OBJECT ORIENTED PROGRAMMING THROUGH JAVA

(Common to CSE, IT & ECE)

Course Code :13CT1111

L	T	P	C
4	0	0	3

Course Educational Objectives:

The main objective of the course is to expose the students to object oriented programming principles and how these principles can be applied with JAVA programming language. Upon completion of the course, the student should be able to:

- Understand fundamentals of programming such as variables, conditional and
- iterative execution, methods, etc.
- Understand fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
- Have the ability to write a computer program using JAVA to solve specified problems.
- Have the ability to write multithread programs.
- Develop GUI based applications.
- Write small network based programs.

Course Outcomes:

At the end of the course the student will be able to

- Learn a new way of approaching the job of programming.
- Employ techniques for developing robust, reusable software.
- Learn the concept of algorithm design and implementation.
- Write Java codes using both console or command-line and dialog box or graphical user interface styles.
- Write, compile, execute, and debug their Java programs.

UNIT-I (12 Lectures)

FUNDAMENTALS OF OBJECT-ORIENTED PROGRAMMING:

Introduction, Object-Oriented Paradigm, Basic concepts of Object-Oriented Programming, Benefits of Object-Oriented Programming, Applications of Object-Oriented Programming

THE HISTORY AND EVOLUTION OF JAVA:

creation of Java, Java's Bytecode, Javas buzzwords, evolution of Java. An overview of Java- Simple Java Program. Date types, variables, automatic type conversion, Arrays, operators,expressions, control statements.

UNIT-II (12 Lectures)

INTRODUCING CLASSES:

Class fundamentals, declaring objects, assigning object reference variables, introducing methods- overloading methods,argument passing,recursion, access control,static keyword, final keyword, using command line arguments, variable length arguments.

Constructors, this keyword, garbage collection, finalize() method.

STRING HANDLING:

String class, String Buffer class, StringBuilder class.

INHERITANCE:

Inheritance basics, using super, creating a multilevel hierarchy, how constructors are called, Method overriding, dynamic method dispatch, using abstract classes, using final with inheritance, the Object class.

UNIT-III (14 Lectures)

PACKAGES AND INTERFACES: Packages, access protection, importing packages, interfaces.

Exploring java.lang package: Wrapper classes, Math class.

Exploring java.util package: Vector, Scanner, Date, Calendar, StringTokenizer, Random.

Exploring java.io package: Byte streams, Character streams, File, RandomAccessFile.

EXCEPTION HANDLING:

Exception-handling fundamentals, Exception types, uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws, finally, Java's built-in exceptions, creating your exception subclasses, using exceptions.

MULTITHREADING:

Java thread model, Main thread, creating a thread, creating multiple threads, using isAlive() and join(), thread priorities, synchronization, Interthread communication, suspending, resuming and stopping threads, using multithreading.

UNIT-IV (12 Lectures)

APPLETS:

Applet basics, architecture, skeleton, simple applet display methods, repainting, status window, HTML applet tag, passing parameters to applets.

AWT:

AWT classes, window fundamentals, working with frame windows, creating a frame window in an applet, creating a windowed program, displaying information within a window, working with graphics, working with color, working with fonts, AWT control fundamentals, Labels, using buttons, applying checkboxes, checkboxgroup, choice controls, using lists, scrollbars, textfield, text area, using layout managers, Menu bars and menus, dialog boxes.

UNIT-V (10 Lectures)

EVENT HANDLING:

Two event handling mechanisms, delegation event model, event classes, sources of events, event listeners interfaces, using the delegation event model, adapter classes, inner classes, handling events by extending AWT components.

SWINGS:

origin of swings, swings built on AWT, two key swing features, MVC architecture, components and containers, swing packages, simple swing application, event handling, painting in swing, JLabel, JTextField, JTabbedPane, JScrollPane,JList,JComboBox,Trees,JTable.

NETWORKING:

Basics,networking classes and interfaces,InetAddress,TCP/IP,URL

TEXT BOOKS:

1. E.Balaguruswamy, *"Programming with Java A Primer"*, 4th Edition, TataMcGraw-Hill, 2009.
2. Herbert Schildt, *"Java The complete reference"*, 8thEdition, McGrawHill, 2011.

REFERENCES:

1 Timothy budd, *"An introduction to object-oriented programming"*, 3rdEdition, Pearson Education, 2009.

2. Y. Daniel Liang, "*Introduction to Java programming"*, 9thEdition, Pearson education, 2012.
3. Ivor Horton, *"Beginning Java"*, 7thEdition, Wrox Publications, 2011.
4. Cay.S.Horstmann and Gary Cornell "*Core Java 2, Vol I, Fundamentals"*, 9thEdition, Pearson Education, 2012.
5. Cay.S.Horstmann and Gary Cornell, "*Core Java 2, Vol II, Fundamentals"*, 9thEdition, Pearson Education, 2012.