

**NIRMA UNIVERSITY**  
**Institute of Technology**  
**B. Tech. Computer Science and Engineering**  
**Open Elective**

| L | T | P | C |
|---|---|---|---|
| 2 | 0 | 2 | 3 |

|                    |                             |
|--------------------|-----------------------------|
| <b>Course Code</b> | XXXX                        |
| <b>Course Name</b> | 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. design and develop computer programs to solve real world problems based on object-oriented principles.
3. implement multi-threaded applications and exception handling.

**Syllabus:**

**Teaching  
Hours**  
05

**Unit I**

**Introduction:** A Review of programming paradigms, Introduction to Object Oriented Programming and principles, Comparison of Object Oriented approach with other programming approaches, Bytecode, JVM and JDK, Introduction to classes and methods

**Unit II**

**Data Types, Variables and Operators in Java:** Arrays: single and multi-dimensional arrays, alternative array declaration statements, Control Statements and loops

**Unit III**

**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 in Java, Inheritance: Basics, member access, multilevel hierarchy, constructor call sequence, method overriding, dynamic method dispatch, abstract classes

**Unit IV**

**Packages and Interfaces:** Defining a package, finding packages and CLASSPATH, access protection, importing packages, interfaces, variables in interfaces, extending interfaces, instance of operator

04

07

04

*MH*  
18.3

**Unit V****05**

**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).

**Unit VII****05**

**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.

**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 applications of above syllabus with minimum 10 experiments to be incorporated.

**Suggested Readings<sup>^</sup>:**

1. Herbert Schildt, Java – The Complete Reference, Tata McGraw Hill
2. Balagurusamy, 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

L=Lecture, T=Tutorial, P=Practical, C=Credit

---

<sup>^</sup>this is not an exhaustive list

10.3