



Course Outline

TT1200: Core Design Patterns and Frameworks for C++ and / or Java™ Programmers

Duration: 3 days

Skill Level: Introductory

Focus: OO Java or C++ application development & design

Format: 50% hands-on coding and group dynamics labs, combined with expert lecture, open discussions and detailed patterns walk-throughs.

Language / Tools: This edition has a Java/C++ focus, although we can easily present geared for .Net

Delivery Format: Available for onsite private classroom presentation, or live online / virtual presentation

Customizable: Yes

Geared for experienced developers, *Core Design Patterns & Frameworks for Java and/or C++s* is lab-intensive hands-on training course that explores the most common object-oriented design patterns (Gang of Four) and how to use these patterns to develop solid, robust and reusable software development applications. This course can be offered for either C++ or Java programmers, with accompanying labs using the language (C++ or Java) appropriate to your background. For onsite presentations, this class can be instructed to both groups simultaneously by our experienced instructors.

► Course Objectives: What You'll Learn

Working in a hands-on environment, developers will explore key Creational, Structural and Behavior Design patterns and how they used most effectively in building robust, reusable applications. This course combines the use of hands-on coding labs with several “mini-projects” to be completed throughout the training to get the students using and reviewing the Patterns in a practical manner. All lab coding work and examples can be performed working in either a C++ or Java development environment, geared for each student’s specific background. Explicit labs have been designed for both backgrounds. Course examples in the Student Guide are written in Java.

Throughout the course we will explore the following patterns, varying the levels of coverage to drill down on the most commonly used Patterns, and to simply survey others. Students will compare and contrast the patterns and explore the advantages and disadvantages of using certain patterns for explicit development functions in both the C++ and Java environments.

► Patterns In This Course

The following Patterns are addressed in this course, covered to the specified level below. This coverage level can be adjusted based on the requirements of your organization, and which patterns you use or would like to review.

Creational Patterns	
Pattern	Coverage
Abstract Factory	Full
Factory Method	Full
Singleton	Full
Builder	Brief

Prototype	Brief
Structural Patterns	
Pattern	Coverage
Composite	Full
Adapter	Full
Proxy	Full
Bridge	Brief
Façade	Brief
Decorator	Brief
Behavior Patterns	
Pattern	Coverage
Observer	Full
Strategy	Full
Iterator	Full
Visitor	Brief
Interpreter	Brief
Chain of Command	Brief
Mediator	Brief
State	Brief

► Course Overview & Structure

This course consists of approximately 50% hands-on lab work. Throughout the course students will be led through a series of progressively advanced topics, where each topic consists of lecture and group discussion.

This class is “technology-centric”, designed to train attendees in essential design patterns, coupling the most current, effective techniques with the most effective practices.



► Audience & Pre-requisites: Who Should Attend

This is a **basic level** OO training course, designed for developers who need to identify, design, and lead the implementation of OO projects. We will explore and apply the terminology, the specification, the processes and technologies specific to OO.

Attendees should be familiar with UML and have basic programming experience in either Java or C++. This course is not recommended for developers new to Java or new to C++ programming.

► Related Courses – Suggested Learning Path

Take Before: Students should have basic development skills and experience in the following topics, or attend these courses as a pre-requisite:

- **TT1300 Object Oriented Analysis & Design using UML 2.0**

Take Instead: We offer other courses that provide different levels of knowledge or focus:

- If you require review of OOAD concepts the **TT1250 Core Design Patterns & Frameworks** course may be more appropriate.
- If you wish to emphasize design, consider **TT1310 Domain Analysis & Design using UML**
- .Net developers should consider **TT1260 Object Oriented Design Patterns & Best Practices in .Net**

Take After: We offer a variety of introductory through advanced security, development, project management, engineering, architecture and design courses. Students may want to consider the following topics as a follow-on to this course.

- Advanced core programming classes for Java or .Net
- Secure programming or design courses
- Service-Oriented Analysis and Design
- Web Services – Intro through Advanced
- Architecture & Analysis courses
- Software Engineering, Design or Project Management tracks

Please note all development courses may also be offered in other programming languages or tailored to suit your unique requirements. Please contact us for recommended next steps tailored to your longer term education, project or development objectives.

► Student Materials: What You'll Receive

Our robust course materials include much more than a simple slideshow presentation handout. Student materials include a comprehensive hard-copy course manual, complete with detailed course notes, code samples, diagrams and current reference

materials, all directly related to the course at hand, indexed for ease of use.

Step-by-step lab instructions and project descriptions are clearly illustrated and commented for maximum learning and ease of use.

Our course kits are designed to serve as an excellent and useful reference set, long after we leave your classroom.

► Experiential Learning: Hands-On Labs

This class is “technology-centric”, designed to train attendees in essential OO development skills, coupling the most current, effective techniques with the soundest industry practices.

This workshop is about **50% dynamic hands-on and group lab exercises** and **50% lecture**. Throughout the course students will be led through a series of progressively advanced topics, where each topic consists of lecture, group discussion, comprehensive hands-on lab exercises, and lab review. Multiple detailed lab exercises are laced throughout the course, designed to reinforce fundamental skills and concepts learned in the lessons. At the end of each lesson, developers will be tested with a set of review questions to ensure that he/she has fully understands that topic.

► Delivery Environment & Classroom Set Up

Although this training is skills-centric, this course can be delivered using a variety of IDE combinations, including but not limited to: Eclipse, MyEclipse, RAD, or other IDEs.

Our lab guides are complete with software-specific instructions, screen shots and detailed tutorials for using the software you select. In most cases we can easily port our classes to run in the environment of your choosing.

For course deliveries or virtual presentation using open-source tools, we'll provide our unique **LoadNGo Instant Classroom Kit**, which enables students to run the entire course off of a DVD that hosts the entire course set up software, labs, and other pertinent useful educational resources, whitepapers and more. You only need to provide the hardware and appropriate O/S, and we'll do the rest. No installation needed. **Great for secure environments.** Minimum set up burden for your team or firm, with maximum results for your students.

No matter which set up option or software your firm requires, we're pleased to provide a detailed set up guide for all private or on-site courses, and as much assistance as you require to prepare your students or classroom for the course. Our support personnel and instructors can be contacted for any advice you may require to prepare your classroom and/or students for attendance



Workshop Topics Covered

Session: Introduction to Design Patterns

- Defining Design Patterns
- Why Use Design Patterns?
- Principles Behind Patterns
- Describing Patterns

- Bridge Pattern
- Façade Pattern
- Decorator
- Survey of Structural Patterns

Session: Creational Patterns

- Abstract Factory Design Pattern
- Singleton Design Factory
- Builder Design Pattern
- Factory Method Design Pattern
- Prototype Design Pattern
- Survey of Creational Patterns

Session: Behavioral Patterns

- Observer
- Strategy
- Iterator
- Visitor
- Interpreter
- Chain of Responsibility
- Command
- Mediator
- State
- Comparison and Summary

Session: Structural Patterns

- Overview of Structural Patterns
- Composite Pattern
- Adapter
- Proxy

Session: Frameworks (Optional)

- Introduction to Frameworks
- Frameworks Illustration 1: JDBC
- Frameworks Illustration: Struts (Optional)
- Frameworks Illustration 2: Swing

***Need more info?** Please note a more detailed outline, as well as lists of lab exercises and project descriptions, is available. Please contact us at Training@triveratech.com for info.*

***Need courseware?** This course is fully customizable, and also available for license with complete support for qualified organizations. Please contact Courseware@triveratech.com for details*

► Why Work With Us?

- **We provide a solid object oriented design and development foundation.** Students will learn how to develop (and reuse!) essential OO patterns coding & design skills and concepts properly, using best design practices, grounding them for advanced curriculum. Students will be prepared for designing and implementing real solutions, right after the class ends. Students will learn the importance of developing well-designed OO applications.
- **Our courses are focused - no "fluff" included.** We offer more than a "laundry list" approach to teaching. All lessons have clear objectives, are fundamental to core OOAD development and design practices, and are reinforced by hands-on labs and solid practical examples. Each lesson has performance driven objectives that ensure students will learn technologies and skills core to fundamental OO application and patterns design – nothing more, nothing less.
- **Our materials are comprehensive, and current.** Our comprehensive manuals include not only a hard copy of the course presentation, but also detailed reference notes, pertinent diagrams and charts, current lists of suggested online resources and articles, and often technical tutorials or white papers geared to the topics at hand.
- **We set you up!** Hands-on courses also include our unique materials for each student, complete with our **LoadNGo Instant Classroom** course set up DVD, software, and a multitude of learning resources that complement the course. Run the course right off the DVD – minimal set up for your company – maximum results for your students.
- **We foster "Learning by Doing".** Progressive labs are designed in such a way that students get a firm grasp on fundamental skills while they work toward defending a complete application. All labs are take-home, and all solution code is presented in an easy to use self-study format for future use and review.
- **True content ownership gives us flexibility & quality above the rest.** These course materials are wholly-owned by our company and fully customizable - at little or no cost - to help you best meet your learning objectives. We have many dedicated experts available worldwide to instruct your team, and can provide services around the globe, either locally or virtually. We work closely with you to produce the most effective events and materials for your team, within your desired timeframe and budget.
- **We bring years of practical, current experience into the classroom and content.** Our instructors and course authors are also skilled mentors, Java, J2EE, .Net, SOA, and web services developers, architects and security-oriented professionals. We believe that learning, using and maintaining solid software execution and delivery methods are as important as gaining sharp coding skills. Best Practices for software development and execution, beyond technical coding skills, are enforced throughout all of our courses and discussions. Our team brings this



extensive experience into every classroom and engagement.

- **We're skills-centric.** Although our team has extensive experience using a variety of tools and solutions, our core content is “technology-centric”. Our aim is to teach you the best skills and solutions out there – not to sell you software from any particular vendor.
- **We're Java & J2EE authors and industry speakers.** Our team was selected to write the online *J2EE, EJB, EJB CMP-CMR and Web Services Tutorial Series for IBM developerWorks*® (www.ibm.com) These are the same instructors who train our classes and author the courseware. Most of our trainers/consultants have also authored additional articles on web services, EJB< Struts, J2EE and advanced Java topics, and are recognized speakers and presenters on the industry technical seminar circuit. Our team is comprised on several successful published authors. Members of our team have written or contributed to: *Eclipse Kick Start, Mastering Eclipse; Professional Jakarta Struts; Using Java Tools for Extreme Programming; Mastering Resin; Mastering TomCat and others.*

