



# Java Courses

---

## Java Courses

### Course Description

This course teaches students how to develop Java applications. Topics covered include the Java programming language syntax, OO programming using Java, exception handling, file input/output, threads, collection classes, and networking. Students will develop and test Java applications (typically) using Eclipse. This course is a pre-requisite to all Application Server courses, and speciality Java Technology courses such as Struts, Spring, and Hibernate.

### Course Summary

In this introductory course, you'll learn and practice essential computer science concepts using the Java programming language. You'll learn about Object Oriented Programming, a technique that allows you to use code written by other programmers in your own programs. You'll put your new Java programming skills to the test by solving real-world problems faced by software engineers.

### Why This Course?

Java is one of the most popular programming languages used by software developers today. It is the core language used in developing Android apps, and is also commonly used in back-end web development. If you're new to programming and want to enter either of these fields, this course is a great place to get started.

Even if you don't have a career trajectory in mind, Java programming is a great option for first-time coders due to its popularity and ease of use. This course will provide you with a solid foundation in computer science and Object Oriented Programming concepts, as well as set you on the path for success as a software engineer.

### Benefits

- Compile and run a Java application.
- Understand the role of the Java Virtual Machine in achieving platform independence.
- Navigate through the API docs.
- Use the Object Oriented paradigm in Java programs.
- Understand the division of classes into Java packages.
- Use Exceptions to handle run time errors.
- Select the proper I/O class among those provided by the JDK.
- Use threads in order to create more efficient Java programs.



## Course Outline

- **Chapter 1: Introduction**
  1. What is Java?
  2. History
  3. Versioning
  4. The Java Virtual Machine
  5. Writing a Java Program
  6. Packages
  7. Simple Java Programs
- **Chapter 2: Language Components**
  1. Primitive Data Types
  2. Comments
  3. The for Statement
  4. The if Statement
  5. The while and do while Statements
  6. The switch Statement
  7. The break Statement
  8. The continue Statement
  9. Operators
  10. Casts and Conversions
  11. Keywords
- **Chapter 3: Object-Oriented Programming**
  1. Defining New Data Types
  2. Constructors
  3. The String Class
  4. String Literals
  5. Documentation
  6. Packages
  7. The StringBuffer Class
  8. Naming Conventions
  9. The Date Class
  10. The import Statement
  11. Deprecation
  12. The StringTokenizer Class
  13. The DecimalFormat Class
- **Chapter 4: Methods**
  1. Introduction
  2. Method Signatures
  3. Arguments and Parameters
  4. Passing Objects to Methods
  5. Method Overloading
  6. Static Methods



7. The Math Class
8. The System Class
9. Wrapper Classes
- **Chapter 5: Arrays**
  1. Introduction
  2. Processing Arrays
  3. Copying Arrays
  4. Passing Arrays to Methods
  5. Arrays of Objects
  6. The Arrays Class
  7. Command Line Arguments
  8. Multidimensional Arrays
- **Chapter 6: Encapsulation**
  1. Introduction
  2. Constructors
  3. The this Reference
  4. Data Hiding
  5. Public and private Members
  6. Access Levels
  7. Composition
  8. Static Data Members
- **Chapter 7: Inheritance & Polymorphism**
  1. Introduction
  2. A Simple Example
  3. The Object Class
  4. Method Overriding
  5. Polymorphism
  6. Additional Inheritance Examples
  7. Other Inheritance Issues
- **Chapter 8: Abstract Classes and Interfaces**
  1. Introduction
  2. Abstract Classes
  3. Abstract Class Example
  4. Extending an Abstract Class
  5. Interfaces
- **Chapter 9: Exceptions**
  1. Introduction
  2. Exception Handling
  3. The Exception Hierarchy
  4. Checked Exceptions
  5. Advertising Exceptions with throws
  6. Developing Your Own Exception Classes
  7. The finally Block



- **Chapter 10: Input and Output in Java**
  1. Introduction
  2. The File Class
  3. Standard Streams
  4. Keyboard Input
  5. File I/O Using Byte Streams
  6. Character Streams
  7. File I/O Using Character Streams
  8. Buffered Streams
  9. File I/O Using a Buffered Stream
  10. Keyboard Input Using a Buffered Stream
  11. Writing Text Files
- **Chapter 11: Threads**
  1. Threads vs. Processes
  2. Creating Threads by Extending Thread
  3. Creating Threads by Implementing Runnable
  4. Advantages of Using Threads
  5. Daemon Threads
  6. Thread States
  7. Thread Problems
  8. Synchronization
- **Chapter 12: Collections**
  1. Introduction
  2. Vectors
  3. Hashtables
  4. Enumerations
  5. Properties
  6. Collection Framework Hierarchy
  7. Lists
  8. Sets
  9. Maps
  10. The Collections Class
- **Chapter 13: Networking**
  1. Networking Fundamentals
  2. The Client/Server Model
  3. InetAddress
  4. URLs
  5. Sockets
  6. A Time-of-Day Client
  7. Writing Servers
  8. Client/Server Example



# Java Courses

---

## J2EE Course

### General Definitions and Background

- Introductions and Definitions
- Architectural Building Blocks: Components, Technologies and Platforms
- Multi-Tier Architectures in the Enterprise
- RMI: Distributed Computing with RMI
- Exercises

### Java 2 Enterprise Edition in detail

- Java 2 Platform, Enterprise Edition (J2EE)
- Enterprise Java and Supporting Technologies
- Business Components: Enterprise Java Beans
- Web Components: Servlets and JSPs
- Exercises

### J2EE Supporting Technologies

- JNDI – Java Naming and Directory Interface
- Java Messaging Services- JMS
- Exercises

### Design Considerations

- J2EE Connector Architecture and Legacy Systems in the Enterprise
- Design Consideration and Tradeoff
- Performance and Scalability Considerations
- Security Considerations in Enterprise Systems
- Availability Considerations in the Enterprise
- Exercises
- Manageability and Other Considerations in the Enterprise
- J2EE Design Patterns
- J2EE and Web services in the Enterprise
- Summary and J2EE Future Evolution

### ❖ Hands-on Assignments

The key concepts taught will be immediately followed up by the hands-on assignments in the lab to facilitate ease of understanding.