JOIN LIVE PROJECTS TRAINING AND DEVELOPMENT

Are You Already a Trained Professional and Still Not Getting Jobs

Then you must have problems with your practical knowledge in web development. We have found that 80% of the candidates don't get jobs in the industry even after going through training programs in web development because they lack real life project experiences.

There are professionals, who want to shift from one field to another, say from database management to programming, because they think there are better career opportunities in that field. But, again, they do not get their first openings simply because they do have live project experience in that field.

You can join us and get trained in one of our live projects under the guidance of a highly experienced team of projects managers. At any point in time, we have multiple **PHP live projects, ASP.NET live projects, web designing live projects, and software testing live projects running in our organization** for you to choose from. Moreover, you will get opportunities to work in our own development centre of Webslate Software.

Each of our projects will give you a working experience in complete life cycle of project from designing the project specifications, to creating database, communicating with the client, programming the project, hosting, beta testing and taking care of client feedback. We have seen that once a student completes our live projects, 100% of them get multiple job opportunities. Each of our projects will give you a working experience in complete life cycle of project from designing the project specifications, to creating database, communicating with the client, programming the project, hosting, beta testing and taking care of client feedback. We have seen that once a student completes our live projects, 100% of them get multiple job opportunities.

SUMMER AND WINTER TRAINING

Do you want to make your vacations more useful and productive with summer and winter training? If you want to excel in the information technology sector, Webslate Software is the place to be during your vacations. A wide gamut of industry friendly summer and winter training awaits you here. As you step in with a dream to create your own identity, the Webslate Software Summer and Winter Training courses lets you realize your goals.

If you want to carve a niche as a premier information technology professional, Webslate Software Summer and Winter Training are the ideal courses to join. You may just be pursuing masters in computer applications course and lack adequate time for enrolling in other full time curriculum. These vacations are the ideal time to join these training courses and get and the project

These courses lasts for a short time frame and you can enroll for various options like PHP Summer Training, Web Designing winter training, SEO summer training, Dot Net winter Training etc. You get the chance to interact with industry experts who provides valuable suggestions and tips. It is the best way to get enriched with high end summer and winter training programs where live project knowledge is also provided. Stay ahead of all in the vacations by enrolling with Webslate Software and reach the pinnacle of IT excellence.

Training Features

High-standards of training from prominent industry experts :-

➢ Hands-on soft skills development.
➢ One-on-one interaction with each student; personal workstations.
➢ Anytime broadband internet access Convenient 'on floor' practice facility.
➢ Practical sessions at work stations and labs.
➢ Regularly updated course material to offer the lates.
➢ A resourceful library (reference books, e-books) at disposal.
➢ Involvement in 'real life international projects'.
➢ Weekend classes for working participants.
➢ Advanced web servers to host your work.

TRAINING PROCESS

Quality training demands quality infrastructure. Webslate Software not only provide you with the cream of the industry experts as the training faculty, but also the state-of-the-art infrastructure for a complete learning experience.

➢ State-of-the-art servers
➢ Unique wireless network
➢ High capacity Broadband Internet service
➢ The latest range of work terminals
➢ Interactive audio-video multimedia sessions
➢ Advanced web servers to host your work
➢ A library with reference books for web programming and the latest trends
➢ A valuable collection of e-books/ books
➢ Exemplary reports & case studies from our own success stories
➢ Centrally air-conditioned environment

We pay individual attention to each student: Our batch is limited to 20 students only. This way we ensure each one gets the best of us with carefully designed one-on-one coaching structure.

We deliver the best knowledge in the simplest form: We at Webslate Software believe in nothing but the best. Hence we design our course in accordance with the time-tested modules practiced by the top-level universities and companies. The resultant easy to learn and easier to apply approach adds to a comprehensive learning experience.

Our knowledge streaming process:

Our goal is to shape IT professionals who have the potential to deliver international standards. We encourage our students to streamline their learning curve through an industry-oriented outlook with equal stress on hands-on field application along with theories.

➢ Interactive classes with industry experts.
➢ Practical sessions at work stations and labs.
➢ A scope to self-assessment, to bring out hidden talents and develop intuition.
➢ Unique opportunity to get involved in 'real life international projects'.
➢ Weekend classes designed ideally for working professionals.
➢ Option for post-course extended learning to ensure continuity.

CORE JAVA TRAINING

Introduction to Java tutorials for new java programmers.

Java is a powerful object-oriented programming language with simple code structure. You can create applications and applets with graphics and user interfaces because Java has built-in application programming interface (API) and due to its rich set of interfaces java has a platform in itself. Java provides errorless programming with platform independent feature. Java developers tried to write the code structure similar to the syntax of C++ so it would be easy for C++ programmers to learn java. In comparison to C++, Java handles many operation like creation and deletion of memory automatically, it helps to make bug-free code in Java application.

The course we offered is designed and centered for new programmers. We provides you the course materials with extreme knowledge-full examples prepared by Java professionals. It makes you perfect in Java programming as well as for being a successful software professional. Here we also provides topic based examples with description in more generic way.

We prepare the course which covered all the topic of Java with point of view of Java professionals. We also keep in mind the topic which covered by sun-certification and try to make you perfect for certification. We covered all necessary topic of object-oriented programming in Java and this is the strong reason for you to trust our courses. The course we provides you is often easy to understand the concept of object-oriented programming because all the chapters are further divided in more topic.

Java Training Objectives:

After successfully completion of object-oriented programming course in Java. You are aware of the :

➢ Uses of all data type and file input and output features.
➢ Uses of Objects, classes, interfaces and inheritance.
➢ Error handling in your program with exception handlers.
➢ Object oriented design and analysis.
➢ Serialization and Parsing the objects using object serialization.
➢ Write multi-threaded programs and synchronize threads in program.
➢ Familiar with RMI, UML diagram, relationship with classes, MIDIets, Applet and Swing Components.

1. Getting Started with Java SE

   ➢ What is Java ?
   ➢ Installing Java.
   ➢ The JDK Directory Structure.
   ➢ SDK structures
   ➢ Object Orientation.
   ➢ Java Language.
   ➢ Java Virtual Machine.
   ➢ World Wide Web and Java.
   ➢ Java Platforms.
   ➢ Java Tools.
   ➢ Java as a Programming Language.
   ➢ Java as an Object Oriented Language.

2. First Java Programs

   ➢ Writing your first Java program "Hello, World", your first small step towards learning Java.
   ➢ Program Structure.
   ➢ Output in Java.
   ➢ Variables and Expressions.

3. Datatypes and Variables

   ➢ Primitive Datatypes.
   ➢ Variable Names.
   ➢ Numeric Literals.
   ➢ Character Literals.
   ➢ String.
   ➢ Java String Tutorial and examples.
   ➢ String Literals.
   ➢ Arrays.
   ➢ Java Array Tutorial and Examples.

4. Introduction to Objects

   ➢ Object Models.
   ➢ Classes and Objects.
   ➢ Abstract methods and Classes.
   ➢ Input in Java.
   ➢ Input Wrapper Class.
   ➢ Packages.

5. Data Types and Operators

   ➢ Strong Typing.
   ➢ Integer Data Types.
   ➢ Floating Point.
   ➢ Conversions Between Types.
   ➢ Arithmetic Operators.
   ➢ Doing Math in Java.
   ➢ Precedence.
   ➢ Errors in Integer Arithmetic.

6. Operators and Expressions

   ➢ Expressions.
   ➢ Assignment Operator.
   ➢ Arithmetic Operators.
   ➢ Relational Operators.
   ➢ Logical (Conditional) Operators.
   ➢ Increment and Decrement Operators.

7. Bitwise and Bit Shift Operators

   ➢ Java Bitwise AND " &" Operator.
   ➢ Java bitwise NOT "~" operator.
   ➢ Java bitwise XOR "^" operator.
   ➢ Java bitwise OR " |" operator.
   ➢ Java Left Shift "<<" Operator.
   ➢ Java Right Shift ">>" Operator.
   ➢ Java unsigned right shift ">>>"operator .
   ➢ Java Truth-Table.
   ➢ Truth Tables.
   ➢ Shifting and Masking.

8. Control Flow

   ➢ Java Control Statement.
   ➢ Conditional (if) Statements.
   ➢ Adding an else if.
   ➢ Conditional (switch) Statements.
   ➢ while and do-while Loops.
   ➢ for Loops.
   ➢ A for Loop Diagram.
   ➢ Enhanced for Loop.
   ➢ The continue Statement.
   ➢ The break Statement.

9. Booleans and Enumerations

   ➢ Boolean Variables.
   ➢ Logical and Relational Operators.
   ➢ If Tests.
   ➢ Compound Statements.
   ➢ Switch Statement.

10. Loops and Program Flow

   ➢ While Loops.
   ➢ For Loops.
   ➢ Loops and If Tests Together.
   ➢ Nested If Statements.

11. Object-Oriented Programming

   ➢ Introduction to Object-Oriented Programming.
   ➢ Classes and Objects.
   ➢ Fields and Methods.
   ➢ Encapsulation.
   ➢ Access Control .
   ➢ Inheritance.
   ➢ Polymorphism.
   ➢ Interface.
   ➢ Best Practices.

12. Objects and Classes

   ➢ Structured Data.
   ➢ Classes.
   ➢ References.
   ➢ Defining a Class.
   ➢ Instantiating and Creating an Object .
   ➢ Assignment of Objects.
   ➢ Instance Data and Class Data.
   ➢ Abstract Classes.
   ➢ Access Modifiers.
   ➢ Initialization.
   ➢ Garbage Collection.
   ➢ Methods.
   ➢ Public vs. Private.
   ➢ Encapsulation.

➢ Constructors.
➢ Formatted Output.

13. Using Java Objects

➢ Printing to the Console.
➢ printf Format Strings.
➢ String Builder and String Buffer.
➢ Methods and Messages.
➢ to String.
➢ Parameter Passing.
➢ Comparing and Identifying Objects.
➢ Destroying Objects.
➢ Using the Primitive-Type Wrapper Classes.

14. Methods

➢ Methods.
➢ Calling Methods.
➢ Defining Methods.
➢ Method Parameters.
➢ Method Overriding.

15. Characters and Strings

➢ Char Data Type.
➢ Character Codes.
➢ ASCII and Unicode.
➢ String Class.
➢ String Input and Output.
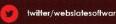➢ String Methods.

16. Modular Programming

- ➢ Monolithic Programs.
- ➢ Static Variables and Methods.
- ➢ Functional Modularity.
- ➢ Object Modularity.
- ➢ Top-Down and Bottom-Up Development.
- ➢ Pass-By-Value and Pass-By-Reference.
- ➢ Nested Classes.

17. Arrays

- ➢ One Dimensional Arrays
- ➢ Subscripts.
- ➢ Initializing Arrays.
- ➢ Arrays and the For-Each Loop.
- ➢ Copy and Assignment.
- ➢ Arrays of Objects.
- ➢ Multidimensional Arrays.
- ➢ Searching.

18. Exception Handling and More Flow Control

- ➢ Exceptions Overview.
- ➢ Exceptions.
- ➢ Declaring Exceptions.
- ➢ Defining and Throwing Exceptions.
- ➢ Errors and Runtime Exceptions.
- ➢ Catching Exceptions.
- ➢ The finally Block.
- ➢ Exception Methods.
- ➢ Assertions.
- ➢ Errors in Integer Arithmetic.
- ➢ Floating Point Operations.
- ➢ I/O Exceptions vs. Runtime Exceptions.

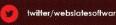- ➢ Break: o Continue.
- ➢ Do.

19. Input/Output Streams

- ➢ Overview of Streams.
- ➢ Bytes vs. Characters.
- ➢ Converting Byte Streams to Character Streams.
- ➢ File Object.
- ➢ Binary Input and Output.
- ➢ Print Writer Class.
- ➢ Reading and Writing Objects.
- ➢ Basic and Filtered Streams.

20. Core Collection Classes

- ➢ The Collections Framework.
- ➢ The Set Interface.
- ➢ Set Implementation Classes.
- ➢ The List Interface.
- ➢ List Implementation Classes .
- ➢ The Queue Interface.
- ➢ Queue Implementation Classes.
- ➢ Implementing a Stack.
- ➢ The Map Interface.
- ➢ Map Implementation Classes.

21. Collection Sorting and Tuning

- ➢ New Features in JSE 6.
- ➢ Changing in I/O(JSE 6).
- ➢ Using Java 6.0 Features with Collections.
- ➢ Sorting with Comparable.
- ➢ Sorting with Comparator .

- ➢ Sorting Lists and Arrays.
- ➢ Collections Utility Methods.
- ➢ Tuning Array List
- ➢ Navigable Map and Navigable Set.
- ➢ Tuning Hash Map and Hash Set.

22. Inner Classes

- ➢ Inner Classes.
- ➢ Member Classes.
- ➢ Local Classes.
- ➢ Anonymous Classes.
- ➢ Instance Initializers.
- ➢ Static Nested Classes.

23. Introduction to Threads

- ➢ Overview of thread.
- ➢ Life Cycle of thread.
- ➢ Creating Threads.
- ➢ Multithreading.
- ➢ Deadlock.
- ➢ Inter-Thread Communication.
- ➢ Thread States.
- ➢ Runnable Threads.
- ➢ Coordinating Threads.
- ➢ Interrupting Threads.
- ➢ Runnable Interface.
- ➢ ThreadGroups.

24. Packages

- Packages
  - Package Categories.
  - Create your own Package.
  - How to import a package.
  - Create Sub packages.
- The import Statement.
- Static Imports.
- CLASSPATH and Import.
- Defining Packages.
- Package Scope.

25. Advanced Java Features

- Reusable Software Components.
- Abstraction.
- Inheritance.
- Inheritance Hierarchies.
- Polymorphism.
- Abstract Classes.
- Interfaces.
- Collections.
- Iterators.
- Auto-Boxing.

26. Introduction to JDBC

- Features of JDBC 3.0.
- Features of JDBC 4.0.
- New Features in JDBC 4.0.
- Difference between JDBC 3.0 and JDBC 4.0.
- The JDBC Connectivity Model.
- Database Programming.
- Connecting to the Database.

➢ Creating a SQL Query.
➢ Getting the Results.
➢ Updating Database Data.

27. JDBC SQL Programming

➢ Error Checking and the SQL Exception Class.
➢ The SQL Warning Class.
➢ JDBC Driver Types.
➢ Result Set Meta Data.
➢ Using a Prepared Statement.
➢ Parameterized Statements.
➢ Stored Procedures.
➢ Transaction Management.

Advanced Java™ Training Course

Course Summary

Advanced Java™ Training course provides you with advanced skills for programming in Java language.

This course delves deeper into the data structures, file input and output features, exception handing, object oriented design and analysis, object serialization, database connectivity with JDBC, multithreaded programming, and networking capabilities of Java.

Upon completion, you will be able to understand, design, and develop complex Java-based applications.

Objectives

Upon successful completion of Advanced Java™ Training course, you will be able to:

➢ Use reflection to examine objects at runtime and properly compare objects for equality.
➢ Handle errors in your program by writing exception handlers.
➢ Create and manipulate arrays and collections.
➢ Read and write files using the java.io package.
➢ Serialize objects using object serialization.
➢ Write multi-threaded programs and synchronize threads.
➢ Access data from relational databases using JDBC. Perform basic network communication using sockets.

Java Training Outline

Generics

➢ Using Generics.
➢ Type Erasure.
➢ Type Boundaries.
➢ Wildcards.
➢ Generic Methods.
➢ Strengths and Weaknesses of Generics.
➢ Legacy Code and Generics.

0522-4041496,+91-9455050063
webslatesoftware@gmail.com
www.webslatesoftware.com
L-2/630 Vineet Khand, Gomti Nagar, Lucknow-INDIA

Threads

- Java Thread Model.
- Creating and Running Threads.
- Manipulating Thread State.
- Thread Synchronization.
- Volatile Fields vs. Synchronized Methods.
- wait and notify.
- join and sleep.
- The Concurrency API.
- Atomic Operations.

Reflection

- Uses for Meta-Data.
- The Reflection API.
- The Class<T> Class.
- The java.lang.reflect Package.
- Reading Type Information.
- Navigating Inheritance Trees.
- Dynamic Instantiation.
- Dynamic Invocation.
- Reflecting on Generics.

Annotations

- Aspect-Oriented Programming and Java.
- The Annotations Model.
- Annotation Types and Annotations.
- Built-In Annotations.
- Annotations vs. Descriptors (XML).

Sockets

- ➤ The OSI Reference Model.
- ➤ Network Protocols.
- ➤ The Socket Class.
- ➤ The Server Socket Class.
- ➤ Connecting Through URL Objects.
- ➤ HTTP and Other TCP Servers.
- ➤ Datagram Clients and Servers.
- ➤ Non-Blocking Sockets.

Overview

- ➤ Three Tiers for J2EE.
- ➤ Three Tiers for J2SE.
- ➤ The Case Study.
- ➤ Design Patterns.
- ➤ Domain and Service Models.

The Presentation Tier

- ➤ The Standalone/Client Application.
- ➤ JDesktopPane and JinternalFrame.
- ➤ Adapting JList, JTable, and JTree to Services.
- ➤ Presentation-Tier Patterns.

The Business Tier

- ➤ Distributing the Application.
- ➤ A Chain of Services.
- ➤ Logging.
- ➤ Business-Tier Patterns.
- ➤ Designing for Latency.

The Persistence Tier

- ➢ A Database is Not a Persistence Tier!
- ➢ Persistence Frameworks.
- ➢ Persistent-Object Strategies.
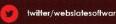- ➢ Persistence-Tier Patterns.
- ➢ Caching.

Introduction to JFC

- ➢ Abstract Windowing Toolkit Basics.
- ➢ Simple Layout Management.
- ➢ Simple Event Handling.
- ➢ Lightweight Controls.
- ➢ JFC Feature Set.
- ➢ JFC Architecture and Relationship to AWT.

JFC Application Design

- ➢ Role of a JFrame.
- ➢ Building a Frame-Based JFC Application.
- ➢ Panes.
- ➢ Using Dialogs.

JFC Components

- ➢ JFC Component Class Hierarchy.
- ➢ JComponent Features.
- ➢ Simple Control Types.
- ➢ Text Components.
- ➢ Menus.
- ➢ Managing Look and Feel.

RMI Architecture

- ➢ Motivation for RMI.
- ➢ RMI, EJB, and CORBA.
- ➢ RMI Architecture.
- ➢ Lifetime of a Remote Method Invocation
- ➢ Registries.
- ➢ Naming and URL Resolution.
- ➢ Interface Design.
- ➢ The Remote Interface.
- ➢ Implementation Classes.
- ➢ The Remote Object and Remote Server Classes.
- ➢ The Unicast Remote Object Class.
- ➢ Server Implementation.
- ➢ Using the Registry.
- ➢ Client Implementation.
- ➢ Code Deployment.

Practical RMI

- ➢ RMI Marshaling.
- ➢ Passing Objects.
- ➢ The Factory Pattern.
- ➢ Serialization vs. Remote Reference.
- ➢ Designing for Latency.
- ➢ The Transfer Object Pattern.
- ➢ Controlling Object Location.
- ➢ Exception Handling.

Database and SQL Fundamentals

- ➢ Relational Databases and SQL.
- ➢ Database, Schema, Tables, Columns and Rows.
- ➢ SQL Versions and Vendor Implementations.
- ➢ DDL -- Creating and Managing Database Objects.
- ➢ DML -- Retrieving and Managing Data.

- ➢ Sequences.
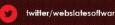- ➢ Stored Procedures.
- ➢ Using SQL Terminals.

JDBC Fundamentals

- ➢ What is the JDBC API?
- ➢ JDBC Drivers.
- ➢ Making a Connection.
- ➢ Creating and Executing a Statement.
- ➢ Retrieving Values from a Result Set.
- ➢ SQL and Java Data types.
- ➢ Creating and Updating Tables.
- ➢ Handling SQL Exceptions and Proper Cleanup.
- ➢ Handling SQL Warning.

Advanced JDBC

- ➢ SQL Escape Syntax.
- ➢ Using Prepared Statements.
- ➢ Using Callable Statements.
- ➢ Scrollable Result Sets.
- ➢ Updatable Result Sets.
- ➢ Transactions.
- ➢ Commits, Rollbacks, and Save points.
- ➢ Batch Processing.

Introduction to Row Sets

- ➢ Row Sets in GUI and J2EE programming.
- ➢ Advantages of Row Sets.
- ➢ Row Set Specializations.
- ➢ Using Cached Row Sets.

Latest Java Frame Work On Offer

- ➢ **Spring.**
- ➢ **Hibernate.**
- ➢ **Apache wickets.**
- ➢ **Android.**

Student Testimonials

The instructor was very knowledgeable and was able to answer all questions that came up. He clearly has extensive knowledge in this domain area.