Core Java Syllabus

Overview:

Java programming language is developed by Sun Microsystems. Java is object oriented, platform independent, simple, secure, architectural–neutral, portable, robust, multi-threaded, high performance, distributed and dynamic. It can be used to develop software and also applets. A java program can run on various operating systems without rewriting the code. And this is possible because of java run-time environment which tells the operating system what to do by interpreting the java code.

Objective:

- To become familiar with the features of Java Language
- To discover how to write Java code according to Object-Oriented Programming principles.
- To become comfortable with concepts such as Classes, Objects, Inheritance, Polymorphism and Interfaces
- To learn Java APIs for Collections, I/O Streams
- To design GUI applications and Applets using AWT and Swing.
- To develop Multithreaded and Networking applications.

Pre-requisite / Target Audience:

- C language skills (Good to Have)
- This course is designed to meet the needs of those who want to be professional Java developers.
- This will also help the audience to get through the Java Programmer Certification.

Module 1: Java Language Environment

In this Module you will learn what is a java, and its features, and why it is popular? Means by comparing the below of its features with other programming language's you will understand.

- Object Oriented
- Platform Independent
- Automatic Memory Management
- Compiled / Interpreted approach
- Robust
- Secure
- Dynamic Linking
- Multi-Threaded
- Built-in Networking

Model 2: Java Fundamentals

In this module you will learn the basic structure of the programming and how to create your own structural code, and where to use it.

- Data types
- Operators
- Control Statements
- Arrays
- Enhanced for-loop
- Enumerated types,
- Static import
- Auto boxing
- C-style formatted I/O
- Variable arguments

Module 3: Essentials of Object-Oriented Programming

In this module you will learn the basic definitions and uses and how to make our code in more structure way, so that anyone can understand our code, how to make it easier.

- Object and Class Definition
- Using encapsulation to combine methods and data in a single class
- Inheritance and Polymorphism

Module 4: Writing Java Classes

In this module you will learn all the concepts Oops where we will use all these concepts in our daily way life by knowingly or unknowingly. By learning this module you can able to create a code in a standard format.

- Encapsulation
- Polymorphism
- Inheritance
- OOP in Java
- Class Fundamentals
- Using Objects
- Constructor
- Garbage Collection
- Method Overloading
- Method Overriding
- Static Members
- Understanding Interface

- Using Interfaces

Module 5: Packages

In this module you will learn how to re-use/access our class files when it is in same package/different package/different project.

- Why packages
- Understanding Class path
- Access modifiers & their Scope

Module 6: Exception Handling

In this module you will learn how to handle our standalone applications/web applications, whenever there is an error occurs, how to tackle it, and where it is occurring, by learning this module you will get it.

- When an exception occurs.
- Importance of Exception Handling
- Exception Propagation
- Exception Types
- Using try and catch
- throw, throws, finally
- Writing User defined Exceptions

Module 7: I/O Operations in Java

In this module you will learn how to create a file and how to modify/read/write/handle an existing file, through your code and you can make your file access permission rights.

- Byte Oriented Streams
- File Handling
- Readers and Writers

Module 8: Multithreaded Programming

In this module you will learn how to perform multiple tasks at a same time or it may be partially. Here tasks can be either running multiple code simultaneously when some background code is running or to run the code one after another or it may be at a time.

- Introduction to Multi-Threading
- Understanding Threads & its States
- Java Threading Model
- Thread class & Runnable Interface
- Thread Priorities

- Thread Synchronization
- Interthread Communication
- Preventing Deadlocks

Module 9: Java Util Package / Collections Framework

In this module you will learn how to make/get our content in a user's prospective/his requirement, when it is in same file or it may be in a different file even if it is in different format.

- Collection & Iterator Interface
- Enumeration
- List and ArrayList
- Vector
- Comparator
- Set Interface & SortedSet
- Hashtable
- Properties

Module 10: Generics

In this module you will learn how to create our own class type parameters where we can reuse the same code by giving different inputs.

- Introduction to Generics
- Using Built-in Generics Collections
- Writing Simple Generic Class
- Bounded Generics
- Wild Card Generics

Real-time Project involving most of the above concepts with following will be provided

- Product Abstract Document
- Requirement Specification Document
- **Step-by-Step procedure for building the project from ground up by using IDE.**
- Complete Source Code

At the end of the course participants will get the knowledge of:

1. Creating your own windows application and you can able to know how to utilize the various available resources without need of other's help.

Advance Java Syllabus

Overview:

Java is platform independent language and Object oriented Programming language. Using advanced Java programming language we can learn how to design dynamic web applications using Java Server Pages and Java Servlet and how to connect to data base drivers. Advanced java course consist JDBC, HTML, Servlet, JSP and JSTL. Using JDBC concept you can learn database concepts in depth and perform all CRUD operations easily. Using HTML you can develop static web pages. Using Servlet and JSP you can develop dynamic web pages.

Course Objectives:

- To become familiar with the advanced features of Java Language.
- To develop Web Applications using Servlets / JSP and deploy in popular servers like Tomcat.
- To understand Java Servlets and their life cycle
- To understand Java Web application directory structure
- To develop Server side components in a Java Web application
- To understand Java server Pages (JSP) technology
- To develop reusable components using JavaBeans.
- To develop JSP pages using which use JavaBeans
- To develop JSP Custom tags and use them in JSP pages
- Writing Event Listeners in Java Web application
- Handling File uploads in Java Web application
- To discover how to write Java applications this can communicate with Relational Databases.

Pre-requisite / Target Audience:

This course is designed to meet the needs of those who want to be professional Java developers. This will also help the audience to get though the Java Certification called Web Component Developer.
Students should be familiar with Java programming techniques and should be comfortable with concepts such as Classes, Objects, Inheritance, Interfaces I/O Streams, Threading and Networking. Certificate like Sun Certified Java Programmer will be an added advantage.

Module 1: Java Database Connectivity (JDBC)

We know the data stored in a file can't be a permanent. So we use to store data from now onwards in database, in this module we will learn how to connect/store data of standalone/web application in database or to retrieve the data from it.

- Overview of RDBMS
- Introduction to Call Level Interface(CLI)
- Introduction to JDBC
- JDBC Architecture
- Types of JDBC Drivers
- Establishing a JDBC Connection
- Using Statement
- Using Prepared Statement
- Using Callable Statement
- Scrollable and Updatable Result Set
- Inserting & Fetching from BLOB Columns
- Managing Transactions in JDBC
- New Features introduced in JDBC 3.0
- Auto Increment Columns

Module 2: Extensible Markup Language (XML)

In normally web applications will have so many pages server don't know where to start, that's why we use xml we define our start application in xml and all other related simultaneous pages in it, so that server will recognize how many pages are there and where to start, we can store some data in xml also by parsing the xml we can able to retrieve the data.

- eXtensible Markup Language (XML)
- Introduction to XML
- Document Object Model (DOM) using JAXP
- Understanding DOM
- Using DOM in Java
- StAX in Java
- Understanding StAX
- Programming with StAX

Module 3:- Common Gateway Interface

This is the first gateway interface where user programs can run on web server which will invoke the client browser to make the user interface between client and web server.

- Introduction to CGI
- Understanding Environment Variables
- Disadvantages and Limitations of CGI

Module 4: Java Servlets

In this module we will learn how the client request is generated and how to track the user and how to make client application more secure. Don't worry if you don't get it by learning this concept you will get it.

Servlet as J2EE Web Component

- Servlet as an improved CGI
- Servlet Fundamentals / API
- What is a Web-Container
- Servlet Life Cycle / Architecture
- HTTP GET and POST Request Methods
- Processing Html Forms
- What is Name-Value pair
- Content Types and MIME
- Configuration of Web Application
- Understanding the Deployment Descriptor (DD) / web.xml
- Specifying the Welcome file list
- Servlet URL Pattern Mapping
- Init Parameters
- State Management
- Using HTTP Session
- Using Cookies
- Using Application

Module 5: Java Server Pages (JSP)

In this module we can able to differentiate the model, view, server logic so that lot's of client side code processing is reduced on server which is not done in servlets concept.

- JSP Architecture
- JSP Standard / Implicit Objects
- Request
- Response
- Out
- config
- Application
- Session
- Page
- Page Context
- exception

- JSP Page Implementation Class
- JSP Basics & Syntax
- JSP Directive Tags
- Page Directive
- Include Directive
- Taglib Directive
- JSP Action Tags
- Forward Action Tag
- Include Action Tag
- JSP Script related Tags
- Scriptlet Tag
- Expression Tag
- Declaration Tag
- Using Java Beans from JSP
- UseBean Tag
- setProperty Tag
- getProperty Tag
- JSP Custom Tag Library
- JSP 2.0 Tag Files
- JSP 2.0 Simple Tag
- Empty Tag
- Tag with Body Content
- (JSP Fragment)

Module 6: JavaBeans

Each time whenever either storing or retrieving the data from a database we need to a variable, instead of it, in here we will create a property class which will reuse so many times as we want.

- JavaBean Architecture
- JavaBean Characteristics
- Providing Properties & Methods

Module 7: JSP Expression Language (EL)

In this module we will learn to avoid the expression tag of java code from the client side.

- Syntax
- Using different scope objects
- Calling Functions from EL

Module 8: JSP Standard Tag Library (JSTL)

In this module we will completely avoiding the java code from the client side, so that the processing time will be reduced.

- General Purpose Actions
- Conditional Actions
- Iterator Actions

Module 9: Filters in Web Application

Filters are used to authenticate the user and navigate the user to his particular page and not only that it is also helpful to filtering the user given operations.

- Filter Basics
- Filter Lifecycle
- Filter Chaining
- Filter Example

Real-time Project involving most of the above concepts with following will be provided

- Product Abstract Document
- Requirement Specification Document
- **Step-by-Step procedure for building the project from ground up by using IDE.**
- Complete Source Code

At the end of the course participants will get the knowledge of:

Creating your own web application and you can able to know how to utilize the various available resources without need of other's help.

Spring & Hibernate

Overview:

The spring framework is an application framework that provides a lightweight container that supports the creation of simple-to-complex components in a non-invasive fashion. Spring's flexibility and transparency is congruent and supportive of incremental development and testing. The framework's structure supports the layering of functionality such as persistence, transactions, view-oriented frameworks, and enterprise systems and capabilities.

Objective:

- Explain how the issues associated with object persistence in a relational model are addressed by Hibernate
- Understand the relationships between SQL, Java, Spring, and Hibernate
- Discuss the challenges to adopting Hibernate in the enterprise
- Write applications that take advantage of the Hibernate Persistence Manager.
- Map Java classes to relational tables.
- Capture both relational and inheritance associations in metadata using either XML or the Java 5 Annotations mechanism.
- Create and use mappings between Java classes and relational databases.
- Understand how identity and keys are handled in Hibernate.
- Understand the persistent object lifecycle and how that relates to transactions and concurrency.
- Take advantage of Hibernate's data filtering and interception.
- Explain the issues associated with complex frameworks such as Java EE and how Spring addresses those issues
- Write applications that take advantage of the spring container and the declarative nature of assembling simple components into applications.
- Work with Spring's support for transactions
- Understand how to use Hibernate within the Spring framework

Pre-requisite / Target Audience:

Prerequisites for learning Spring Framework include basic knowledge of Java and databases.

- Knowledge of core java.
- Knowledge of database.
- And basic Knowledge of web application development.

Module 1: Spring Basics

This module is the core of the Spring Framework. It provides implementation for features like ,IoC (Inversion of Control) and Dependency Injection with singleton design pattern.

- What is Spring Framework
- Inversion of Control
- Dependency Injection
- Bean Factory
- Developing First Spring Application

Module 2: Built-in Bean Factories

This module provides implementation for the factory design pattern through BeanFactory, and we will learn implementation of ApplicationConext.

- Application Context
- Wiring Beans
- Bean Lifecycle in Container
- Spring Events

Module 3: Spring AOP

In this module we will learn by separating application business logic from system services, Spring Framework supports Aspect Oriented Programming and enables cohesive development.

- Introduction to AOP
- Role of AOP in Spring
- AOP Advice
- AOP Pointcuts
- Spring AOP Introductions
- ProxyFactoryBean

Module 4: Spring Data Access

This module provides JDBC abstraction layer which eliminates the need of repetitive and unnecessary exception handling code.

- JDBC Abstraction Layer
- Data Access Exceptions
- DAO Support

Module 5: Spring O-R Mapping

ORM stands for Object Relational Mapping. This module provides consistency/ portability to our code regardless of data access technologies based on object oriented mapping concept.

- What is O-R Mapping
- O-R Mapping support in Spring
- Hibernate Support / Mapping

Module 6: Spring Transaction Management

This module supports programmatic and declarative transaction management for classes that implement special interfaces and for all your POJOs. All the enterprise level transaction implementation concepts can be implemented in spring by using this module.

- Transaction Abstraction in Spring

- Transaction Strategies
- Programmatic Transaction
- Declarative Transaction

Module 7: Spring Remoting and Enterprise Services

In this module we will learn how spring Remoting will be implemented by following RMI and various remoting technique.

- Introduction to Spring Remoting
- Java RMI in Spring
- Accessing JNDI
- Invoking EJB from Spring
- Web Service in Spring using JAX-RPC Support
- Messaging Support in Spring using JMS
- Sending Mail with Spring Mail
- Scheduling using Timer Support

Module 8: Spring Web MVC Framework

This module contains Model-View-Controller (MVC) based implementation for web applications. It provides all other features of MVC, including UI tags and data validations. Web MVC Architecture

- Role of DispatcherServlet
- Controller
- Handler
- View Resolving
- Data Binding
- File Upload Support

Module 9: Securing Spring Applications

In this module we will learn how spring security will be implemented by HTTP basic authentication and following concept.

- Acegi Security System for Spring
- Authentication
- Access Control
- Web Application Security
- Method Invocation Security

Module 10: Spring Boot

In this module we will learn how Spring Boot makes it easy to create stand-alone, and pre compiled spring based Applications that we can "just run".

- Introduction Spring boot
- Installation of STS in eclipse
- Using Spring STS IDE
- Using Spring Initializer Website
- Hello World example using spring boot
- Java-Based Applications

Relational Persistence Using Hibernate

Module 1: Introduction to Hibernate

In this module we will learn how Hibernate framework simplifies the development of java application to interact with the database. Hibernate is an open source, lightweight, ORM (Object Relational Mapping) tool.

- Drawbacks of direct JDBC
- Plain Old Java Object (POJO)
- What is O-R Mapping
- Simple Database Application

Module 2: Hibernate Configuration

In this module we will learn configuration of hibernate and more about how to add various jar file by creating user defined lib.

Module 3: Hibernate Concepts

In this module we will learn how primary key will be configured and how we can make auto increment and crud example by using hibernate.

- Id and Primary Key
- Id Generation Methods
- SessionFactory
- Session
- Transaction
- Developing CRUD Application

Module 4: Hibernate O-R Mapping

In this module we will learn how different types of operation with respect to database and how the relationship of database can be maintained by following OR-Mapping.

- Mapping Declarations
- Modeling Composition with Relationship
- Modeling Composition with Components
- One-to-One Association
- One-to-Many Association
- Many-to-Many Association

Module 5: Manipulating and Querying

In this module we will come to know how hibernate hibernate detects any changes made to that object and synchronizes it with database when we close or flush the session.

- Persistent Objects
- Object Loading
- Executing Queries
- Iterating Results
- Scalar Results
- Bind Parameters
- Pagination

Module 6: Hibernate Query Language

This module will learn about Hibernate Query Language (HQL) is an object-oriented query language, similar to SQL, but instead of operating on tables and columns, HQL works with persistent objects and their properties

- Select clause
- From clause
- Where clause
- Aggregate functions
- Expressions
- Sorting
- Grouping
- Sub queries

Module 7: Transactions and Concurrency

In this module we will come to know how a transaction is a unit of work in which either all operations must execute or none of them. To understand the importance of transaction,

- Session and Transaction Scopes
- Database Transaction Demarcation

- Optimistic Concurrency Control
- Pessimistic Concurrency Control

At the end of the course participants will be able

- Work in any spring and hibernate based project.
- Understand complete ORM feature by using hibernate.
- Messaging service
- Simple storage service.

Real-time Project involving most of the above concepts with following will be provided

- Product Abstract Document
- Requirement Specification Document
- **Step-by-Step procedure for building the project from ground up**
- Complete Source Code
- Database Script with Sample data
- Instructions to Setup the Project on a Development box
- Instruction to Deploy the project on Production Box / Tomcat server

At the end of the course participants will be able to

1. Work in spring and hibernate application.
2. Will be able to perform all operation including transaction management etc.

AWS Syllabus

Overview:

Amazon Web Services (AWS) is a cloud service from Amazon, which provides services in the form of building blocks, these building blocks can be used to create and deploy any type of application in the cloud.
Amazon Web Services (AWS) provides on-demand computing resources and services in the cloud, with pay-as-you-go pricing. For example, you can run a server on AWS that you can log on to, configure, secure, and run just as you would a server that's sitting in front of you and it provides many of the same benefits: capacity exactly matches your need, you pay only for what you use.

Introduction and getting started with AWS

- Hosting a website on Amazon S3.

- Creating a MySQL DB Instance.
- Work with static deployment.