



MARRI LAXMAN REDDY INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(AN AUTONOMOUS INSTITUTION)

(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)

Accredited by NAAC with 'A' Grade & Recognized Under Section 2(f) & 12(B) of the UGC act, 1956

COURSE CONTENT

| OOPS through java | | | | | | | | |
|--|----------|---------------------|---|-----------------------|---------|------------------------|-----|-------------------|
| IV Semester: CSD / CSE / CSM / ECE | | | | | | | | |
| Course Code | Category | Hours / Week | | | Credits | Maximum Marks | | |
| | | L | T | P | | C | CIA | SEE |
| 2440511 | Core | 3 | 0 | 0 | 3 | 40 | 60 | 100 |
| | | Contact Classes: 45 | | Tutorial Classes: Nil | | Practical Classes: Nil | | Total Classes: 45 |
| Prerequisites: Problem Solving Using C and C++ | | | | | | | | |

Course Overview:

OOPs Through java makes it possible to create full reusable applications with less code and sorted development time. This course is about the fundamentals of Object-Oriented Programming (OOP) Concept and OOP-based software development methodology. It encourages modular objects for reusable code, ensure well organize and maintainable code via encapsulation, inheritance and polymorphism. OOP finds broad application in software development domains:

- Software Development
- GUI Development
- Game Development
- Database Systems
- Simulation and Modeling

Course Objectives: The students will try to learn

- Concepts and features of object-oriented programming
- Java Standard API library such as util, io, applets, GUI based controls.
- Exception handling mechanism, multithreading, packages and interfaces.
- How to use Collection framework
- Internet programming using applets and AWT.

Course Outcomes: After successful completion of the course, students should be able to

- Explain OOPS principles and core Java fundamentals including data types, control statements, arrays, operators, and string handling mechanisms.
- Apply object-oriented concepts including classes, inheritance, packages, interfaces, method overloading, overriding, and dynamic method dispatch in Java programs.
- Implement exception handling and multithreading mechanisms including synchronization and inter-thread communication in Java applications.
- Develop applications using Java Collections Framework and JDBC for database

connectivity and data manipulation.

- Design GUI-based applications using AWT and Swing components with event handling mechanisms under the delegation event model.

Module-I [10]

Principles of OOPS: OOPS Paradigm, Objects, Classes and Methods, Abstraction, Encapsulation, Inheritance, Polymorphism, Dynamic Binding.

JAVA Introduction: History of Java, Java buzzwords, data types, variables, scope and life time of variables, Type conversion and casting, arrays, operators, Operator Precedence, control statements.

Java String Handling: String Constructors, Special string operations, Character Extraction, String Comparisons, Modifying a string, String Buffer.

Module - II [9]

Classes: Class fundamentals, Declaring Objects, methods, Constructors, this keyword, garbage collection, Overloading methods and constructors, Recursion.

Inheritance, Packages and Interfaces – Inheritance basics, Using super, Creating a multilevel hierarchy, method overriding, Dynamic method dispatch, abstract classes, Using final with inheritance, Defining a package, Finding package and class path, Access protection, importing packages, Defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces.

Module –III [9]

Exception handling - Exception types, uncaught exceptions, using try and catch, Multiple catch classes, nested try statements, throw, throws and finally. Java's built-in exceptions, creating own exception sub classes.

Multithreading - Thread life cycle, Thread Creation using thread class and runnable interface, Creating multiple threads, Thread priorities, Synchronizing threads, Inter thread communication.

Module –IV [9]

Collections Framework: Overview, Collection Interfaces, Collection Classes, Accessing a collection via Iterator, Working with Maps

Java Database Connectivity: Types of Drivers, JDBC architecture, JDBC Classes and Interfaces, Basic steps in Developing JDBC Application, Creating a New Database and Table with JDBC.

Module– V [8]

GUI Programming with Swing – Introduction, limitations of AWT, MVC architecture, components, containers, Layout Manager Classes, Simple Applications using AWT and Swing.

Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes.

TEXT BOOKS:

1. Java The complete reference, 9th edition, Herbert Schildt,

McGraw Hill Education (India) Pvt. Ltd.

2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.(Principles of OOPS in UNIT-I)

REFERENCES:

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, University Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object-oriented Application Development, R. A. Johnson, Cengage Learning

ELECTRONIC RESOURCES:

1. <https://www.geeksforgeeks.org/java/java/>
2. <https://www.programiz.com/java-programming>
3. <https://www.w3schools.com/java/>
4. <https://www.codechef.com/learn/course/java>
5. <https://aws.amazon.com/what-is/java/>
6. <https://www.coursera.org/articles/what-is-java-used-for>
7. <https://www.learnjavaonline.org/>
8. <https://www.ibm.com/think/topics/java>

MATERIALS ONLINE:

1. Course template
2. Tutorial question bank
3. Tech talk and Concept Video topics
4. Open-ended experiments
5. Definitions and terminology
6. Assignments
7. Model question paper – I
8. Model question paper – II
9. Lecture notes
10. E-Learning Readiness Videos