## 4.3 OBJECT ORIENTED PROGRAMMING USING JAVA

| L | T | P |
|---|---|---|
| 4 | - | 6 |

**RATIONALE**

Object orientation is a new approach to understand the complexities of the real world. In contrast to the earlier approaches like procedural etc, object orientation helps to formulate the problems in a better way giving high reliability, adaptability and extensibility to the applications. The students are already familiar with this concept of programming in C which is the basic for JAVA. This course offers the modern programming language JAVA that will help the students to implement the various concept of object orientation practically. The students will be able to program in the object oriented technology with the usage of JAVA.

**LEARNING OUTCOMES**

After undergoing the subject, students will be able to:

- install Java IDE, Compiler, Java virtual machines
- debug and compile the program written in Java.
- explain and implement class programs.
- explain and execute the language construct concepts.
- explain and execute member functions.
- explain the concepts of OOPS
- describe and implement inheritance concepts.
- explain and implement Polymorphism using Java program.
- explain and implement the abstract class and interface.
- implement the exception handling in projects
- develop and understand multithreaded programs

**DETAILED CONTENTS**

**1**. Introduction and Features (05 Periods)

Fundamentals of object oriented programming – procedure oriented programming Vs. object oriented programming (OOP), Object oriented programming concepts – Classes, object, object reference, abstraction, encapsulation, inheritance, polymorphism, Introduction of eclipse (IDE) for developing programs in Java

2. Language Constructs (07 Periods)

variables, types and type declarations, data types : Integer, floating point type, character, boolean, all Operators, iteration and jump statement, if then else clause; conditional expressions, input using scanner class and output statement, loops, switch case, arrays, methods.

3. Classes and Objects (08 Periods)

Class fundamentals, constructors, declaring objects (Object & Object Reference), creating and accessing variables and methods, static and non static variables/methods defining packages, Creating and accessing a package, Importing packages, Understanding CLASSPATH, auto boxing , String , String Buffer

4. Inheritance (06 Periods)

Definition of inheritance, protected data, private data, public data, constructor chaining, order of invocation, types of inheritance, single inheritance, multilevel inheritance, hierarchical inheritance, hybrid inheritance , access control (Private Vs PublicVs Protected Vs Default)

5. Abstract Class and Interface (08 Periods)

Defining an interface, difference between classes and interface, Key points of Abstract class & interface, difference between an abstract class & interface, implementation of multiple inheritance through interface.

6. Polymorphism (06 Periods)

Method and constructor overloading, method overriding, up-casting and down-casting.

7. Exception Handling (07 Periods)

Definition of exception handling, implementation of keywords like try, catches, finally, throw& throws, built in exceptions, creating own exception sub classes importance of exception handling in practical implementation of live projects

8. Multithreading (09 Periods)

Difference between multi threading and multi tasking, thread life cycle, creating threads, thread priorities, synchronizing threads.

**LIST OF PRACTICALS**

1. WAP to create a simple class to find out the area and perimeter of rectangle and box using super and this keyword.
2. WAP to design a class account using the inheritance and static that show all function of bank (withdrawal, deposit).
3. WAP to design a class using abstract methods and classes.

4. WAP to design a string class that perform string method (equal, reverse the string, change case).
5. Consider we have a Class of Cars under which Santro Xing, Alto and Wagon R represents individual Objects. In this context each Car Object will have its own, Model, Year of Manufacture, Colour, Top Speed, etc. which form Properties of the Car class and the associated actions i.e., object functions like Create(), Sold(), display() form the Methods of Car Class.
6. In a software company Software Engineers, Sr. Software Engineers, Module Lead, Technical Lead, Project Lead, Project Manager, Program Manager, Directors all are the employees of the company but their work, perks, roles, responsibilities differs. Create the Employee base class would provide the common behaviors of all types of employee and also some behaviors properties that all employee must have for that company.
7. Using the concept of multiple inheritance create classes: Shape, Circle, Square, Cube, Sphere, Cylinder. Your classes may only have the class variable specified in the table below and the methods Area and/or Volume to output their area and/or volume.

| Class | Class Variable | Constructor | Base class |
|---|---|---|---|
| Shape | String name | Shape() | |
| Circle | double radius | Circle( double r, String n ) | Shape |
| Square | double side | Square( double s, String n ) | Shape |
| Cylinder | double height | Cylinder(double h, double r, String n ) | Circle |
| Sphere | None | Sphere( double r, String n ) | Circle |
| Cube | None | Cube( double s, String n ) | Square |

8. WAP to handle the exception using try and multiple catch block.
9. WAP that implement the Nested try statements.
10. WAP to create a package that access the member of external class as well as same package.
11. WAP that show the partial implementation of interface.
12. WAP to create a thread that implement the Runnable interface.

**INSTRUCTIONAL STRATEGY**

The subject is totally practical based. Students should be given clear idea about the basic concepts of programming. In practical session student should be asked to draw flow chart

write algorithm and then write program for algorithm and run on computer. It is required that students should maintain records (files with printouts).

## MEANS OF ASSESSMENT

- Assignments and quiz/class tests, mid-term and end-term written tests
- Actual laboratory and practical work, exercises and viva-voce
- Software installation, operation, development and viva-voce

**RECOMMENDED BOOKS**


1. Programming with Java: A Primer; E. Balagurusamy
2. Head First Java, O-REILLY, Kathy Sierra & Bert Bates.
3. OCA Java SE Programmer I Certification Guide , Wiley Publisher , Mala Gupta
4. PROGRAMMER'S GUIDE TO JAVA SE 8 , Pearson , Khalid E Mughal
5. e-books/e-tools/relevant software to be used as recommended by AICTE/UPBTE/NITTTR.


**Websites for Reference:**http://www.spoken-tutorial.org , http://swayam.gov.in

**SUGGESTED DISTRIBUTION OF MARKS**

| Topic No. | Time Allotted (Periods) | Marks Allotted (%) |
|---|---|---|
| 1. | 5 | 14 |
| 2. | 7 | 12 |
| 3. | 8 | 13 |
| 4. | 6 | 13 |
| 5. | 8 | 13 |
| 6. | 6 | 12 |
| 7. | 7 | 12 |
| 8. | 9 | 11 |
| **Total** | **56** | **100** |