

**CP303: OBJECT ORIENTED PROGRAMMING USING JAVA**  
**CREDITS = 6 (L=4, T=0, P=2)**

**Course Objective:**

To impart knowledge of principles of object oriented programming paradigm using JAVA to develop applications.

**Teaching and Assessment Scheme:**

Teaching Scheme			Credits	Assessment Scheme				Total Marks
L	T	P	C	Theory		Practical		
				ESE	CE	ESE	CE	
4	0	2	6	70	30	30	20	150

**Course Contents:**

Unit No.	Topics	Teaching Hours
1	<b><u>Basics of Java:</u></b>  Introduction, Features of Java, byte code and JVM, three OOP principles (Inheritance, Polymorphism, Encapsulation), lexical issues, Data types, Operator, Control and loop Statements, Class Fundamentals, Object and Object reference, Object Life time and Garbage Collection, Creating Objects, Constructor and initialization code block, Access Control, Modifiers, Nested class, Inner Class, Anonymous Classes, Abstract Class and Interfaces, Defining Methods, Method Overloading, Dealing with Static Members, Use of “this” reference, Use of Modifiers with Classes & Methods, Generic Class Types.	08
2	<b><u>Array and String:</u></b>  Single and Multidimensional Array, String class, String Buffer class, Operations on string, Command line argument, Use of Wrapper Class.	03
3	<b><u>Inheritance and Interfaces:</u></b>  Use of Inheritance, Inheriting Data members and Methods, constructor in inheritance, Multilevel Inheritance, method overriding, Handle multilevel constructors – <i>Super</i> keyword, <i>Final</i> keywords, Creation and Implementation of an interface, Interface reference, instance of operator, Interface inheritance, Dynamic method dispatch, Understanding of Java Object Class, Comparison between Abstract Class and interface.	08

Unit No.	Topics	Teaching Hours
4	<b><u>Packages and Exception Handling:</u></b>  Defining a package, finding packages and CLASSPATH, access protection, importing packages, java.lang package, Exception types, uncaught exceptions, try, catch, throw, throws, finally, multiple catch clauses, nested try statements, built-in exceptions, custom exceptions, Generic Programming – Generic class and Generic methods.	10
5	<b><u>Multithreading:</u></b>  Java Thread Model, Use of Multithread programming, Thread class and Runnable interface, Thread priority, Thread synchronization, Thread communication, Deadlock.	08
6	<b><u>Managing I/O Operation:</u></b>  Streams and the new I/O Capabilities, Understanding Streams, The Classes for Input and Output, The Standard Streams, Working with File Object, File I/O Basics, Reading and Writing to Files, Buffer and Buffer Management, Read/Write Operations with File, Channel, Serializing Objects.	08
7	<b><u>Applet, Event Handling and GUI design using Swing:</u></b>  Applet Fundamental, Applet Architecture, Applet Skeleton, Requesting Repainting, status window, HTML Applet tag, passing parameters to Applets, various event handling mechanisms, Delegation Event Model, Events, Event Sources, Event Listeners, various classes related to event sources and event listeners, basics of AWT and JFC, Introduction To Swing, MVC Architecture, Applets, Applications and Pluggable Look and Feel, Basic swing components : Text Fields, Buttons, Toggle Buttons, Checkboxes, and Radio Buttons.	15
<b>TOTAL</b>		<b>60</b>

#### List of References:

1. Herbert Schildt, “*The Complete Reference, Java*”, McGraw-Hill.
2. E.Balaguruswamy, “*Programming with Java A Primer*”, McGraw-Hill.
3. Horstmann & Cornell, “*Core Java*” Volume-I Fundamentals, Pearson Education.
4. Joseph O’Neil, “*Teach yourself Java*”, TMH publication.
5. Patrick Niemeyer and Jonathan Knudsen, “*Learning Java*”, O’reilly Media.

#### Course Outcomes (COs):

At the end of this course students will be able to ...

1. Implement object oriented programming concepts in Java.
2. Demonstrate use of Multithreading in Java application.
3. Design application using event based GUI handling principles.

4. Design an application incorporating features of Generic programming
5. Develop application incorporating features like Package, Exception Handling, I/O handling.
6. Enhance logical reasoning and programming skills.