NEXSOFT
IT SOLUTIONS
Our Vision ... Your Success

Machilipatnam

offers

FULL STACK JAVA

Front End	–	HTML, CSS, JAVASCRIPT
Back End	–	Core Java + Advanced JAVA(Servlets, JSP, JDBC) + Hibernate + SPRING + SPRING MVC + SPRING BOOT
DATABASE	–	MySQL
DevOps	–	AWS + GIT

Why you must learn FULL STACK JAVA?

- Earn a higher salary with Java
- Choose FULL STACK JAVA for a tension-free career prospect as it has HUGE job market.

By the End of FULL STACK JAVA course from NEXSOFT, you will be able to

- Create and Implement software applications.
- Develop stand-alone applications and server-based web applications.
- Develop Database based applications using JDBC and Hibernate framework.
- Handle application programming like multi-threading and string-handling.
- Develop web applications using SPRING framework and develop REST APIs using SPRING BOOT.
- Use GIT for version controlling.
- Use AWS for cloud deployment.

#First Floor, 17/534, above Bank of Baroda, Ramanaidupeta, Machilipatnam, Andhra Pradesh.
WhatsApp or call to +91 789 789 8511

Syllabus

JAVA Introduction

- Introduction to JAVA – WHAT, WHY, HOW, WHERE?
- The language of 50 keywords
- Our first Java program
- Variable and datatype
- Understanding variables
- Primitive Data Types (Integer datatype)
- Primitive Data Types (floating point datatype)
- Primitive Data Types (boolean and char)
- BigDecimal class introduction
- String basics
- Type casting

JAVA Setup

- Download process and installation of Java (Windows)
- Download and installation process of Eclipse

JAVA Decision Making Under Java

- Arithmetic operators
- Making decision with operators
- Ternary operator
- Assignment operators
- Making decision with if else
- Discussion forum
- Exercise

JAVA Flow control

- Switch case - Examples
- For loop
- While loop
- Do while loop
- Loops Revisited
- Break and continue
- Nested loops
- Nested loops - exercise
- Modulo operator
- Sum Of digit solution

JAVA Methods

- Methods
- Methods parameters and return type
- Method overloading

Java Some good Information

- Code Block, Indentation and statements
- Java literals

JAVA Object oriented programming concepts

- Constructors Introduction
- Default constructor
- Usage of constructors
- Understanding inheritance
- Working with inheritance
- Type of inheritance - Exercise
- Composition Introduction
- Composition Setting up
- Working with composition
- Adding functionality
- Encapsulation
- Polymorphism
- Benefits of Polymorphism

Java Core concepts

- Interfaces
- Abstract class - Introduction
- Multiple inheritance using interfaces
- Inner classes
- Types of nested class
- Local inner class
- Anonymous object
- Anonymous inner class
- Advantages of inner class
- User input
- Static elements
- Static inner class
- Final keyword
- Final keyword with method and class
- Packages
- Packages continues
- Scope
- Access modifier
- Access modifier
- Exception handling
- Exception handling, Multiple catch blocks
- Finally block
- Throw and throws
- User defined exception
- Checked and unchecked exceptions
- Enum
- Strings
- Difference between String literal and String Object
- String methods
- String formatting

CORE concepts

ARRAY

- Array overview
- How to initialize array
- Foreach loop
- Methods - Call by value and reference
- Issues with array

Collections framework Overview

- ArrayList
- Autoboxing and unboxing
- ArrayList and Stack overview
- Stack methods
- LinkedList
- LinkedList operations
- List interface
- Iterators
- Sorting and reversal of elements
- Compare To method overview
- Comparable interface

Generics

- Generics overview
- Getting started with generics
- Understanding generics
- Type parameters
- Generic method
- Bounded type parameter
- Comparable interface
- Wildcards in Generics

Collections continues

- Sets
- Set types
- Custom sorting using comparator
- Queue
- Dequeue
- Maps
- TreeMap
- Equals and hashcode methods
- Search under Maps

Multi threading concurrency Control In JAVA

- Bout multi-threading
- Multithreading overview
- Creating thread by extending the thread class
- Creating thread by implementing runnable interface
- Synchronization under concurrency control
- Synchronized method
- Synchronized block
- Synchronized block on object
- Static synchronization
- Usage of volatile keyword
- Wait and notify
- Interrupt
- Join overview
- Thread pools
- CountDownLatch
- Blocking Queue
- Reentrant lock
- Deadlock

Lambda Expression

- Lambda expressions
- Lambda expressions with variables and iterations
- Predicates

File Handling Under JAVA

- Data streams
- Creating file on disk
- Creating a directory
- Writing onto files
- Reading files with BufferedReader
- Reading files with scanner
- Scanner vs Buffered reader
- File deletion
- Try with resources
- Serialization of Objects

Debug-debugging

- Getting started with debugging
- Debugging, runtime variable value update
- Debugging Continues
- How to apply watch on variable(s)

ADVANCED JAVA concepts

GIT Basics

- GIT overview
- GIT features overview
- Setting up GIT
- Clone and import GIT project into eclipse
- Clone GIT project directly from eclipse

JSP & SERVLETS

- Before we start with JSP and Servlets
- Installation update 2021
- Project setup

- Hello Servlets
- Hello JSP
- Servlets life cycle
- JSP expressions element
- JSP scriptlets element
- JSP declarations element
- JSP comment element
- JSP Directive element

JSP & SERVLETS : Forms

- Forms overview
- Form elements
- Forms under JSP
- Forms under Servlets
- Basic form validations

JSP & SERVLETS :Understanding Java Beans

- Beans overview
- Bean scope types - session, page & application
- Request scope
- Beans with web forms

JSP & SERVLETS :Session Management

- Session under JSP overview
- Introduction to cookie
- Read and write operation of cookie
- User logout (Via cookie)
- User logout (using session attribute)
- Organizing application
- Fixing Redirect and forward links
- Handling session without cookie
- Servlet Filters

JSP & SERVLETS : JSTL Core TAGS

- JSTL setting up
- JSTL set and remove tags
- Reading from Bean using Expression Language
- Decision making under JSTL
- JSTL Choose and when tags
- JSTL for loop
- JSTL forEach loop
- JSTL forTokens tag
- JSTL import and param tag
- JSTL URL and redirect tag
- JSTL catch tag

JSP & SERVLETS JSTL Function Tags:

- STL length functions
- JSTL trim and escape Xml function
- JSTL more functions
- JSTL even more functions
- JSTL Split and Join function

JSP & SERVLETS JSTL/XML Tags:

- Adding Sample XML file
- JSTL XML Parse and Out tag
- JSTL XML ForEach and If tag
- JSTL XML Choose When otherwise

JSP & Servlets-Project on JSTL/18N:

- Setting up
- Adding properties
- Integration with website (i18n)

JSP & Servlets JSTL formatting Tags:

- Formatting date & number
- Formatting date & number (Document)

JSP & SERVLETS Building custom Tag:

- Building custom tag

SP & SERVLETS Integrated web template:

- Extracting header and footer from template
- Integrate template with project
- Integrate template with project using JSTL

JSP & SERVLETS Revisiting servlets:

- Servlets initialization
- Understanding more about Servlet

JSP & SERVLETS interacting with Database:

- Setting tools required
- Use of workbench
- Setting up JNDI
- Testing connection

JSP & SERVLETS Listing Data on webpage:

- Setting up
- Listing data on webpage
- Making use of include directive

JSP & SERVLETS add Record into database:

- Form implementation for record addition

- Organizing application
- Almost done
- Finalize the feature

JSP & SERVLETS Update Record into database:

- Upgrading the list
- Updating Controller
- Populating form
- Adding Update functionality

JSP & SERVLETS delete record from date base:

- Upgrading the list
- Adding delete functionality

JSP & SERVLETS adding JSTL Support:

- Add JSTL support

JSP & SERVLETS Image /File upload:

- Image upload form
- Handle image files
- Upload image onto filesystem

Hibernate Introduction:

- Let's Start with hibernate
- Hibernate Overview
- Installing MySQL & SQL workbench

Hibernate Frame work:

- Setting up Project
- Setting Up Hibernate Configuration File
- Session Factory and Session
- Adding Entity Class
- Hibernate in Action
- CRUD - Retrieving Record from Database
- CRUD - Updating a Record in Database
- CRUD - Deleting Record from Database

JSP & SERVLETS Integration of Hibernate

- Integrate Hibernate with JSP and Servlets
- Add Hibernate Support
- Understanding Hibernate configuration
- Hibernate entity class
- Hibernate in action

JSP & SERVLETS Building the application:

- List available files
- Display image files on JSP page
- Improve view of the page
- Adding update information form
- Implement update information functionality
- Update information logic revisited
- Update specific column data using Hibernate
- Add view image action
- Implement view image page
- Add delete image action
- Recheck the application working

SPRING Introduction :

- Downloading & Installing required software

SPRING Inversion of control Dependency:

- Dependency injection
- Understand dependency injection

SPRING Autowire:

- Spring
- Autowire introduction
- Autowire scenarios
- Qualifier annotation

SPRING Bean:

- Spring bean
- Constructor injection

SPRING IDE set up:

- Spring project on IntelliJ
- Adding SpringMVC support on Eclipse
- Simple dynamic web project
- Model in webproject

SPRING MVC

- Spring MVC minimal setup
- Basic form with Spring MVC
- Request param and Model under Spring MVC
- ModelAndView and foreach on data

Spring from elements

- Setting up eclipse project
- Getting started with forms
- Input and radio
- Radio-buttons and background
- Dropdown list
- Text area

- Checkboxes

Styling and external resources:

- Adding STS 3 support into Eclipse
- Add external resource
- Add style sheet

Spring frame work:Form validations

- Form Validation using HTML
- Hibernate validator introduction
- Repopulate form data
- ModelMap - Simplifying controller
- Show error messages
- Validation rules

Spring Frame work Database connectivity Using JDBC(XML Configuration)

- Maven Setup for hibernate and MySQL
- Alternate eclipse environment
- Dynamic web project approach
- Installing MySQL
- SQL workbench
- Understanding the Architecture
- Understanding the Architecture(Document)
- Methods of Mapping
- Methods of Mapping(Document)
- Setting Up Project
- Adding Data Access Object (DAOs)
- Defining Beans (XML Configuration)
- Reading from Database
- Showing Information on webpage

Spring Frame work Database connectivity Using JDBC(Annotations):

- Defining Annotations
- Getting Things in Place
- Adding Add User link
- A Walkthrough
- Updating the Controller
- Adding the Validation
- Adding User

Spring Frame work Exception handling under spring:

- Exception Handling

Restful APi(JAX-RS) Web services-Restful Api:

- Introduction To Web Services
- REST webservices overview
- Resource based URIs
- More about resource based URIs
- Rest response
- Status codes
- Idempotence of HTTP Methods
- Hypermedia as the Engine of Application State (HATEOAS)
- The Richardson Maturity Model
- JAX-RS and implementations overview

Restful APi(JAX-RS) Setting up:

- Setting up
- Background story
- Getting started with Restful API (Fixing warnings)
- Getting started with Restful API - First API
- Revisiting backgrounds and handling post request
- Resource based URI for CRUD methods
- PathParams

Restful APi(JAX-RS) Database connectivity(hibernet):

- Installing MySQL
- Creating sample database
- Setting up service
- Service layer continues
- Integrating hibernate
- Add DAO layer
- Implementing DAO layer
- XML response
- JSON response

Restful APi(JAX-RS) Understanding JAX-RS:

- Building client - introduction
- Client building steps
- URL building
- Post request
- Read response and wrapper class in brief
- Build HATEOS model
- Put and Delete operation

Restful APi(JAX-RS) Under the hood :

- Alternate Maven project setup
- Alternate application setup
- Message BodyReader and Writer
- Implement MessageBody Writer
- Resource types
- Param with resource types
- Param converters
- Param converter implementation
- Param converter implementation

Restful APi(JAX-RS) Building Client:

- Building client - Introduction
- Client building steps
- URL building
- Post request
- Read response and wrapper class in brief
- Build HATEOS model
- Put and Delete operation

SPRING BOOT Back ground :

- Setting up requirements
- Maven overview
- Set up Maven project for Spring Boot

SPRING BOOT Introduction :

- Building client - introduction
- Client building steps
- URL building
- Post request
- Read response and wrapper class in brief
- Build HATEOS model
- Put and Delete operation

SPRING BOOT SPRING MVC+Restful web services background:

- Setting up restful webapp
- Defining the rules
- Bootstraping Spring boot project

SPRING BOOT Building restful micro services with spring boot :

- Add view all posts functionality
- Add view specific post functionality
- Postman
- Add post functionality
- Fixing the issue
- Perform update operation
- Perform delete operation

SPRING BOOT Restful Micro services with Data base connectivity :

- Setting up
- Installing MySQL
- Setting up connection
- Create database table
- Configuring entity class
- Update Service layer
- Curd operations

Micro services with spring boot :

- Let's understand micro-service again
- Setting up Spring MVC project
- Spring boot micro-service
- Communicate with micro-service
- Improve app with best practices
- Improve the app - Customize White-label Error Page
- Improve the view - add JSTL support
- Micro-service discovery service overview
- Create Eureka server
- Eureka client
- Load balance

SPRING BOOT :

- About spring.jpa.hibernate.ddl-auto
- Spring DevTools

SPRING FRAMEWORK (Legacy) Spring Frame work-Introduction:

- About Spring
- Spring DevTools

SPRING FRAMEWORK (Legacy) :setting up

- Setting up System
- Setting up Environment
- Downloading Spring framework

SPRING FRAMEWORK (Legacy) :Inversion of control-Getting started(XML Config):

- Understanding the Scenario
- Coding the Scenario
- Understanding Inversion of control
- Implementing Inversion of Control
- ClassPathXmlApplicationContext

SPRING FRAMEWORK (Legacy) :Inversion of control-Continue (XML Config):

- Constructor Arguments
- Setting Bean Properties
- Setting Bean Property using P Namespace

SPRING FRAMEWORK (Legacy) :IOC –Dependency Injection (XML Config):

- Dependency Injection (Constructor argument)
- Dependency Injection (Property setter)
- Dependency Injection (Inner Bean)

SPRING FRAMEWORK (Legacy) :Understanding Spring bean(XML Config):

- What is a Bean?
- What is Bean Scope
- Bean scopes - Setting up
- Bean scopes - Singleton & Prototype Scope
- Bean LifeCycle
- Beans init and destroy

SPRING FRAMEWORK (Legacy) : Spring bean and collection (XML Config):

- Setting list as Constructor arg
- Setting Set as Constructor arg
- Setting Map as Property
- Setting Property as Property

SPRING FRAMEWORK (Legacy) : Auto wiring (XML Config):

- Getting Started with Constructor
- Constructor Autowiring (Removing Ambiguities)
- Autowiring Default Setting - Basics
- Autowiring Properties - Setting up
- Autowire byName
- Autowire byType

Project to deploy into AWS cloud

NEXSOFT IT SOLUTIONS
#First Floor, above Bank of Baroda,
Revathi center, Ramanaidupeta,
Machilipatnam, Andhra Pradesh.

WhatsApp or Call to +91 789 789 8511