27.09.2022

III Semester

OBJECT ORIENTED PROGRAMMING WITH JAVA LABORATORY

Course Code	**21CSL35**	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	0:0:2:0	SEE Marks	50
Total Hours of Pedagogy	24	Total Marks	100
Credits	1	Exam Hours	03

Course Objectives:

CLO 1. Demonstrate the use of Eclipse/Netbeans IDE to create Java Applications.
CLO 2. Using java programming to develop programs for solving real-world problems.
CLO 3. Reinforce the understanding of basic object-oriented programming concepts.

Sl. No.	
	Note: two hours tutorial is suggested for each laboratory sessions.
	Prerequisite
	• Students should be familiarized about java installation and setting the java environment. • Usage of IDEs like Eclipse/Netbeans should be introduced.
Sl. No.	***PART A – List of problems for which student should develop program and execute in the Laboratory***
1	Aim: Introduce the java fundamentals, data types, operators in java Program: Write a java program that prints all real solutions to the quadratic equation ax2+bx+c=0. Read in a, b, c and use the quadratic formula.
2	Aim: Demonstrating creation of java classes, objects, constructors, declaration and initialization of variables. Program: Create a Java class called **Student** with the following details as variables within it. USN Name Branch Phone Write a Java program to create n Student objects and print the USN, Name, Branch, and Phone of these objects with suitable headings.
3	Aim: Discuss the various Decision-making statements, loop constructs in java Program: A. Write a program to check prime number B.Write a program for Arithmetic calculator using switch case menu
4	Aim: Demonstrate the core object-oriented concept of Inheritance, polymorphism Design a super class called **Staff** with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a Java program to read and display at least 3 staff objects of all three categories.
5	Aim: Introduce concepts of method overloading, constructor overloading, overriding. Program: Write a java program demonstrating Method overloading and Constructor overloading.
6	Aim: Introduce the concept of Abstraction, packages. Program: Develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using packages.
7	Aim: Introduction to abstract classes, abstract methods, and Interface in java

	Program: Write a program to generate the resume. Create 2 Java classes Teacher (data: personal information, qualification, experience, achievements) and Student (data: personal information, result, discipline) which implements the java interface Resume with the method biodata().
8	Aim: Demonstrate creation of threads using Thread class and Runnable interface, multi-threaded programming. Program: Write a Java program that implements a **multi-thread** application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number.
9	Aim: Introduce java Collections. Program: Write a program to perform string operations using ArrayList. Write functions for the following a. Append - add at end b. Insert – add at particular index c. Search d. List all string starts with given letter.
10	Aim: Exception handling in java, introduction to throwable class, throw, throws, finally. Program: Write a Java program to read two integers a and b. **Compute** a/b and print, when b is not zero. Raise an exception when b is equal to zero.
11	Aim: Introduce File operations in java. Program: Write a java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file in bytes
12	Aim: Introduce java Applet, awt, swings. Programs: Develop an applet that displays a simple message in center of the screen. Develop a simple calculator using Swings.
	PART B – Practical Based Learning
01	A problem statement for each batch is to be generated in consultation with the co-examiner and student should develop an algorithm, program and execute the program for the given problem with appropriate outputs.

Course Outcome (Course Skill Set)

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

- CO 1. Use Eclipse/NetBeans IDE to design, develop, debug Java Projects.
- CO 2. Analyze the necessity for Object Oriented Programming paradigm over structured programming and become familiar with the fundamental concepts in OOP.
- CO 3. Demonstrate the ability to design and develop java programs, analyze, and interpret object-oriented data and document results.
- CO 4. Apply the concepts of multiprogramming, exception/event handling, abstraction to develop robust programs.
- CO 5. Develop user friendly applications using File I/O and GUI concepts.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course. The student has to secure not less than 35% (18 Marks out of 50) in the semester-end examination (SEE).

Continuous Internal Evaluation (CIE):

CIE marks for the practical course is **50 Marks.**

The split-up of CIE marks for record/ journal and test are in the ratio **60:40.**

- Each experiment to be evaluated for conduction with observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments designed by