

INSTRUCTIONS TO PAPER SETTERS:

1. Question No. 1 should be compulsory and cover the entire syllabus. There should be 10 questions of short answer type of 2.5 marks each, having at least 2 questions from each unit.
2. Apart from Question No. 1, rest of the paper shall consist of four units as per the syllabus. Every unit should have two questions to evaluate analytical/technical skills of candidate. However, student may be asked to attempt only 1 question from each unit. Each question should be 12 marks including subparts, if any.

OBJECTIVE: *In this course student will become familiar with features of Java language, they will learn how to write Java code according to Object-Oriented Programming principles, how to design GUI applications using AWT, how to develop multithreaded and Networking applications and how to create dynamic pages.*

PRE-REQUISITES

- Basic Object Oriented Programming Concepts
- [Pre-requisite based Study Material](#) (Website course Material)
- Java Programming, NPTEL by Dr. Debasis Samanta, IIT Kharagpur (12 Weeks), <https://nptel.ac.in/courses/106/105/106105191/>

COURSE OUTCOMES

After the completion of the theory course, the students will be able to learn-

CO1	Illustrate the Object-Oriented paradigm, Java language constructs and JVM internal architecture. (BTL2)
CO2	Apply the concepts of exception handling, multithreading, and collection framework. (BTL3)
CO3	Analyze the use of event handling and JFC based toolkit in creating GUI-based computing solutions.(BTL4)
CO4	Design database enabled client-server applications using JDBC, RMI, I/O operations, network programming and relevant concepts. (BTL6)
CO5	Elaborate the functional programming concepts introduced in Java 8 and beyond.(BTL6)

UNIT - I

OOP Paradigm: Comparison of programming paradigms, Characteristics of Object-Oriented Programming Languages, Object-based programming languages

Java Fundamentals: Brief History of Java, Structure of a Java program, Importance and features of Java, Introduction to JVM and its architecture including set of instructions. Overview of JVM Programming. Internal and detailed explanation of a valid .class file format. Instrumentation of a .class file, Bytecode engineering libraries, Overview of class loaders and Sandbox model of security. Basic language construct of Java-including keywords, constants, variables, operators, looping and

decision-making construct

Implementation of OOPs concepts in Java: Objects, Classes and their implementation, Encapsulation, Data Abstraction, Inheritance, Polymorphism, Dynamic Binding, Message Passing, default parameter values, using reference variables with function

Arrays and String: creating an array, one and two-dimensional arrays, String arrays and methods, Classes: String and String Buffer classes, Wrapper classes: Basics types, using super, multilevel hierarchy abstract and final classes, Object class, packages and interfaces, access protection, extending interfaces, packages
[No. of Hrs.: 12]

UNIT - II

Exception Handling: Fundamentals exception types, caught and uncaught exceptions, throw, throws, built in exceptions, custom exceptions

Multithreaded Programming: Multithreading basics, Happens-before ordering, Java Thread Model: priorities, synchronization, messaging, thread class, Runnable interface, inter-thread communication, suspending, resuming, and stopping threads.

The Collection Framework: The Collection Interface, Collection Architecture in Java, Collection Classes, traversing Collections, working with Maps & Sets

Networking fundamentals: networking classes and interfaces, using java.net package, TCP/IP, and Data-gram Programming.
[No. of Hrs.: 12]

UNIT - III

Anonymous Classes and Inner classes in Java: Core concept and its implementation and types of anonymous classes, nested and inner classes, and their implementation

Event Handling: Different Mechanism, the Delegation Event Model, Event Classes, Event Listener Interfaces, Adapter and Inner Classes, working with windows, Graphics and Text, using AWT controls, Layout managers and menus, handling Images, animation, sound and video.

Swing: Introduction to JFC (Java Foundation Classes), features of Swing, comparison with AWT, Advanced Control
[No. of Hrs.: 10]

UNIT - IV

JDBC: Introduction to DBMS & RDBMS, DBC API, JDBC Application Architecture, Obtaining a Connection, JDBC Models: Two Tier and Three Tier Model, Result Set, Prepared Statement, Callable Statement.

Input/output Programming: Basics, Streams, Byte and Character Stream, predefined streams, Reading and writing from console and files.

Java 8 Concepts: Default and Functional Interfaces, Lambda Expression, Java stream API and Pipelines, Try with Resources, Java 8 Memory optimization.

RMI (Remote Method Invocation): Introduction, Steps in creating a Remote Object, Generating Stub & Skeleton, RMI Architecture, RMI packages.
[No. of Hrs.: 10]

TEXT BOOKS:

1. Herbert Schildt, "Java - The Complete Reference", Oracle Press, 9th Edition, 2014. [HS]
2. Kathy Sierra and Bert Bates, "Head First Java", O'Reilly Publications, 2nd Edition, 2005. [KSB]

REFERENCES:

1. E. Balaguruswamy, "Programming with Java", Tata McGraw Hill, 4th Edition, 2009.
2. Cay Horstmann, "Computing Concepts with Java 2 Essentials", John Wiley & Sons, 2nd Edition, 1999. **[CH]**
3. Decker and Hirshfield, "Programming Java: An Introduction to Programming using JAVA", Vikas Publication, 2nd Edition, 2000.
4. N. P. Gopalan and J. Akilandeswari, "Web Technology - A Developers' Perspective", PHI, 2nd Edition, 2014.
5. Eric Jendrock, Jennifer Ball and Debbie Carson, "The Java #EE5 Tutorial", Pearson, 3rd Edition, 2007.
6. Daniel Liang, "Introduction to Java Programming", Pearson, 7th Edition, 2010.
7. Bill Vanners, "Inside Java Virtual Machine", Tata McGraw Hill, 2nd Edition, 2000. **[BV]**
8. Shelley Powers, "Dynamic Web Publishing", Techmedia, 2nd Edition, 1997.

OBJECTIVE: *In this course student will become familiar with features of Java language, they will learn how to write Java code according to Object-Oriented Programming principles, how to design GUI applications and Applets using AWT, how to develop multithreaded and Networking applications and how to create dynamic pages.*

COURSE OUTCOMES

After the completion of the practical course, the students will be able to learn-

CO1	Apply Object-Oriented and Java language constructs for creating Java programs. (BTL3)
CO2	Make use of exception handling, multithreading, and collection framework for constructing effective solutions. (BTL3)
CO3	Inspect the use of event handling and JFC based toolkit for GUI-based computing solutions. (BTL4)
CO4	Design database enabled client-server applications using JDBC, RMI, I/O operations, network programming and relevant concepts. (BTL6)
CO5	Elaborate the functional programming concepts introduced in Java 8 and beyond. (BTL6)