

NIRMA UNIVERSITY
Institute of Technology
B. Tech. Computer Science and Engineering
Semester – III

L	T	P	C
2	0	4	4

Course Code	2CS302
Course Name	Object Oriented Programming

Course Learning Outcomes (CLO):

At the end of the course, students will be able to –

1. interpret the basic principles of object oriented programming
2. develop computer programs to solve real world problems based on object-oriented principles
3. implement multi-threaded applications with basic input-output operations and exception handling

Syllabus:

**Teaching
Hours:**

Unit I

02

Introduction:A Review of programming paradigms, Introduction to Object Oriented Programming, Comparison of Object Oriented approach with other programming approaches

Unit II

02

History and Overview of Java:Creation of Java, , Evolution of Java, features of Java, byte code, Java Development Kit (JDK), Java Virtual Machine (JVM) , Introduction to three OOP principles (Inheritance, Polymorphism, Encapsulation), Introduction to classes and methods

Unit III

06

Data Types, Variables and Operators in Java: Arrays: One dimensional array, multi-dimensional array, alternative array declaration statements. Control Statements like Selection statements (i.e if, switch etc.), iteration statements (i.e while, do-while, the for-each version of the for Loop, Nested Loops etc.) , jump statements (i.e break, continue)



Unit IV

10

Classes and Methods: class fundamentals, objects, assigning object reference variables, methods in class, constructors, this keyword, garbage collection, finalize() method, overloading methods, argument passing, access control, static, final, nested and inner classes, command line arguments, variable-length arguments. String Handling: Basics of String handling in Java, String class methods, String Buffer Class methods Inheritances: Basics, member access and inheritance, super class references, using super, multilevel hierarchy, constructor call sequence, method overriding, dynamic method dispatch, abstract classes, Object class Packages and Interfaces: defining a package, finding packages and CLASSPATH, access protection, importing packages, interfaces (defining, implementation, nesting, applying), variables in interfaces, extending interfaces, instance of operator

Unit V

10

Exception Handling: fundamental, exception types, uncaught exceptions, try, catch, throw, throws, finally, multiple catch clauses, nested try statements, built-in exceptions, custom exceptions (creating your own exception sub classes). Multithreaded Programming: Java thread model, thread priorities, synchronization, messaging, Thread class, Runnable interfaces, creating a thread(s), Thread class methods, Synchronization, Inter thread Communication, volatile operators. Managing I/O: Streams, Byte Streams and Character Streams, Predefined Streams, Reading console Input, Writing Console Output, PrintWriter class, File management classes

Self-Study:

The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Laboratory Work:

Laboratory work will be based on above syllabus with minimum 10 experiments to be incorporated that will be considered for evaluation.

Suggested Readings^:

1. Herbert Schildt, Java – The Complete Reference, Tata McGraw Hill
 2. Balaguruswamy, Programming with Java – A primer, Tata McGraw Hill
 3. David Flanagan, Student Workbook Java in a Nutshell, O'Reilly
 4. Cay S. Horstmann, Core Java(TM), Volume I—Fundamentals, Prentice Hall
- 28.

L = Lecture, T = Tutorial, P = Practical, C = Credit ^ This is not an exhaustive list