

SIDDAGANGA INSTITUTE OF TECHNOLOGY, TUMAKURU

LECTURE PLAN FOR THE ACADEMIC YEAR 2025 – 2026

Faculty	Mrs Rajani GN	Dept.	Information Science and Engineering
Class	3 rd SEM	Course	Object Oriented Programming with Java(S3ISES03)

Course objectives:

This course will enable students to:

1.	Introduction to object oriented programming.
2.	Formal introduction to Java programming language.
3.	Overall development of problem solving and critical analysis.

Course Code	Course Title	Teaching & Learning Scheme							
		Classroom Instruction (CI) (in hrs / sem)		Lab Instruction (LI) (in hrs / sem)	Self-Learning (SL) (in hrs / sem)	Term Work (TW) (in hrs / sem)	Assessment (hrs / sem)	Total no. of Hours per sem	Total Credits (C)* (Total Hours/30)
		L	T	P	SL	TW	A		
ESC	Object Oriented Programming with Java(S3ISES03)	42	-	-	14	27	7	90	3

Sl. No.	Date	Topics	Remarks
		An Overview of Java	
1	1-09-2025	Object-Oriented Programming, A First Simple Program, A Second Short Program.	
2	2-09-2025(SH)	Study hour: revision on- Object-Oriented Programming, A First Simple Program, A Second Short Program.	
3	04-09-2025	Two Control Statements, Using Blocks of Code, Lexical Issues.	
4	04-09-2025	The Java Class Libraries, Java Is a Strongly Typed Language, The Primitive Types, Integers, Floating-Point Types, Characters, Booleans	
5	08-09-2025	A Closer Look at Literals, Integer literals, Floating-point literals, Boolean literals, Character literals, string literals.	

6	09-09-2025(SH)	Study hour: revision on- Literals	
7	11-09-2025	Variables, declaring a variable, dynamic initialization, the scope and lifetime of variables,	
8	11-09-2025	Type conversion and casting, automatic type promotion in expressions	
9	15-09-2025	Arrays, one dimension array.	
10	16-09-2025(SH)	Study hour: revision on-Arrays	
11	18-09-2025	Multidimensional arrays, Alternative array declaration, Few words about strings.	
		UNIT 2 Operators	
12	18-09-2025	Arithmetic Operators, The Bitwise Operators, Relational Operators, Boolean Logical Operators	
13	22-09-2025	The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses,	
14	23-09-2025(SH)	Study hour: revision on- Operators	
15	25-09-2025	Control Statements: Java selection statements, if, Nested ifs, the if-else-if ladder	
16	25-09-2025	Switch, Nested switch statements.	
17	29-09-2025	Iteration statements: While, do-While, for	
18	30-09-2025(SH)	Study hour: revision on- Control Statements	
19	06-10-2025	Declaring loop control variables inside the loop, using the Comma	
20	09-10-2025	Some for loop Variables, the for-Each Version of the for loop, iteration over Multidimensional arrays, Applying the enhanced for	
21	09-10-2025	Nested loops, Jump statements: Using break, using continue, return	
		UNIT 3 Introducing Classes	
22	16-10-2025	Class Fundamentals, Declaring Objects, Assigning Object Reference Variables	
23	16-10-2025	Introducing Methods, Constructors, The this Keyword, Garbage Collection	
24	21-10-2025(SH)	Study hour: revision on- Class Fundamentals	
25	23-10-2025	The finalize() Method, A Stack Class	
26	23-10-2025	A Closer Look at Methods and Classes: Overloading Methods, Using Objects as Parameters, A Closer Look at Argument Passing.	
27	27-10-2025	Returning Objects, Recursion, Introducing Access Control, Understanding static, Introducing final, Arrays Revisited	
28	28-10-2025(SH)	Study hour: revision on- Returning Objects, Recursion	
29	30-10-2025	Inheritance: Inheritance basics, Member access and inheritance, A Superclass Variable Can Reference a Subclass Object.	
30	30-10-2025	Using super to Call Superclass Constructors, Creating a Multilevel Hierarchy.	
31	03-11-2025	When Constructors Are Called, Method Overriding.	

32	04-11-2025(SH)	Study hour: revision on- Constructors	
33	06-11-2025	Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, The Object Class	
		UNIT 4 Packages and Interfaces	
34	06-11-2025	Packages, Defining a Package, Finding Packages and CLASSPATH, A Short Package Example.	
35	10-11-2025	Access Protection, An Access Example, Importing Packages	
36	11-11-2025(SH)	Study hour: revision on- Packages	
37	13-11-2025	Interfaces, Defining an Interface, Implementing Interfaces, Nested Interfaces	
38	13-11-2025	Applying Interfaces, Variables in Interfaces, Interfaces Can Be Extended	
39	17-11-2025	Exception Handling: Fundamentals, Exception Types.	
40	18-11-2025	Uncaught Exceptions, Using try and catch.	
41	20-11-2025	Displaying a Description of an Exception, Multiple catch Clauses.	
42	20-11-2025	Nested try Statements :Throw, throws, finally, Java's Built-in Exceptions	
43	24-11-2025	Creating Your Own Exception Subclasses, Chained Exceptions, Using Exceptions.	
		UNIT 5 Multithreading and Generics	
44	25-11-2025	Multithreading: Thread life cycle and methods, The Main Thread, Creating a Thread	
45	01-12-2025	Creating Multiple Threads, Using isAlive() and join(), Thread Priorities	
46	02-12-2025	Synchronization, Interthread Communication.	
47	04-12-2025	Suspending, Resuming, and Stopping Threads.	
48	04-12-2025	Generics :What Are Generics?, A Simple Generics Example.	
49	08-12-2025	A Generic Class with Two Type Parameters, The General Form of a Generic Class.	
50	09-12-2025	Bounded Types, Using Wildcard Arguments.	
51	11-12-2025	Creating a Generic Method, Generic Interfaces.	
52	11-12-2025	Study hour: revision class	

Important Dates to be remembered:

Sl. No.	Important Events	Dates
1.	I – Test	13 th -15 th Oct. 2025
2.	Last date for dropping of course	21 th Oct. 2025
3.	II – Test	27 th -29 th Nov. 2025
4.	Last date for withdrawal of course	1 st Dec. 2025
5.	Last working day	13 th Dec. 2025
6.	Preparation Holidays	14 th Dec.- 19 th Dec. 2025
7.	Semester End Examination	20 th -05 th Jan. 2026

8.	Announcement of Results	17 th Jan. 2026
----	-------------------------	----------------------------

Activities to meet Teaching Learning Scheme

Sl No	Activity Planned	Number of Hours
1.	Class Room Teaching	42
2.	Study hours	14
2.	Formative Assessment [Test (2 No.) +Quiz (2 No.) + Semester End Exam]	07
3.	Activity Based Learning: Assignment Activity1: 2. Concept Review (7 hours) • Review Key Topics: ○ Summarize the key concepts covered in the previous lesson (e.g., OOP principles, data types, control structures). ○ Engage students with a quick Q&A to assess their understanding. 3. Assignment Introduction (3 hours) • Present the Assignment: ○ Explain the details of the assignment, including objectives, requirements, and deadlines. ○ Example Assignment: “Create a Simple Java Application” ▪ Students will design a console-based application that incorporates OOP principles, utilizes data types and control structures, and handles exceptions. • Key Requirements: ○ Outline specific functionalities (e.g., user input, processing data, displaying output). ○ Include grading criteria (e.g., code quality, documentation, functionality). 4. Guided Work Session (4 hours) • Independent or Group Work: ○ Allow students to start working on the assignment in class. ○ Provide support and guidance as they brainstorm ideas, outline their applications, and begin coding. • Encourage Collaboration: ○ If appropriate, encourage students to discuss ideas with peers while ensuring they complete their own work. 7. Post-Class Assignment Work • Completion of Assignment: ○ Students continue working on the assignment at home, using resources discussed in class and seeking help if needed. 8. Assessment • Evaluate the Assignment: ○ Use a rubric to assess each submission based on criteria such	14+13 = 41

	as functionality, code quality, documentation, and adherence to requirements. Seminar Activity2: Objective: Students will understand key Core Java concepts and improve their presentation and collaboration skills through seminar-style discussions. Core Java Topics Overview • Presenting Key Topics: (Hours 2) ○ Briefly introducing key topics in Core Java, such as: ▪ Object-Oriented Programming (OOP) Principles ▪ Data Types and Variables ▪ Control Structures ▪ Exception Handling ▪ Multithreading Group Formation and Topic Assignment (Hours 2) • Dividing Students into Groups: ○ Organizing students into small groups (4-5 students each). • Assigning Topics: ○ Assigning each group a specific Core Java topic to explore and present. Provide a mix of foundational and advanced topics to ensure coverage. Seminar Presentations (Hours 9) • Group Presentations: ○ Each group presents their topic to the class (5-7 minutes per group). ○ Encouraging the use of visual aids (slides, diagrams, code snippets) to enhance understanding.	
--	---	--

TEXT BOOKS

- | | |
|---|--|
| 1 | Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2017. |
|---|--|

REFERENCE BOOKS

- | | |
|---|--|
| 1 | Introduction to Java Programming (Comprehensive Version), Daniel Liang, Seventh Edition, Pearson, 2008. |
| 2 | Cay S Horstmann, "Core Java - Vol. 1 Fundamentals", Pearson Education, 10th Edition, 2016. |
| 3 | Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft, "Java 8 in Action", Dreamtech Press/Manning Press, 1st Edition, 2014. |

Course Outcomes: Upon completion of this course the student will be able to:	
CO1.	Apply basic principles of the object-oriented programming.
CO2.	Show competence in the use of the Java programming language in the development of small to medium-sized application programs.
CO3.	Demonstrate the packages and interfaces of Java.
CO4.	Apply I/O programming to solve real world programming.
CO5.	Demonstrate an introductory understanding of multithreaded programming.

Mapping of Course Outcomes with Program Outcomes

		POs												PSOs		
		1	2	3	4	5	6	7	8	9	10	11	12	1	2	3
COs	CO1	3												1		
	CO2	3												1		
	CO3	3												1		
	CO4			2										1		
	CO5			3										1		
AVG		3		3										1		

Assessment Tools	COs				
Direct AT	CO1	CO2	CO3	CO4	CO5
CIE (Individual)	✓	✓	✓	✓	✓
SEE (Individual)	✓	✓	✓	✓	✓
Assignments (Individual/Group)	--	✓	✓	✓	✓
Micro Projects (Group)	--	--	--	--	--
Topic seminar (Individual)	--	--	--	--	--
Case studies (Individual/Group)	--	✓	--	--	--
Online courses (Individual)	--	--	--	--	--
Indirect AT					
Course end survey (Students)	•	•	•	•	•
Student profile (Faculty)	--	--	--	--	--

Course delivery methods, assessment tools and sample questions:

CO1	Understand the basic principles of the object-oriented programming.
------------	---

Delivery Methods	Black board Teaching, PowerPoint Presentation
Assessment Tools	CIE- Test 1, SEE
Sample Questions	- Write a java program to print the following output. 1 1 2 1 2 3 1 2 3 4 1 2 3 4 5 - Discuss the various primitive data types used in java. - Explain lexical issues in java. - How are arrays declared and initialized in Java. Explain with suitable example. - Explain scope and lifetime of variables with an example - Explain automatic type promotion in expressions with rules and demo program. >>> ^ && ?:

CO2	Show competence in the use of the Java programming language in the development of small to medium-sized application programs.
Delivery Methods	Black Board Teaching, Group Activity, Power point Presentation.
Assessment Tools	CIE-Test 1, SEE, Assignment.
Sample Questions	- What are the different types of operators in java ? explain them. - Write simple program to demonstrate arithmetic operators. - Explain iteration statements in java. - Explain java selection statements. - Demonstrate the use of: - Continue statement in while loop. - Break statement in do while loop.

CO3	Demonstrate the packages and interfaces of Java.
Delivery Methods	Black Board Teaching, Power point Presentation
Assessment Tools	CIE: Test 2, SEE, Assignment
Sample Questions	- Explain class with an example? - What are constructors? Explain with an example. - What is inheritance? Explain different types of inheritance with suitable example. - Explain method overriding with an example? - Explain abstract class and abstract method with an example. - Explain the following:

	• This keyword • Garbage collection in java. • Finalize () method.
CO4	Apply I/O programming to solve real world programming.
Delivery Methods	Black Board Teaching, Power point Presentation
Assessment Tools	CIE: Test 2, SEE, Assignment
Sample Questions	1. What is package? how to create and import package in java. With an example. 2. What is an interface? 3. How to define and implement interface by taking suitable examples. 4. Explain Exception Types. 5. How do you create exception class? Explain with a program.

CO5	Demonstrate an introductory understanding of multithreaded programming.
Delivery Methods	Black Board Teaching, Power point Presentation
Assessment Tools	CIE: Test 3, SEE, Assignment
Sample Questions	1. How to create thread with suitable example. 2. How to create multiple threads with example. 3. Create a generic method <code>printArray(T[] array)</code> that takes an array of any type and prints its elements to the console. 4. Explain what a generic interface is in Java. How does it differ from a regular interface? Provide a simple example of a generic interface. 5. Define a generic interface <code>Processor<T></code>, which declares a method <code>process(T item)</code>. Implement this interface in a class <code>StringProcessor</code> that processes a string (e.g., converts it to uppercase).

T. J.
Faculty

H. D. 04/9/2025
HoD

P. J.
Principal
4/9/2025