

CSE12114	Object Oriented Programming Lab	L	T	P	C
Version 1.0	Contact Hours – 30	0	0	2	1
Pre-requisite/Exposure	Knowledge of programming basics				
Co-requisite	NIL				

Course Objectives:

- To understand how to design, implement, test, debug, and document programs that use basic data types and computation, simple I/O, conditional and control structures, string handling, functions and object oriented approaches.

Course Outcomes:

On the completion of this course the student will be able to

CO1. Define classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem.

CO2. Illustrate object-oriented modelling techniques like classes and Instances modelling techniques **CO3.** Interpret fundamentals of object-oriented programming in Java, including defining

Classes, invoking methods, using class libraries, etc.

CO4. Construct programming solutions with exception handling and multi-threading concept **CO5.** Solve GUI program with proper event handling techniques.

Course Description:

This course investigates object-oriented methods including object-oriented programming methodologies and techniques. Current methodology is emphasized. The use of object-oriented features such as encapsulation, information hiding, inheritance and polymorphism is reinforced by class assignments and programming exercises. The importance of multi-threading and exception handling is introduced in this course.

Course Content:

Units	Lecture Hours
Unit-I: Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape. Write a Java program to illustrate the parameterized constructor. Write a Java program to add two numbers with int and float types using method overloading.	09

Units	Lecture Hours
Unit-II: Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box	09
Unit-III: Write a Java program to list all the files in a directory including the files present in all its subdirectories. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).	09
Unit-IV: Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in selected color. Initially, there is no message shown. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout. Develop an applet in Java that displays a simple message. Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named "Compute" is clicked.	09
Unit-V: Write a Java program that simulates a Banking GUI application with facilities of deposit, withdraw and check balance in an account. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set of integers.	09

Text Books:

- Java Fundamentals - A Comprehensive Introduction, Illustrated Edition ByDaleskrien, Herbert Schildt, Mcgraw- Hill Education.

Reference Books:

- Java For Programmers, 2nd Edition By Paul Deitel And Harvey Deitel, Pearson Education. Thinking In Java, Low Price Edition By Bruce Eckel, Pearson Education

Relationship between Course Outcomes (COs) and Program Outcomes (POs):

1 = Weakly Mapped

2 = Moderately Mapped

3 = Strongly Mapped

SEMESTER V

Course Code	Course Name	COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
CSE12	Object	CO1211	2	2	3	2	3	1	3	-	3	-	-	3	3	1	1
114	Oriented	4.1															
	Program ming Lab	CO1211 4.2	3	2	3	1	1	3	1	-	1	-	-	3	2	3	3
		CO1211	2	2	2	2	2	3	2	-	2	-	-	1	1	3	3
		4.3															
		CO1211	2	2	2	1	2	2	2	-	2	-	-	1	1	1	3
		4.4															
		CO1211	2	2	2	3	1	3	2	-	3	-	-	3	2	3	2
		4.5															
		CO1211	2.	2.	2.	1.	1.	2.	2.	-	2.	-	-	2.2	1.8	2.2	2.4
		4	2	0	4	8	8	4	0		2						