Term Work Marks: 25 Marks (Total marks) = 15 Marks (Experiment) + 5 Marks (Assignments/tutorial/write up) + 5 Marks (Attendance)

Practical & Oral Exam: An Oral & Practical exam will be held based on the above syllabus.

Lab Code	Lab Name	Teaching Scheme (Contact Hours)			Credits Assigned			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITC304	Java Lab (SBL)	--	04	--	--	02	--	02

Lab Code	Lab Name	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITC304	Java Lab (SBL)	--	--	--	--	25	25	50

Lab Objectives:

Sr. No.	Lab Objectives
The Lab experiments aims:	
1	To understand the concepts of object-oriented paradigm in the Java programming language.
2	To understand the importance of Classes & objects along with constructors, Arrays ,Strings and vectors
3	To learn the principles of inheritance, interface and packages and demonstrate the concept of reusability for faster development.
4	To recognize usage of Exception Handling, Multithreading, Input Output streams in various applications
5	To learn designing, implementing, testing, and debugging graphical user interfaces in Java using Swings and AWT components that can react to different user events.
6	To develop graphical user interfaces using JavaFX controls.

Lab Outcomes:

Sr. No.	Lab Outcomes	Cognitive levels of attainment as per Bloom's Taxonomy
On successful completion, of course, learner/student will be able to:		
1	Explain the fundamental concepts of Java Programing.	L1, L2
2	Use the concepts of classes, objects, members of a class and the relationships among them needed for a finding the solution to specific problem.	L3
3	Demonstrate how to extend java classes and achieve reusability using Inheritance,	L3

	Interface and Packages.	
4	Construct robust and faster programmed solutions to problems using concept of Multithreading, exceptions and file handling	L3
5	Design and develop Graphical User Interface using Abstract Window Toolkit and Swings along with response to the events.	L6
6	Develop Graphical User Interface by exploring JavaFX framework based on MVC architecture.	L6

Prerequisite: Basics of Computer Programming

Hardware & Software Requirements:

Hardware Requirements	Software Requirements	Other Requirements
PC With Following Configuration 1. Intel PIV Processor 2. 2 GB RAM 3. 500 GB Harddisk 4. Network interface card	1. Windows or Linux Desktop OS 2. JDK 1.8 or higher 3. Notepad ++ 4.JAVA IDEs like Netbeans or Eclipse	1. Internet Connection for installing additional packages if required

DETAILED SYLLABUS:

Sr. No.	Module	Detailed Content	Hours	LO Mapping
0	Prerequisite	Basics of Computer Programming.	**02**	-
I	Java Fundamentals	**Overview of procedure and object oriented Programming**, Java Designing Goals and Features of Java Language. **Introduction to the principles of object-oriented programming**: Classes, Objects, Abstraction, Encapsulation, Inheritance, Polymorphism. Keywords, Data types, Variables, Operators, Expressions, Types of variables and methods. **Control Statements**: If Statement, If-else, Nested if, switch Statement, break, continue. **Iteration Statements:** for loop, while loop, and do-while loop (Perform any 2 programs that covers Classes, Methods, Control structures and Looping statements) 1) Implement a java program to calculate gross salary & net salary taking the following data. Input: empno, empname, basic Process: DA=70% of basic HRA=30% of basic CCA=Rs240/- PF=10% of basic PT= Rs100/- 2) Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same	**07**	LO1

		as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Write a Java program to take as input the speed of each racer and print back the speed of qualifying racers. 3) Write a Java program that prints all real solutions to the quadratic equation $ax^2+bx+c = 0$. Read in a, b, c and use the quadratic formula. If the discriminate b^2-4ac is negative, display a message stating that there are no real solutions? 4) Write a Menu driven program in java to implement simple banking application. Application should read the customer name, account number, initial balance, rate of interest, contact number and address field etc. Application should have following methods. 1. createAccount() 2. deposit() 3. withdraw() 4. computeInterest() 5. displayBalance() 5)Write a menu driven Java program which will read a number and should implement the following methods 1. factorial() 2. testArmstrong() 3. testPalindrome() 4. testPrime() 5. fibonacciSeries() 6) Create a Java based application to perform various ways of Method overloading.		
II	Classes, objects, Arrays and Strings	**Classes & Objects**: Reference Variables, Passing parameters to Methods and Returning parameters from the methods, Static members, Non-Static members Nested and Inner Classes. Static Initialization Block(SIB), Instance Initialization Block(IIB) **Constructors**: Parameterized Constructors, chaining of constructor, finalize() Method, Method overloading, Constructors Overloading. Recursion, Command-Line Arguments. Wrapper classes, InputBufferReader, OutputBufferReader, String Buffer classes, String functions. **Arrays & Vectors**: One and Two Dimensional arrays, Irregular arrays, dynamic arrays, Array List and Array of Object. (Perform any 3 programs that covers Classes & objects, Constructors, Command Line Arguments, Arrays/Vectors,String function and recursions). **Experiments:** 1) Write a program that would print the information (name, year of joining, salary, address) of three employees by creating a class named 'Employee'. The output should be as follows:	**07**	LO1 LO2

		Name Year of joining Address Robert 1994 64C- WallsStreat Sam 2000 68D- WallsStreat John 1999 26B- WallsStreat 2) Write a program to print the area of a rectangle by creating a class named 'Area' having two methods. First method named as 'setDim' takes length and breadth of rectangle as parameters and the second method named as 'getArea' returns the area of the rectangle. Length and breadth of rectangle are entered through keyboard. 3) Write a Java program to illustrate Constructor Chaining. 4) Create a class 'Student' with three data members which are name, age and address. The constructor of the class assigns default values name as "unknown", age as '0' and address as "not available". It has two members with the same name 'setInfo'. First method has two parameters for name and age and assigns the same whereas the second method takes has three parameters which are assigned to name, age and address respectively. Print the name, age and address of 10 students. Hint - Use array of objects. 5) Write a java programs to add n strings in a vector array. Input new string and check whether it is present in the vector. If it is present delete it otherwise add it to the vector. 6) Print the sum, difference and product of two complex numbers by creating a class named 'Complex' with separate methods for each operation whose real and imaginary parts are entered by user. 7) Write menu driven program to implement recursive Functions for following tasks. a) To find GCD and LCM b) To print n Fibonacci numbers c) To find reverse of number d) To solve 1 +2+3+4+.....+(n- l)+n 8) Print Reverse Array list in java by writing our own function.		
III	Inheritance, Packages and Interfaces.	**Inheritance**: Inheritance Basics, Types of Inheritance in Java, member access, using Super- to call superclass Constructor, to access member of super class(variables and methods), creating multilevel hierarchy, Constructors in inheritance, method overriding, Abstract classes and methods, using final, Dynamic Method Dispatch **Packages**: Defining packages, creating packages and Importing and accessing packages **Interfaces**: Defining, implementing and extending interfaces, variables in interfaces, Default Method in Interface ,Static Method in interface, Abstract Classes vs Interfaces. (Perform any 3 programs covering Inheritance, Interfaces and Packages).	**10**	LO1 LO3

Experiments

1) Create a Teacher class and derive Professor/ Associate_Professor/Assistant_Professor class from Teacher class. Define appropriate constructor for all the classes. Also define a method to display information of Teacher. Make necessary assumptions as required.

2) Create a class Book and define a display method to display book information. Inherit Reference_Book and Magazine classes from Book class and override display method of Book class in Reference_Book and Magazine classes. Make necessary assumptions required.

3) A university has two types of students — graduate students and research students. The University maintains the record of name, age and programme of every student. For graduate students, additional information like percentage of marks and stream, like science, commerce, etc. is recorded; whereas for research students, additionally, specialization and years of working experience, if any, is recorded. Each class has a constructor. The constructor of subclasses makes a call to constructor of the superclass. Assume that every constructor has the same number of parameters as the number of instance variables. In addition, every subclass has a method that may update the instance variable values of that subclass. All the classes have a function display_student_info( ), the subclasses must override this method of the base class. Every student is either a graduate student or a research student.
Perform the following tasks for the description given above using Java :

(i) Create the three classes with proper instance variables and methods, with suitable inheritance.
(ii) Create at least one parameterised constructor for each class.
(iii) Implement the display_student_info( ) method in each class.

4) An employee works in a particular department of an organization. Every employee has an employee number, name and draws a particular salary. Every department has a name and a head of department. The head of department is an employee. Every year a new head of department takes over. Also, every year an employee is given an annual salary enhancement. Identify and design the classes for the above description with suitable instance variables and methods. The classes should be such that they implement information hiding. You must give logic in support of your design. Also create two objects of each class.

		5) Consider a hierarchy, where a sportsperson can either be an athlete or a hockey player. Every sportsperson has a unique name. An athlete is characterized by the event in which he/she participates; whereas a hockey player is characterised by the number of goals scored by him/her. Perform the following tasks using Java : (i)Create the class hierarchy with suitable instance variables and methods. (ii) Create a suitable constructor for each class. (iii) Create a method named display_all_info with suitable parameters. This method should display all the information about the object of a class. (iv) Write the main method that demonstrates polymorphism. 6) Create an interface vehicle and classes like bicycle, car, bike etc, having common functionalities and put all the common functionalities in the interface. Classes like Bicycle, Bike, car etc implement all these functionalities in their own class in their own way 7) Create a class "Amount In Words" within a user defined package to convert the amount into words. (Consider amount not to be more than 100000).		
IV	Exception Handling, Multithreading, Input Output streams	**Exception Handling:** Exception-Handling Fundamentals, Exception Types, Exception class Hierarchy, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally , Java's Built-in Exceptions, Creating Your Own Exception Subclasses **Multithreaded Programming:** The Java Thread Model and Thread Life Cycle, Thread Priorities, Creating a Thread, Implementing Runnable, Extending Thread, Creating Multiple Threads, Synchronization: Using Synchronized Methods, The synchronized Statement **I/O Streams:** Streams, Byte Streams and Character, The Predefined Streams, Reading Console Input, Reading Characters, Reading Strings, Writing Console Output, Reading and Writing Files. (Perform any 3 programs that cover Exception Handling, Multithreading and I/O Streams). **Experiments:** 1) Write java program where user will enter loginid and password as input. The password should be 8 digit containing one digit and one special symbol. If user enter valid password satisfying above criteria then show "Login Successful Message". If user enter invalid Password then create InvalidPasswordException stating Please enter valid password of length 8 containing one digit and one Special Symbol.	**10**	LO1 LO3 LO4

		2) Java Program to Create Account with 1000 Rs Minimum Balance, Deposit Amount, Withdraw Amount and Also Throws LessBalanceException. It has a Class Called LessBalanceException Which returns the Statement that Says WithDraw Amount(_Rs) is Not Valid. It has a Class Which Creates 2 Accounts, Both Account Deposite Money and One Account Tries to WithDraw more Money Which Generates a LessBalanceException Take Appropriate Action for the Same. 3) Create two threads such that one thread will print even number and another will print odd number in an ordered fashion. 4) Assume that two brothers, Joe and John, share a common bank account. They both can, independently, read the balance, make a deposit, and withdraw some money. Implement java application demonstrate how the transaction in a bank can be carried out concurrently. 5) You have been given the list of the names of the files in a directory. You have to select Java files from them. A file is a Java file if it's name ends with ".java". For e.g. File- "Names.java" is a Java file, "FileNames.java.pdf" is not. **Input:** test.java, ABC.doc, Demo.pdf, add.java, factorial.java, sum.txt **Output:** tset.java, add.java, factorial.java		
V	GUI programming- I (AWT, Event Handling, Swing)	**Designing Graphical User Interfaces in Java**: Components and Containers, Basics of Components, Using Containers, Layout Managers, AWT Components, Adding a Menu to Window, Extending GUI Features **Event-Driven Programming in Java**: Event-Handling Process, Event-Handling Mechanism, Delegation Modelof Event Handling, Event Classes, Event Sources, Event Listeners, Adapter Classes as Helper Classes in Event Handling. **Introducing Swing:** AWT vs Swings, Components and Containers, Swing Packages, A Simple Swing Application, Painting in Swing, Designing Swing GUI Application using Buttons, JLabels, Checkboxes, Radio Buttons, JScrollPane, JList, JComboBox, Trees, TablesScroll pane Menus and Toolbar (Perform any 3 programs that contain AWT, Event handling and Swing to build GUI application). 1)Write a Java program to implement Swing components namely Buttons, ,JLabels, Checkboxes,	**12**	LO1 LO4 LO5

		Radio Buttons, JScrollPane, JList, JComboBox, Trees, Tables Scroll pane Menus and Toolbars to design interactive GUI. 2) Write a program to create a window with four text fields for the name, street, city and pincode with suitable labels. Also windows contains a button MyInfo. When the user types the name, his street, city and pincode and then clicks the button, the types details must appear in Arial Font with Size 32, Italics. 3) Write a Java program to create a simple calculator using java AWT elements. .Use a grid layout to arrange buttons for the digits and basic operation +, -, /, *. Add a text felid to display the results. 4) Write a Java Program to create a Student Profile form using AWT controls. 5) Write a Java Program to simulate traffic signal light using AWT and Swing Components. 6) Write a Java Program to create a color palette. Declare a grid of Buttons to set the color names. Change the background color by clicking on the color button. 7) Build a GUI program that allows the user to add objects to a collection and perform search and sort on that collection.(Hint. Use Swing components like JButton, JList, JFrame, JPanel and JOptionPane.)		
VI	GUI Programming-II (JavaFX)	JavaFX Basic Concepts, JavaFX application skeleton, Compiling and running JavaFX program,Simple JavaFX control:Label,Using Buttons and events, Drawing directly on Canvas. (Perform any one program that contains the concept of JavaFX). 1)Write a Java program to design a Login Form using JavaFX Controls. 2)Write Java program to draw various shapes on Canvas using JavaFX.	**04**	LO1 LO5 LO6

Text Books:

1. Herbert Schildt, “Java-The Complete Reference”, Tenth Edition, Oracle Press, Tata McGraw Hill Education.
2. E. Balguruswamy, “Programming with Java A primer”, Fifth edition, Tata McGraw Hill Publication
3. Anita Seth, B.L.Juneja, “ Java One Step Ahead”, oxford university press.

References:

1. D.T. Editorial Services, “Java 8 Programming Black Book”, Dreamtech Press.
2. Learn to Master Java by Star EDU Solutions
3. Yashvant Kanetkar, “Let Us Java” ,4th Edition ,BPB Publications.

Term Work:

The Term work shall consist of at least 15 practical based on the above list. The term work Journal must include at least 2 Programming assignments. The Programming assignments should be based on real world applications

which cover concepts from more than one modules of syllabus**.**

Term Work Marks: 25 Marks (Total marks) = 15 Marks (Experiment) + 5 Marks (Assignments/tutorial/write up) + 5 Marks (Attendance)

Practical & Oral Exam: An Oral & Practical exam will be held based on the above syllabus.

Course Code	**Course Name**	**Teaching Scheme (Contact Hours)**			**Credits Assigned**			
		Theory	Practical	Tutorial	Theory	Practical	Tutorial	Total
ITM301	Mini Project – 1 A for Front end /backend Application using JAVA	--	04	--	--	02	--	02

Course Code	**Course Name**	Examination Scheme						
		Theory Marks				Term Work	Pract. /Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg.				
ITM301	Mini Project – 1 A for Front end /backend Application using JAVA	--	--	--	--	25	--	25

Course Objectives

1. To acquaint with the process of identifying the needs and converting it into the problem.
2. To familiarize the process of solving the problem in a group.
3. To acquaint with the process of applying basic engineering fundamentals to attempt solutions to the problems.
4. To inculcate the process of self-learning and research.

Course Outcome: Learner will be able to…

1. Identify problems based on societal /research needs.
2. Apply Knowledge and skill to solve societal problems in a group.
3. Develop interpersonal skills to work as member of a group or leader.
4. Draw the proper inferences from available results through theoretical/ experimental/simulations.
5. Analyse the impact of solutions in societal and environmental context for sustainable development.
6. Use standard norms of engineering practices
7. Excel in written and oral communication.
8. Demonstrate capabilities of self-learning in a group, which leads to life long learning.
9. Demonstrate project management principles during project work.

Guidelines for Mini Project

- Students shall form a group of 3 to 4 students, while forming a group shall not be allowed less than three or more than four students, as it is a group activity.