# Java
## Full Stack with
## GenAI Integration

React JS is the most popular JavaScript framework and platform for developing client-side (front-end) mobile and desktop web apps or single page applications (SPAs).

## HTML
### Introduction to HTML

- headings
- paragraph
- formatting tags
- image
- anchor
- lists
- table
- id
- class
- iframe
- div
- span
- form
- audio
- video
- html vs html5

## CSS
### Introduction to CSS

- ways to use CSS (inline , internal, external)
- css selectors
- css color
- border
- padding
- margin
- height
- width
- lists
- text
- display (none, inline , block , inline‑block)

- flex
- grid
- position
- z-index
- media

# Java Script

## Js Basics

- keywords
- identiers
- datatypes
- variables and constants
- tokens
- operators

# Decision Making Statements

## Conditions Statements Or Selection Statements

- if
- if else
- nested if else
- if else if ladder
- switch case

## Iteration Statements Or Loops

- for
- while
- do while
- for in
- for of

## Jump Statements

- continue
- break

## Functions

- function declarations,  denitions and calling
- types of functions
- anynomous functions
- at arrow functions or arrow functions
- self invoked function (IIFE)
- storage classication or  varibale scope (var let const and without keywords)
- recursion

## Strings Methods
## array Methods
## Global Execution Context and Function execution context

- scope chain rule and lexical scoping
- closure

### Objects

- literal and keys
- object functions
- nested objects

- Object Prototypes
- setter and getter methods

### Set and Map

### WeakMap and WeakSet

### Dates and Time

### OOPS

- literal and keys
- object functions
- nested objects
- Object Prototypes
- setter and getter methods

### DOM Manuplation

- DOM
- BOM
- event handling
- time based events

# Advanced Js

### How Javascript Works

- hoisting
- strict mode
- closure
- synchronous and asynchronous programming

### Event propagation

- bubbling
- capturing

### Event loop

### function curring

### ajax and json

### calling api using ajax

### callback hell

### promise

### async await

### fetch

### modules

### version

### Typescript

- Why Typescript
- Basic Types
- Class and Interfaces
- Modules

# React Js

### INTRODUCTION TO REACT JS

- What is React JS?
- What is SPA?
- DOM vs Virtual DOM
- Advantages and Disadvantages
- Key Features

## ENVIRONMENTAL SETUP

- Node | NPM
- Installation of CLI
- Setup Project
- Directory Structure
- Code Editors
- How React JS Application Boot

## BASIC FEATURES OF REACT JS

- React Concepts
- JSX and TSX
- Render Elements
- Function and Class Components
- Props and State
- Handling Events
- Dynamic Data Rendering
- Property Binding

## KEY FEATURES OF REACT JS

- Conditional Rendering
- List and Keys
- Forms Handling
- Forms Validations

## COMPONENT LIFECYCLE HOOK

- Understanding component lifecycle
- All Lifecycle Hooks

## EVENT HANDLING REACT

- Understanding React Event System
- Passing arguments to event Handlers

## NETWORK CALL

- Fetch
- Axios

## CUSTOM SERVICES

- Introduction to Services
- Building a Service

## LOCAL DATA STORAGE

- Local Storage
- Session Storage
- Cookies

## ROUTING WITH REACT ROUTER

- Setting up React Router
- Configuring route with Route Component
- Making routes dynamic with Route Params
- Working with nested routes
- Link and NavLink
- Redirect Routes

## UI COMPONENTS

- Material Design
- PrimeNG

## INTRODUCTION TO REDUX

- Why Redux
- Install and setup
- Store ,Reducer , actions
- Dispatcher
- High order Components
- mapStateToProps and mapDispatchToProps usage

### ADVANCE REDUX
- Async Actions
- Middleware
- Redux Thunk and Redux Saga

### React Hooks
- Why We Need Hooks.
- Different Types Of Hooks
- Using State And Effect Hooks
- Usereducer , Useref Etc.
- Custom Hooks
- Rules Of Hooks

### Third Party Modules
- Social Login
- Pagination
- Search
- Filter
- JWT Token
- File Upload
- Many More

### Rest Js Testing
- Jest with Enzyme

### Develop a CRUD Application in React Js

# Next js
- Introduction
- Routing
- Page Pre-Rendering (Server Side Rending) & Data Fetching
- Server-side rendering vs Client-side rendering
- Optimizing NextJS Apps
- Adding Backend Code with API Routes
- Working with App-wide State
- Adding Authentication

### Ecommerce Project

## INTRODUCTION TO JAVA
- Why Java was Developed
- Application Areas of Java
- History of Java
- Platform Independency in Java
- USP of Java: Java Features
- Sun-Oracle Deal
- Different Java Platforms
- Difference between JDK,JRE,JVM
- Java Versions
- JVM Architecture
- Installing Java on Windows
- Understanding Path Variable: Why Set Path

## CREATING FIRST JAVA PROGRAM
- Understanding Text Editors to Write Programs
- How to compile java file
- Byte Code and class file
- How to run class file

# JAVA LANGUAGE FUNDAMENTALS
- Identifiers
- Keywords
- Variables
- Literals
- Data Types
- Operators
- Comments
- Looping Statements
- Condition Statements
- Type Casting

## OOP IMPLEMENTATION (PIE)
- Why OOP
- OOP Concepts with Real life examples
- Class& it's Syntax
- Object& it's Syntax
- Reference Variable
- Constructors
- Instance(Non-Static)& Static Variables
- Instance(Non-Static) & Static Methods
- this Keyword and it's usages
- Object & Static Initializers(Anonymous Blocks)
- Inheritance& it's Syntax
- Types of Inheritance
- Object Class as Root of Java Class Hierarchy
- Variable Hiding
- Method Hiding
- Method Overriding
- Method Overloading
- Super keyword and it's usages
- Final keyword and it's usages
- Constructor Chaining
- Upcasting and Downcasting
- Static &Dynamic Binding
- Run Time Polymorphism
- Abstract Keyword(Abstract classes and methods)
- Understanding Interfaces
- Implementation of Encapsulation
- Association with Implementation

## PACKAGES
- Understanding Packages
- Setting Class path
- Reading Input from Keyboard
- Access Modifiers
- With in  Package & Outside Package Implements

## NESTED TYPES
- Static Nested Class
- Non-static Nested Class
- Local Class
- Anonymous Class
- Nested Interface

# ARRAYS
- General Definition of Array
- Advantages from Array
- Arrays in Java
- 1-d Arrays
- 2-d Arrays
- Jagged Arrays
- Array of reference type
- Operations on Arrays
- User Define Array & Object Tpye

# COMMAND LINE ARGUMENTS AND WRAPPER CLASSES
- How to read command line arguments
- Wrapper Classes
- Parsing of Numeric Strings
- String representation of Primitives

# EXCEPTION HANDLING
- Types of Runtime Errors
- Understanding Exceptions
- Exception Class Hierarchy
- Try & Catch Blocks
- Patterns of Catch Block
- Nested Try statements
- Throw, throws and finally
- Creating Custom Exceptions
- Checked & Unchecked Exceptions
- Assertion

# WORKING WITH STRINGS
- What is String
- String Class
- Creating String Object
- Operations on String
- String Buffer Class and it's Methods
- Difference between String and StringBuffer class
- String Builder Class and it's Methods
- Difference between StringBuffer and StringBuilder

# MULTITHREADED PROGRAMMING
- Multitasking: Why Concurrent Execution
- Multiprocessing  v/s Multithreading
- Main Thread (Default Java Thread)
- Creating Child Threads and understanding context switching
- Thread States
- Thread Group
- Thread Synchronization: Methods and Blocks
- Inter-Thread communication
- Daemon Threads
- Deadlock
- Concern Threads

# I/O STREAMS
- What is I/O
- Why Need Streams
- Byte Streams and Character Streams
- Read/Write operations with file
- Scanner Class
- Object Serialization& Deserialization
- Transient keyword
- File Class and it's Methods

## SOCKET PROGRAMMING
- Understanding Fundamentals of a Network
- Socket and ServerSocket Classes
- InetAddress Class
- DatagramSocket and DatagramPacket Classes
- URL,URLConnection,HttpURLConnection Classes

## REFLECTION
- Understanding the Need Of Reflection
- Getting information about class's modifiers, fields, methods, constructors and super classes
- Finding out constant and method declaration belong to an interface
- Creating an instance of the class whose name is not known until runtime
- Getting and setting values of an object's field if field name is unknown until runtime
- Invoking a method on an object if the method is unknown until runtime
- Invoking Private Methods

## EXTENDED & UTILITY CONCEPTS
- Generics
- Lambda Expression
- Annotations
- Object Cloning
- Vargs
- Static-import
- Enum
- Static, Default and Private Methods of Interface
- Var Type
- Java Modules
- Stream API

## COLLECTIONS FRAMEWORK
- What is Collection?
- What is Framework?
- Collections Framework
- Core Interfaces
- Collection, List, Queue,Deque
- Set,NavigableSet, SortedSet
- Map,NavigableMap, SortedMap
- Core Classes
- ArrayList, LinkedList,PriorityQueue,ArrayDeque
- HashSet,LinkedHasSet,TreeSet,
- HashMap,IdentityHashMap,WeakHashMap,LinkedHashMap,Tree Map
- Accessing a Collection via an Iterator
- Accessing List via ListIterator
- Accessing a Collection via for each loop
- Working with User Defined Objects
- The Comparator and Comparable Interfaces
- The Legacy classes and Interfaces.
- Enumeration, Vector ,Stack
- Hashtable, Properties

## DATE & TIME API
- java.util.Date
- java.util.Calender
- java.sql.Date

## SYSTEM PROPERTIES  & INTERNATIONALIZATION (I18N)
- Understanding Locale
- Resource Bundle
- Usage of properties file

- Fetching text from Resource Bundle
- Displaying the text in HINDI
- Displaying date in Hindi

## INTRODUCTION TO SQL(PROJECT BASED)

## DATABASE PROGRAMMING USING JDBC
- Need Of JDBC
- JDBC Drivers
- Statement, PreparedStatement, CallableStatement
- Scrollable and Updatable ResultSet
- Batch Updates
- Transaction
- Metadata
- Connection Data Base
  - Oracle
  - My SQL
  - Mongo DB

## JAVA EE(JAVA PLATFORM ENTERPRISE EDITION)
- Understanding the Concept of Java EE : JEE Specification
- Java EE Architecture
  - Single Tier
  - Two Tier
  - Three Tier
  - N-Tier
- Java EE Components
- Web Components
- Distributed(Business) Components
- Java EE Containers& Servers
- Web Container& Web Server(Apache Tomcat)
- EJB Container& Application Server(Weblogic,Glassfish,Websphere)
- Java EE Services
  - JNDI Service
  - Java Transaction Service
  - JAAS
  - 

## WEB PROGRAMMING
- Servlet
- JSP/JSTL
- Project Work

## JAVA SERVER PAGES (JSP) & JSTL
- JSP Architecture
- JSP Elements
- JSP Objects
- Understanding JavaBeans
- Custom Tags
- Using tags of JSTL
- Expression Language

## PROJECT CLASSES
- Front End Coding
- FORM DESIGNING
  - HTML
  - CSS
  - JAVA SCRIPT
  - BOOTSTRAP
- Back End Coding
- DATABASE DESIGNING
- Connecting forms to database
- Writing Business Logic
- Project Hosting

## DESIGN PATTERN
- Why Design Patterns…?
- Front Controller
- Composite View
- Factory Pattern
- Singleton Pattern
- DAO Pattern

## JAVA MAIL API
- Email System and Protocols
- Sending & Receiving Mails
- Handling Attachments

## INTRODUCTION TO DISTRIBUTED PROGRAMMING
- RMI
- Web Services

## INTRODUCTION TO RESTFULL SERVICES
- @PathParam
- @Path
- @FormParam
- @QueryParam
- @DefaultValue

## OVERVIEW OF JPA FRAMEWORK

# SPRING

## SPRING
- What is Spring?
- Spring modules
- Understanding dependency Injection
- Applying aspect-oriented programming

## MAVEN DEPLOYMENT
- Maven Configuration
- Converting Maven to Eclipse
- Various Maven Command

## BASIC BEAN WIRING
- Containing your Bean
- Creating bean
- Injecting into bean properties
- Auto wiring
- Controlling bean creation
- Aspect Oriented Programming

## HITTING THE DATABASE
- Learning spring's data Access Philosphy
- Configuring a data source
- Using JDBC with Spring
- Working with JDBC Templates
- Using Spring's DAO Support Classes for JDBC
- Integrating Hibernate with Spring
- Caching

# HIBERNATE

## INTRODUCTION TO ORM
- Need of ORM
- Problems using JDBC Directly
- ORM Implementation

## INTRODUCTION TO HIBERNATE
- Hibernate Architecture
- Hibernate configuration
- Hibernate's Support for Other Technologies
- Installing Hibernate
- A "Hello world" stand alone application

## CREATING PERSISTING CLASSES
- Mapping a basic Java Class
- Mapping a Class with Binary Data
- Mapping a Serializable Class
- Mapping a class with Data/ calendar attributes
- Mapping a Read-only class
- Mapping a class using Versioning /Timestamps

## MAPPING INHERITENCE WITH JAVA CLASSES
- Table-Per –class Hierarchy Mapping
- Table-Per –subclass Hierarchy Mapping
- Table-Per –concrete-subclass Hierarchy Mapping
- Persistence interfaces

## WORKING WITH COLLECTIONS
- Associations
- Lazy initialization
- Mapping Maps/Sorted Maps
- Mapping Sets/Sorted Sets
- Mapping lists
- Mapping Arrays
- Mapping a Bidirectional Association

## HIBERNATE CACHING
- How caching improves performence
- First level lache
- Second level cache

# SPRING BOOT

**Introduction To Spring Boot**

**Spring Boot Annotation**
**Spring Boot & JDBCTemplete**
**Spring Boot & JPA Hibernate**
**Spring Boot Rest API**

**Spring Boot MVC**
**Spring Boot Security**
**Introduction to Micro Services**

# Generative AI Integration

**Introduction to AI & Large Language Models**
**Understanding GPT Models**

- What is OpenAI
- How to get API key
- Understanding Library
- How to generate text
- How to perform sentiment analysis
- How to generate image from text
- How to implement chatbot

---

**Partners :** ORACLE
WORKFORCE DEVELOPMENT PROGRAM

redhat.
CERTIFIED
TRAINING PARTNER

UBI
Unica Business and Innovation Centre

edexcel

Java

---

**DUCAT**
**The IT Training School**

**E-mail: info@ducatindia.com**
**Visit us: www.ducatindia.com**
**www.facebook.com/ducateducation**

**NOIDA**
A-43 & A-52, Sector-16,
Noida - 201301, (U.P.) INDIA
📞 70-70-90-50-90
📱/🟢 +91 99-9999-3213

**SOUTH EXTENSION (DELHI)**
D-27,South Extension-1
New Delhi-110049
📞 70-70-90-50-90
📱 +91 98-1161-2707

**2.0 NOIDA SEC-63**
H-43 Sector-63
Noida-201301
📞 70-70-90-50-90
📱/🟢 +91 7042175774

**GURGAON**
1808/2, 2nd floor old DLF,
Near Honda Showroom,
Sec.-14, Gurgaon (Haryana)
📞 70-70-90-50-90

**GHAZIABAD**
1, Anand Industrial Estate,
Near ITS College, Mohan Nagar,
Ghaziabad (U.P.)
📞 70-70-90-50-90
   +91 9810851363

**PITAMPURA (DELHI)**
Plot No. 366, 2nd Floor,
Kohat Enclave, Pitampura,
( Near- Kohat Metro Station)
Above Allahabad Bank,
New Delhi- 110034.
📞 70-70-90-50-90