

Course Code: 403
Course Title: Java Programming Language

Course Code	403																																																						
Course Title	Java Programming Language																																																						
Credits	4																																																						
Course Category	Major Course																																																						
Level of Course	300-399 (Higher Level)																																																						
Teaching per Week	4 Hrs. (3 Hours Theory + 2 Hours Practical work)																																																						
Minimum weeks per Semester	15 (Including class work, examination, preparation etc.)																																																						
Review / Revision	2023-2024																																																						
Implementation Year:	A.Y. 2024-2025																																																						
Purpose of Course	To teach Object Oriented Programming (OOP) concepts through Coding using Java as programming language.																																																						
Course Objective	1. To make students understand the syntax and Object Oriented Programming (OOP) concepts using Java. 2. To make students understand various inbuilt Java classes and their working. 3. To make students understand the importance of OOP methodology. 4. To make students understand various types of OOP techniques.																																																						
Pre-requisite	Prior Knowledge object oriented concepts.																																																						
Course Outcomes	CO1: Understand the core principles of object-oriented programming (OOP) and apply them proficiently in Java, including classes, objects, inheritance, polymorphism, and encapsulation. CO2: Develop the ability to design, implement, and test Java applications, employing OOP concepts to create modular, reusable, and maintainable code. CO3: Demonstrate competence in utilizing Java's built-in libraries and frameworks to solve real-world problems efficiently, leveraging object-oriented design patterns where applicable. CO4: Analyze and debug Java programs effectively, employing best practices in error handling, exception handling, and debugging techniques to ensure robustness and reliability. CO5: Collaborate with peers in team-based Java projects, effectively communicating ideas, contributing to code reviews, and integrating individual contributions into cohesive software solutions.																																																						
Mapping between Course Outcomes(CO) with Program Specific Outcomes(PSO)	<table><tr><td></td><td>PSO1</td><td>PSO2</td><td>PSO3</td><td>PSO4</td><td>PSO5</td><td>PSO6</td><td>PSO7</td><td>PSO8</td></tr><tr><td>CO1</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO2</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO3</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO4</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><td>CO5</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>		PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8	CO1									CO2									CO3									CO4									CO5								
	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PSO7	PSO8																																															
CO1																																																							
CO2																																																							
CO3																																																							
CO4																																																							
CO5																																																							
Course Content	Unit 1. Introduction to Java 1.1 Properties of Java 1.2 Comparison of java with C++ 1.3 Java Compiler, Java Interpreter 1.4 Identifier, Literals, Operators, Variables, Keywords, Data Types 1.5 Branching: If – Else, Switch 1.6 Looping: While, Do-while, For																																																						

	1.7 Type Casting Unit 2. Classes and Objects 2.1 Simple Class, Field 2.2 Access Controls, Object creation 2.3 Construction and Initialization 2.4 Inheritance and Polymorphism in Java 2.4.1 Data encapsulation, overriding and overloading methods 2.5 this and super keywords 2.6 Static members, static block, static class 2.7 Interfaces: 2.7.1 Introduction to Interfaces, Interface Declaration. 2.7.2 Inheriting and Hiding Concepts. 2.7.3 Inheriting, Overloading and Overriding Methods and constructors. 2.7.4 Interfaces Implementations. Unit 3. Basic Concepts of Strings and Exceptions : 3.1 Strings 3.1.1 Basic String operations, String Comparison 3.1.2 String methods (charAt(), concat(), equals(), indexOf(), isEmpty(), join(), lastIndexOf(), length(), split(), substring(), trim()) 3.1.3 StringBuffer class and its constructors. 3.1.4 StringBuffer methods : (append(), insert(), update(), delete(), reverse(), capacity()) 3.2 Introduction to Exceptions: 3.2.1 Exception Types, User defined Exception 3.2.2 Throw, Throws 3.2.3 Try, Catch and Finally Unit 4. Threads and Packages: 4.1 Thread 4.1.1 Introduction to Threads, Thread Model 4.1.2 Priority of Threads 4.2 Package Naming, Type Imports 4.2.1 Package Access, Package Contents 4.2.2 Package Object and Specification Unit 5. Data Structure Implementation using Java Class 5.1 Implementation of Data Structure using Java Class: 5.1.1 Concepts of singly and singly circular link-list 5.1.2 Singly Link List : Create, traverse, insert, delete node 5.1.3 Singly circular link list: create, traverse, insert, delete node.
Reference Books	1. Java Programming Language – Ken Arnold James Gosling, David Holmes: –Addison Wesley (Pearson Education) 2. Java – The complete reference, – Herbert Schildt: – Tata McGrawHill 3. Java 2 From Scratch: – Steven Haines: –PHI. 4. Programming in Java – E-Balaguruswamy: – Tata McGraw Hill 5. Java: How to Program: – Deitel & Deitel: – PHI
Teaching Methodology	Class Work, Discussion, Lab work, Self-Study, Seminars and/or Assignments
Evaluation Method	50% Internal assessment. 50% External assessment.