V1.0

Training Program on

Java Essentials/ Basics

4P Advisory Services

www.4pa.in

Java Essentials Training Program

What is Java?

Java is a general-purpose, high-level programming language originally developed by Sun Microsystems and now owned by Oracle Corporation. It is designed to be platform-independent, meaning that the same Java code can be run on multiple platforms, such as Windows, Mac OS, and Linux, without needing to be recompiled. This is achieved through the use of the Java Virtual Machine (JVM), which interprets the compiled Java code and executes it on the host system.

Java is widely used for developing a variety of applications, from web applications and mobile apps to enterprise software and games. It is an object-oriented language, which means that it is based on the concept of objects, which encapsulate data and behaviour and can be manipulated through methods. Java also supports features such as garbage collection, automatic memory management, and exception handling, which help make it a popular choice for large-scale, mission-critical applications. Additionally, Java has a vast standard library and a large community of developers, which makes it easy to find resources and support for programming in Java.

Advantages of Java as a programming language:

- Platform independence: One of the biggest advantages of Java is its platform independence. Java programs can run on any platform without needing to be recompiled, making it a versatile language that can be used on a variety of operating systems.
- Object-oriented programming support: Java is an object-oriented programming language, which means it supports the concepts of encapsulation, inheritance, and polymorphism. This makes it easier to build complex software systems and maintain them over time.
- Large standard library: Java comes with a large standard library that provides a wide range of functionality, from basic data types and control structures to advanced networking and security features.
- Memory management: Java uses a garbage collector to manage memory, which makes it easier to write robust and error-free code. The garbage collector automatically frees up memory that is no longer in use, preventing common memory-related errors such as buffer overflows and memory leaks.
- Security: Java is designed with security in mind, and includes features such as bytecode verification, class loaders, and a security manager to prevent unauthorized access and protect against malware.
- Community support: Java has a large and active community of developers, which means there are plenty of resources available for learning, troubleshooting, and sharing knowledge.

Overall, Java is a powerful and versatile programming language that offers many benefits for developers looking to build robust, secure, and cross-platform software applications.

.

Project Management, Agile, Service Management, Devops, .NET, SQL, AI/ML, Excel,DBMS, and More

Audience

The Java Fundamentals training is a 5-days course designed for:

- Software Professionals
- Developers working in an IT environment
- Professionals who are new to Java programming or who want to refresh their knowledge of Java fundamentals.
- Software developers who want to transition from another programming language to Java and need to learn the basics of Java programming language.
- IT professionals who want to understand the basics of Java programming language to support Java-based applications and systems.
- Anyone who is interested in building a career in software development and wants to learn a popular and versatile programming language like Java.

Overall, the training program is suitable for anyone who wants to learn Java programming language from scratch or wants to strengthen their existing knowledge of Java fundamentals.

Learning Objectives:

- Understand the basics of Java programming language
- Write basic Java programs using data types, variables, control structures, and arrays
- Understand object-oriented programming concepts such as classes, objects, inheritance, and polymorphism
- Write Java programs using object-oriented programming concepts
- Understand file input/output operations and exception handling in Java
- Understand multi-threading and networking concepts in Java
- Write Java programs using advanced concepts such as collections, generics, and streams
- Understand the principles of software development and apply them to Java programming
- Gain hands-on experience through practical activities and exercises to reinforce learning.

Overall, the training program aims to equip participants with a strong foundation in Java programming language and the skills and knowledge needed to build Java-based applications and systems.

Candidate Prerequisites

- Basic computer skills: Participants should be comfortable using a computer and have a basic understanding of computer programming concepts.
- Basic programming knowledge: Participants should have a basic understanding of programming concepts such as variables, data types, and control structures.
- Basic understanding of software development: Participants should have a basic understanding of software development concepts such as software requirements, design, and testing.
- Familiarity with any programming language: Although not mandatory, participants with some programming knowledge or experience in any programming language will find it easier to learn Java programming.

It is recommended that participants discuss their background and experience with the trainer to ensure that they are prepared for the training program.

Infrastructure Prerequisites

Software:

Java Development Kit (JDK) 11 or higher installed on the computer
Integrated Development Environment (IDE) such as Eclipse, NetBeans, or IntelliJ IDEA for Java development
Git version control system for managing code files
Maven build automation tool for Java projects
Any database software for Java connectivity such as MySQL, PostgreSQL, or Oracle.

Hardware:

A computer or laptop with at least 8GB RAM and a dual-core processor for running Java development software and tools
Sufficient storage capacity to store project files and data.

Cloud Infrastructure:

Optional, but beneficial for learning cloud-based Java development concepts
Access to cloud platforms such as Amazon Web Services (AWS), Microsoft Azure, or Google Cloud Platform (GCP)
Knowledge of cloud concepts such as virtual machines, storage, and networking.

Overall, participants should have access to a computer with the necessary software and hardware requirements to complete the exercises and activities within the training program. Cloud infrastructure is optional but recommended for learning cloud-based Java development concepts.

Training Outline:

Note: The duration and the contents are approximate. The trainer may change the approach and the duration, based on the response of the participants.

Day 1

Session 1: Introduction to Java and Basic Syntax

- Pre-Training Test and Introduction
- Introduction to Java and its history
- Setting up the development environment
- Basic syntax and data types in Java
- Operators and expressions

Session 2: Control Structures and Arrays

- Control structures in Java (if, switch, loops)
- Arrays and their use
- Practical activity: Write a program that prompts the user for a temperature in Celsius and converts it to Fahrenheit.

Day 2

Session 3: Object-Oriented Programming in Java

- Introduction to object-oriented programming (OOP) concepts
- Classes and objects in Java
- Constructors and methods

Session 4: Inheritance and Polymorphism

- Encapsulation and access modifiers
- Inheritance in Java
- Practical activity: Create a simple game that demonstrates inheritance.

Day 3:

Session 5: Exception Handling

- Exception handling in Java
- Types of exceptions and how to handle them

Session 6: Input/Output Operations

- File input/output operations in Java
- Streams and readers/writers in Java
- Buffered input/output streams
- Practical activity: Write a program that reads a text file and counts the number of words in it.

Day 4:

Session 7: Collections

- Introduction to collections in Java
- ArrayList, LinkedList, and Vector
- HashSet, LinkedHashSet, and TreeSet

Session 8: Generics

- Generics in Java
- Type parameters and wildcards
- Practical activity: Write a program that uses collections and generics to store and display a list of employee objects.

Day 5:

Session 9: Multi-threading and Synchronization

- Multi-threading in Java
- Synchronization and locking

Session 10: Concurrency and Reflection

- Concurrency and thread safety
- Reflection in Java
- Annotations and their use
- Practical activity: Write a program that demonstrates multi-threading and reflection in Java.
- Post-Training Test and feedback

OPTIONAL PROJECT

**Note: Additional Days Will Be Needed
Depending on the organization, the project may change**

Project: Employee Management System

Description:

Implement an Employee Management System using Java that allows the user to perform CRUD (Create, Read, Update, Delete) operations on employee records. The system should have a graphical user interface (GUI) that allows the user to interact with the system and perform the following operations:

Create new employee records: The user should be able to enter the details of a new employee (such as name, age, address, salary, etc.) and add it to the system.

Read employee records: The user should be able to view the details of all employees in the system.

Update employee records: The user should be able to select an employee from the system and update their details.

Delete employee records: The user should be able to select an employee from the system and delete their record.

Requirements:

- The system should be implemented using Java and should utilize object-oriented programming concepts such as classes, objects, and inheritance.
- The system should use a GUI toolkit such as Swing or JavaFX to provide a user-friendly interface.
- The system should use file input/output operations to store employee records persistently.
- The system should utilize collection classes and generics to manage employee records efficiently.
- The system should demonstrate exception handling and multi-threading concepts in Java.

This project can be completed individually or in a group and the trainer can use the performance, to assess the knowledge and skills of the participants, which they acquired from the training program