GANPAT UNIVERSITY

FACULTY OF DIPLOMA ENGINEERING

Programme	Diploma Engineering	Branch	Computer Engineering/Information Technology
Semester	II	Version	1.0.0.0
Effective from Academic Year	2025-26	Effective for the batch Admitted in	JULY 2025
Course code	1CEIT2101	Course Name	Object-Oriented Programming with Java

I.TEACHING-LEARNING AND ASSESSMENT SCHEME

Course Type	Course Code	Learning Scheme						Assessment Scheme										
		Actual Contact Hrs./Week			SLH	NLH	Credits	Theory			Practical				Based on SL			Total Marks
		CL	TL	LL				FA-TH	SA-TH	TOTAL	FA-PR		SA-PR	TOTAL		SLA		
								MAX	MAX	MAX	MIN	MAX	MAX	MAX	MIN	MAX	MIN	
DSC	1CEIT2101	4	-	4	2	10	5	40	60	100	40	60	40	100	40	20	08	220

Abbreviation:	**CL**-Classroom Learning	**TL**- Tutorial Learning	**LL**-Laboratory Learning
	SLH-Self Learning Hours	**NLH**-Notional Learning Hours	**SLA**-Self Learning Assessment
	FA-Formative Assessment (Term work+Mid Sem Exam+ Attendance)		**SA**-Summative Assessment

II. PRE-REQUISITES

Basic programming knowledge (C or Python), understanding of control structures and data types.

III. INDUSTRY /EMPLOYER EXPECTED OUTCOMES

Able to design, develop, test, and deploy object-oriented Java applications using standard libraries and tools. They should be proficient with OOP concepts, exception handling, file I/O, basic GUI, and multithreading.

IV. COURSE LEARNING OUTCOMES

At the end of the course, students will be able to achieve the following course learning outcomes:

CO1: Understand and apply object-oriented principles using Java.
CO2: Implement class structures, constructors, and method overloading.
CO3: Develop programs using inheritance, interface and package.
CO4: Apply exception handling and file input output operations.
CO5: Build basic GUI and multithreading applications.

V. THEORY LEARNING OUTCOMES AND ALIGNEG COURSE CONTENT:

Name of Unit	Theory Learning Outcomes (TLOs) aligned to Cos	Learning Content mapped with TLOs & COs	Marks	Hours
Unit-1 Basics of Java	**TLO 1.1** Explain the features and structure of Java. **TLO 1.2** Define and explain object-oriented concepts (encapsulation, abstraction, inheritance, polymorphism). **TLO 1.3** Write basic Java programs using classes and methods. **TLO 1.4** Explain if…else, switch...case, for, while and do…while loop.	1.1 Java features, JDK, JRE, JVM 1.2 OOP concepts with real-life examples 1.3 Writing and executing simple programs 1.4 control and looping statements 1.5 Wrapper classes, Garbage Collection 1.6 String manipulation functions, StringBuffer class	**8**	**10**

	TLO 1.5 Explain how java automatically clean the memory and types conversion. **TLO1.6** Explain string and mathematical functions.	functions, Mathematical Functions		
Unit-2 Classes, Objects and Constructors	**TLO 2.1** Define and use user-defined classes, objects and methods. **TLO 2.2** Implement method overloading and constructors. **TLO 2.3** Use access specifiers and static members.	2.1 Class creation, instance/static variables and methods. 2.2 Constructors 2.3 Overloading, static methods, this keyword	**12**	**8**
Unit-3 Inheritance, Interface and Packages	**TLO 3.1** Explain inheritance types and their implementation. **TLO 3.2** Apply method overriding and polymorphism. **TLO 3.3** Use abstract classes and interfaces. **TLO 3.4** Explain user defined and inbuilt packages.	3.1 Inheritance types, super and final keyword 3.2 Overriding, runtime polymorphism 3.3 Abstract classes, interfaces 3.4 Package	**12**	**10**
Unit-4 Exception Handling and File I/O	**TLO 4.1** Explain types of exceptions and use try-catch-finally blocks. **TLO 4.2** Handle multiple exceptions and use throw/throws. **TLO 4.3** Explain reading and writing text and .csv files.	4.1 Checked and unchecked exceptions 4.2 try-catch-finally, throw and throws keyword 4.3 FileReader and FileWriter.	**10**	**6**
Unit-5 Collections and Threads	**TLO 5.1** Use Java Collection framework – ArrayList, HashMap. **TLO 5.2** Implement multithreading using Thread and Runnable. **TLO 5.3** Use thread methods and synchronization.	5.1 List/Map usage 5.2 Thread class & Runnable interface 5.3 Thread life cycle, sleep(), join(), synchronization	**10**	**6**
Unit-6 GUI Programming Using Java Swing	**TLO 6.1** Create GUI using Swing components. **TLO 6.2** Handle events in GUI programs. **TLO 6.3** Develop simple form-based applications.	6.1 JFrame, JLabel, JTextField, JButton 6.2 ActionListener 6.3 Layout managers, simple login form	**8**	**5**

VI. LABORATORY LEARNING OUTCOME AND ALIGNED PRACTICAL			
Sr. No.	**Practical / Laboratory Learning Outcome (LLO)**	**Practical Titles**	**Relevant COs**
1	**LLO 1.1** Understand object-oriented concepts using Java.	Write a program to demonstrate classes and objects.	CO1
2	**LLO 1.2** Use instance variables and methods.	Create a class with instance variables and methods to perform basic operations.	CO1
3	**LLO 1.3** Use of string and mathematical functions.	Write a program to use string and mathematical functions	C01
4	**LLO 2.1** Implement constructors in Java.	Demonstrate default and parameterized constructors.	CO2
5	**LLO 2.2** Demonstrate constructor overloading and method overloading.	Program for constructor overloading and method overloading.	CO2

6	**LLO 2.3** Use static variables, methods, and `this` keyword.	Create a program using static methods, static variables, and `this` keyword.	CO2
7	**LLO 3.1** Understand and implement inheritance.	Implement single, multilevel and hierarchical inheritance.	CO3
8	**LLO 3.2** Apply method overriding and polymorphism.	Create a Java program for runtime polymorphism using method overriding.	CO3
9	**LLO 3.3** Use abstract classes.	Write a program demonstrating abstract class and abstract methods.	CO3
10	**LLO 3.4** Use interfaces in Java.	Implement multiple inheritance using interfaces.	CO3
11	**LLO 4.1** Handle exceptions using try-catch-finally.	Create a program demonstrating exception handling using try-catch-finally.	CO4
12	**LLO 4.2** Use throw and throws keywords.	Write a Java program using `throw` and `throws`.	CO4
13	**LLO 4.3** Perform file read/write using FileReader/FileWriter	Program for reading and writing text files using FileReader and FileWriter.	CO4
14	**LLO 5.1** Use ArrayList in Java Collections.	Create and manipulate ArrayList for storing student records.	CO5
15	**LLO 5.2** Use HashMap for key-value operations.	Create a Java Address Book using HashMap.	CO5
16	**LLO 5.3** Implement threading using Thread class.	Write a program creating threads using Thread class.	CO5
17	**LLO 5.4** Implement threading using Runnable interface.	Create a multithreading program using Runnable interface.	CO5
18	**LLO 5.5** Use thread lifecycle methods and synchronization.	Demonstrate thread sleep, join, and synchronization.	CO5
19	**LLO 6.1** Create GUI using Java Swing. (Optional/Advanced)	Create a basic GUI application (Login form) using JFrame, JLabel, JTextField, JButton.	CO5
20	**LLO 6.2** Integrate Java collections and file I/O in a mini project. (Integration)	Mini Project: File-based Student Record System using ArrayList and FileWriter.	CO4, CO5

VII. SUGGESTED MICRO PROJECT/ASSIGNMENTS/ACTIVITIES FOR SELF LEARNING/SKILL DEVELOPMENT (SELF LEARNING)

- Create a free account on LeetCode / hackerrank, select your preferred programming language, and start with Easy problems. Solve at least 30 problems.
- Console-based Student Record Management System
- File-based Library Management using Java I/O
- Basic GUI Application using Swing (e.g., Login form)
- Multithreaded Stopwatch/Timer Application
- Application using Java Collections (e.g., Address Book)

VIII. LIST OF INSTRUMENTS / EQUIPMENT / TRAINER BOARD

- Desktop systems with JDK and Java IDE (Eclipse/NetBeans/IntelliJ)
- Projector/Smart board
- Reference code library
- Java compiler and runtime environment

IX. LIST OF REFERENCE BOOKS

Sr.No.	TITLE	AUTHOR	PUBLICATION
1	Programming with Java	E. Balagurusamy	Tata McGraw Hill
2	Core Java Volume I - Fundamentals	Cay S. Horstmann	Pearson Education
3	Java: The Complete Reference	Herbert Schildt	McGraw Hill Education

X. LINK OF LEARNING WEB RESOURCE

Sr.No.	Resource Link
1	https://nptel.ac.in/courses/106105191
2	https://www.geeksforgeeks.org/java/
3	https://www.javatpoint.com/java-oops-concepts
4	https://docs.oracle.com/javase/tutorial/java/concepts/index.html
5	https://www.youtube.com/watch?v=do0Yk_d1ixM
6	https://www.w3schools.com/java/java_oop.asp

XI. SUGGESTED WEIGHTAGE TO LEARNING EFFORTS & ASSESSMENT PURPOSE

Unit	Unit Title	Aligned COs	Learning Hours	R-Level	U-Level	A-Level	Total Marks
1	Basics of Java	CO1	10	2	3	3	8
2	Classes, Objects and Constructors	CO2	8	3	4	5	12
3	Inheritance, Interface and Packages	CO3	10	2	4	4	12
4	Exception Handling and File I/O	CO3	6	2	4	4	10
5	Collections and Threads	CO4	6	2	4	4	10
6	GUI Programming	CO5	5	2	3	5	8
	Grand Total		**45**	**13**	**22**	**25**	**60**

XII. COs AND POs AND PSOs MAPPING

Course Outcome (Cos)	Programme Outcomes (POs)							Programme Specific Outcomes (PSOs)		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PSO1	PSO2	PSO3
CO1	3	2	2	2	0	1	0	3	1	1
CO2	3	3	2	2	0	1	0	3	1	1
CO3	3	3	3	3	1	1	1	3	2	2
CO4	3	3	3	3	1	2	1	3	2	2
CO5	3	3	3	3	2	2	2	3	2	3

Legends:- *3-High; 2-Moderate/Medium; 1-Slight/Low; 0-None*