a Java program for finding the cube of a number using a package for various data types and then import it into another class and display the results.
18. Write a Java program for demonstrating the divide by zero exception handling.
19. Write a Java program that reads a list of integers from the user and throws an exception if any numbers are duplicates.
20. Create an exception subclass UnderAge, which prints "Under Age" along with the age value when an object of UnderAge class is printed in the catch statement. Write a class ExceptionDemo in which the method test() throws UnderAge exception if the variable age passed to it as an argument is less than 18. Write the main() method to demonstrate the working of the program.

Text Books:

1. Balaguruswamy, E. Programming with JAVA: A Primer. 7th ed. India: McGraw Hill Education, 2023.
2. Schildt, H. Java: The Complete Reference. 12th edition. McGraw-Hill Education, 2022.
3. Programming in Java 2,BySomasundaram, K,Jaico Publishing House

Reference Books:

4. Liang, Y. Daniel. Introduction to Java Programming. 7th ed., Pearson, 2008.
5. Malhotra, S., and S. Choudhary. Programming in Java. 2nd ed., Oxford UP, 2014.
6. Ivon Horton, "Beginning Java-2", SPD Publication4
7. Ramesh Bangia, Learning JAVA-2, Khanna Book Publishing Co (P) Ltd