

NANO SCIENTIFIC RESEARCH CENTRE PVT.LTD

AN ISO 9001:2008 CERTIFIED COMPANY



Java™



spring
boot

JAVA FULL STACK SYLLABUS



NANO SCIENTIFIC RESEARCH CENTRE PVT.LTD

SYLLABUS COPY

1.CORE JAVA



- 1.1. Introduction To Java
- 1.2. Java History
- 1.3. Why Java
- 1.4. Features of Java
- 1.5. Environment Setup
- 1.6. Hello World Application
- 1.7. Java Program Internal
- 1.8. JDK, JRE, JVM
- 1.9. JVM- Java Virtual Machine
- 1.10. Class, Objects, Data Types
- 1.11. Variables
- 1.12. Type Casting
- 1.13. Operators
- 1.14. Java Comments
- 1.15. Object Oriented Programming Concepts
 1. Abstraction
 2. Encapsulation
 3. Polymorphism
 4. Inheritance
- 1.16. Java Coding Standards
- 1.17. Package – Creating package, naming package
- 1.18. Java Keywords
- 1.19. Access Specifiers
 - Public, Private, Protected, Default
- 1.20. Static Concept
 - Block, Variable, Method, Class
- 1.21. Scanner Class
- 1.22. Control Statement
 - If, If...else, Nested if...else and Switch Case
- 1.23. Java Break Statement
- 1.24. Java Continue Statement
- 1.25. Flow Statement- for, while, do..while
- 1.26. Constructor- Default, Parameterized
- 1.27. Inheritance

- 1.28. Extends, super and this keyword
- 1.29. Method Overloading, Method Overriding
- 1.30. Interface
- 1.31. Abstract Class
- 1.32. Abstract Class vs Interface
- 1.33. Garbage Collection
- 1.34. Marker Interface
- 1.35. Serialization
- 1.36. Transient
- 1.37. Java instanceof
- 1.38. Instance initializer block
- 1.39. Volatile Keyword in Java
- 1.40. Exception Handling
- 1.41. Exception Handling
 - try, catch, throw, throws, finally
- 1.42. String Concept
 - String Basics
 - String Methods
- 1.43. String Buffer
- 1.44. String Builder
- 1.45. Custom Immutable Class
- 1.46. Java - Arrays
- 1.47. Collection Framework
 - Collection Basics
 - Iterator Interface
 - Methods of Collection Interface
1. List
 - A] Array List
 - B] Linked List
 - C] Vector
2. Set
 - A] Hash Set
 - B] Tree Set
3. Map
 - A] Hash Map
 - B] Tree Map
 - C] Hashtable



SYLLABUS COPY

1.48. Collection Framework Advance Concepts

1. Comparable Interface
2. Comparator Interface
3. Collections class
4. Differentiate Comparable & Comparator

1.49. Java Enum

1.50. Date Concept

1.51. File Handling Concept

1. File Handling Basics
2. Create Folder
3. Create File- .txt, .pdf, .xlsx, .docs
5. Read File
6. Delete File

1.52. Thread

1. Concept, Lifecycle
2. Extends Thread
3. Implement Runnable Interface
4. Thread Priorities
5. Thread Methods

1.53. Multithreading

1.54. Thread Synchronization

1. Synchronization
2. Object Locking
3. Inter Thread Communication

1.55. JDK 1.8.0 New Features with Hands-on

1. Lambda Expression
2. Functional Interface
3. Default Method in Interface
4. Static Method in Interface
5. Method References
6. Date Time API
7. Stream API in details

[All Advanced concepts as per Industry Practices]

8. Collectors
9. For each Loop
10. String Joiner Class
11. Parallel Sort
12. Optional Class

1.56. Java Mail API

- Realtime Live Implementation

Programming practices, Concepts & Interview Preparation

2.1 JDBC

1. JDBC Introduction
2. JDBC Architecture
3. Database Overview
4. JDBC Basics
5. MySQL
6. Create Database
7. Create Table
8. Insert, Update, Delete
9. Truncate
10. Java Application Using JDBC Connectivity
11. Handling SQL Exceptions
12. DriverManager
13. ResultSet
14. Connection
15. Statement
16. Prepared Statement
17. Callable Statement
18. DB Connectivity Steps
19. Store Image in SQL
20. Read Image in SQL
21. SQL Queries using SQLYog
22. JDBC CRUD Application

2.2 Servlet

- 2.2.1 Servlet Basics
- 2.2.2 Need of Server Side Programming
- 2.2.3 Servlet Life Cycle
 - 1 Init()
 - 2 Service()

2.1 doGet()

2.2 doPost()

3 Destroy()

2.2.4 Servlet Hello World Application

2.2.5 Web.xml Structure

2.2.6 Servlet Directives

- include(), forward(), sendRedirect()

2.2.7 HttpServletRequest,

HttpServletResponse in Servlet



SYLLABUS COPY

- | | |
|---|--|
| <ul style="list-style-type: none"> 2.2.8 Servlet and JDBC Integration 2.2.9 Servlet, HTML 5, MySQL-JDBC, Apache Tomcat using Real Time Login Application 2.2.10 Servlet, JSP, MySQL <ul style="list-style-type: none"> - JDBC, Apache Tomcat using Custom CRUD Application 2.2.11 Servlet, JSP, MySQL <ul style="list-style-type: none"> - JDBC, Apache Tomcat using Custom Sign Up Sign In Application 2.1 JSP <ul style="list-style-type: none"> 2.1.1 JSP Basics 2.1.2 Creating dynamic Web content with JSP 2.1.3 Scriplet 2.1.4 Declaration 2.1.5 Servlet, JSP, MySQL <ul style="list-style-type: none"> - JDBC, Apache Tomcat Login Application 2.1.6 Servlet, JSP, MySQL <ul style="list-style-type: none"> - JDBC, Apache Tomcat using User Sign Up Sign In Application 2.1.7 Servlet, JSP, MySQL <ul style="list-style-type: none"> - JDBC, Apache Tomcat using Custom CRUD Application with JDK Advanced features 2.1.8 Servlet, JSP, MySQL <ul style="list-style-type: none"> - JDBC, Apache Tomcat using Custom CRUD Application with different Date Patterns and time zones as per Industrial Practices 2.1.9 Servlet, JSP, MySQL <ul style="list-style-type: none"> - JDBC, Apache Tomcat Using Advanced Custom Project Implementation | <ul style="list-style-type: none"> 2.2. Spring MVC <ul style="list-style-type: none"> 1. Spring MVC Basics 2. Spring MVC Annotations 3. Spring MVC Hello World Application 4. Spring MVC Restful Web Services Basics 5. Spring MVC without Maven 6. Apache Maven Basics 7. Spring MVC with Maven 8. Spring MVC Application Deployment 2.3. Spring Boot <ul style="list-style-type: none"> 1. Spring Boot Introduction 2. Spring Boot Basics 3. Why Spring Boot 4. Spring Boot Features 5. Main Goal of Spring Boot 6. Creating Project- Spring Initializer 7. Spring Boot Hello World Application 8. Spring Boot Annotations 9. Spring Boot Devtool Dependency <ul style="list-style-type: none"> - Without Restart application will see changes on browser 10. Spring Boot Actuator 11. Spring Boot Build Systems 12. Spring Boot Code Structure 13. Spring Boot Tomcat Custom Port Number 14. Spring Boot with RESTful Service 15. Spring Boot with Logger 16. Deploy Spring Boot Applications on External Tomcat Server 17. Spring Boot – Common Application Properties (application.properties) 18. Spring Boot, Hibernate with RESTful Web Service 19. Spring Boot, Spring Data with RESTful Web Service 20. Spring Boot, Hibernate CRUD Application 21. Spring Boot, Spring Data JPA CRUD Application 22. Spring Boot, Spring- JdbcTemplate CRUD 23. Spring Boot, Hibernate, MySQL, Project Lombok CRUD Application |
|---|--|

2. FRAMEWORK



- 2.1. Spring Framework
 - 1. Spring Framework Introduction
 - 2. Spring Basics
 - 3. Spring Framework Features
 - 4. Inversion of Control
 - 5. Dependency Injection



SYLLABUS COPY

24. Spring Boot with XML Payload using Postman API
25. Spring Boot with JSON Payload using Postman API
26. Spring Boot Global Custom Validations
27. Spring Boot Global Custom Exception Handling
28. Spring Boot Code Debug Practices
29. Spring Boot Application connectivity with Multiple Databases
30. Spring Boot CORS Support
31. Spring Boot with Hibernate, Project Lombok, Swagger UI
32. Spring Boot with Spring Security- JWT
- 2.4. Spring Data- JPA
 1. Spring Data JPA Basics
 2. Spring Data JPA Introduction
 3. Spring Data JPA Architecture
 4. Spring Data JPA application.properties file
 5. JPA Repository
 6. CRUD Repository
 7. Spring Boot, Spring Data JPA CRUD using Postman API
 8. Spring Boot, Spring Data JPA HRM Application using Swagger UI
- 2.5. JUnit- Mockito Framework
 1. What is Unit testing
 2. What is JUnit
 3. JUnit Rules
 4. JUnit Arithmetic Application Example
 5. Mockito Introduction
 6. Mockito Annotations
 7. Mockito Setup
 8. What is Mocking
 9. Need for mocking
 10. Methods of Mockito
 11. Mockito mock() method
 12. Mockito when() method
 13. Mockito verify() method
 14. Mockito doNothing() method
 15. Mockito times() method

3. WEBSERVICES

- 3.1. SOAP Web Service
- 3.2. RESTful Web Service

4. MICROSERVICES Microservices

- 4.1. Microservices Introduction
- 4.2. Microservices Architecture
- 4.3. Microservices Project Structure
- 4.4. Microservices Frameworks
- 4.5. Spring Boot with Microservices Basics
- 4.6. Spring Boot with Microservices Coding Standards
- 4.7. Application.properties vs application.yml file
- 4.8. Spring Boot, Hibernate, Maven, MySQL CRUD Application- POSTMAN API
- 4.9. Spring Boot, Spring Data- JPA, Maven, MySQL CRUD Application- POSTMAN API
- 4.10. Spring Boot, Spring Data- JDBCTemplate, Maven, MySQL CRUD Application
- 4.11. Spring Boot, ORM, Microservices with Project Lombok
- 4.12. Spring Boot, CORS, Spring Data [JDBCTemplate], Microservices CRUD with Project Lombok using React JS
- 4.13. Creating Multiple Services using Spring Boot
- 4.14. Logger with Microservices
- 4.15. Rest Template Implementation
- 4.16. VO package creation for Multi Model Application
- 4.17. Set port number and application name for each service
- 4.18. Service Registry- Eureka Server
- 4.19. API Gateway for traverse centralize request
- 4.20. Hystrix DashBoard
- 4.21. Zipkin
- 4.22. Sleuth
- 4.23. Trace Application- System Health



NANO SCIENTIFIC RESEARCH CENTRE PVT.LTD

SYLLABUS COPY

- 4.24. Spring cloud API Gateway
- 4.25. Trace Fallback controller
- 4.26. Trace Services are up/down using controller service
- 4.27. Trace Dependent Services
- 4.28. Trace Controller | Methods and Services how many times used/hit
- 4.29. Advanced Level Research and Development for Microservices
- 4.30. Microservices HRM/Payroll Application
- 4.31. Microservices Customer | Product | Payment Order Application
- 4.32. Spring boot project with Microservice architecture Real time integration of spring cloud eureka, Spring cloud API Gateway & Hystrix
- 4.33. Spring Boot with Microservices all Advanced Interview Questions & Answers

5. WEB DESIGN

- 5.1. HTML 5
- 5.2. CSS 3
- 5.3. JavaScript
- 5.4. Bootstrap
- 5.5. Payment Gateway Integration
- 5.6. Domain and Hosting
- 5.7. Website Deployment and Go Live

6. REACT JS

React JS



- 6.1. React JS Basics
- 6.2. React JS Features
- 6.3. React JS Setup and Hello World Application
- 6.4. React JS JSX
- 6.5. React JS ES 6 Features
- 6.6. React JS Component
- 6.7. React JS State
- 6.8. React JS Props
- 6.9. React JS Constructor
- 6.10. React JS Lifecycle
- 6.11. React JS Events
- 6.12. React JS Router

- 6.13. React JS Forms
- 6.14. React JS Tables
- 6.15. React JS Portals
- 6.16. React JS ES6
- 6.17. React JS CSS
- 6.18. React JS Hook
- 6.19. React JS Redux
- 6.20. React JS Hooks
- 6.21. React use State Hook
- 6.22. React use Effect Hooks
- 6.23. React JS, Axios, API Integrations
- 6.24. React JS and Back End Integration
- 6.25. React JS Using Back End CRUD Application
- 6.26. Spring Boot, React JS QR Code Application
- 6.27. Spring Boot, React JS HRM Application
- 6.28. Spring Boot, React JS Ecommerce Application
- 6.29. React JS, JDK 1.8, Spring Boot, Data JPA, Maven, MySQL, Microservices, Project Lombok - Project Implementation

Address:

NANO SCIENTIFIC RESEARCH CENTRE PVT LTD

(AN ISO 9001:2008 CERTIFIED)

 #601 & 604, Siri Estates, Opp to Karur Vysya Bank, Opp Lane to R.S.Brothers, Ameerpet, Hyderabad.

 www.nanocdac.com

 +91 96406 48777- Aravind Reddy.K
+91 82975 78555- Mallikarjun.V