

## UpCurve - PE 2024 batch

### Duration

- 90 hours
- 30 sessions
- 3 hours per session
- 3 sessions per week
  - Tuesdays and Fridays between 5:30 PM and 8:30 PM
  - Saturdays between 2:00 PM and 5:00 PM

### Distribution of time

- Intro to DB - 3 sessions
- Java - 14 sessions
- DSA - 6 sessions
- Rest fundamentals - 1 session
- Springboot - 4 sessions



### Session 1: Introduction to Databases and SQL Basics

- Understanding the role of databases in modern computing
- Overview of different types of databases (relational, NoSQL, etc.)
- Introduction to Structured Query Language (SQL)
- Basic SQL Syntax and Data Types
- Creating Databases and Tables
- Inserting, Updating, and Deleting Data

### Session 2: Advanced SQL Queries and Data Manipulation

- Review of SQL Basics
- Retrieving Data with SELECT Statements
- Filtering Data with WHERE Clause
- Sorting Data with ORDER BY Clause
- Joining Tables with INNER JOIN, LEFT JOIN, RIGHT JOIN
- Grouping Data with GROUP BY Clause
- Aggregating Data with Functions (COUNT, SUM, AVG, etc.)
- Subqueries and Nested Queries

### Session 3: Database Design and Optimization

- Introduction to Database Design Principles

- Understanding Entity-Relationship (ER) Modeling
- Normalization and Denormalization
- Indexing and Query Optimization Techniques
- Transactions and ACID Properties

#### **Session 4: Java Programming Fundamentals**

- Review of basic Java syntax and concepts
- Data types, variables, and operators
- Control flow: if-else, loops (for, while, do-while)
- Methods and functions
- Introduction to debugging techniques

#### **Session 5: Object-Oriented Programming (OOP) in Java**

- Principles of OOP: encapsulation, inheritance, polymorphism, and abstraction
- Classes and objects
- Constructors and method overloading
- Inheritance and method overriding
- Introduction to access modifiers

#### **Session 6: Advanced OOP Concepts**

- Interfaces and abstract classes
- Packages and access specifiers
- Composition vs. inheritance
- Enumerations and nested classes
- Best practices in OOP design

#### **Session 7: Exceptions Handling**

- Understanding exceptions and error handling
- Handling exceptions using try-catch blocks
- Throwing and propagating exceptions
- Multiple catch blocks and finally block
- Best practices for exception handling

#### **Session 8: Input-Output (IO) Operations**

- Overview of Java IO API
- Working with streams: InputStream, OutputStream, Reader, Writer
- File input and output operations

- Reading and writing text files
- Binary file handling in Java

### **Session 9: File Handling in Java**



- Working with directories and files
- File manipulation: creating, deleting, renaming files
- File searching and filtering
- Managing file permissions and attributes
- Handling file I/O errors and exceptions

### **Session 10: Introduction to Collections Framework**

- Overview of Java Collections Framework (JCF)
- Lists: ArrayList, LinkedList
- Sets: HashSet, TreeSet
- Maps: HashMap, TreeMap
- Iterators and enhanced for loops

### **Session 11: Streams and Functional Interfaces**

- Introduction to streams in Java
- Stream operations: filter, map, reduce
- Working with functional interfaces
- Lambda expressions in Java
- Stream API best practices

### **Session 12: More on Collections**

- Collection interfaces: Collection, List, Set, Map
- Collection algorithms and utility classes
- Sorting and searching collections
- Customizing and optimizing collections
- Practical exercises and case studies

### **Session 13: JDBC Part 1**

- Overview of JDBC and database fundamentals
- Connecting to databases using JDBC
- Executing SQL queries and updates
- Handling result sets and metadata
- Best practices in JDBC programming

#### **Session 14: JDBC Part 2**

- Prepared statements and stored procedures
- Batch processing and transaction management
- Connection pooling and data source configuration
- JDBC drivers and database-specific features
- Error handling and debugging in JDBC applications

#### **Session 15: Introduction to JPA (Java Persistence API)**

- Understanding ORM (Object-Relational Mapping)
- Introduction to JPA and its architecture
- Setting up JPA with Hibernate or EclipseLink
- Mapping entities to database tables
- Performing basic CRUD operations using JPA

#### **Session 16: Advanced JPA Concepts**

- Entity relationships: One-to-One, One-to-Many, Many-to-One, Many-to-Many
- Lazy loading vs. eager loading
- Fetch strategies and performance optimization
- Transaction management in JPA
- Using JPQL (Java Persistence Query Language)

#### **Session 17: JPA Querying and Optimization**



- Advanced querying with JPQL and Criteria API
- Named queries and dynamic queries
- Caching and query optimization techniques
- Handling concurrency issues in JPA applications
- Profiling and performance tuning with JPA

#### **Session 18: Introduction to Data Structures and Algorithms**

- Overview of the course objectives and structure
- Importance and relevance of Data Structures and Algorithms
- Basic terminologies: Data, Data Structures, Algorithms
- Characteristics of good algorithms
- Big O notation and time complexity analysis
- Introduction to different types of data structures: Arrays, Linked Lists, Stacks, Queues

### **Session 19: Arrays and Linked Lists**

- Review of arrays: Declaration, Initialization, Accessing elements, Insertion, Deletion, Searching
- Advantages and limitations of arrays
- Introduction to Linked Lists: Singly Linked Lists, Doubly Linked Lists
- Operations on Linked Lists: Insertion, Deletion, Searching
- Comparison between arrays and linked lists

### **Session 20: Stacks and Queues**

- Introduction to Stacks: Definition, Operations (Push, Pop, Peek), Applications
- Implementation of Stack using arrays and linked lists
- Introduction to Queues: Definition, Operations (Enqueue, Dequeue), Applications
- Implementation of Queue using arrays and linked lists
- Comparison between stacks and queues

### **Session 21: Trees and Binary Trees**

- Introduction to Trees: Terminologies (Root, Node, Leaf, Parent, Child), Types of Trees
- Introduction to Binary Trees: Definition, Properties, Operations (Traversal, Insertion, Deletion)
- Binary Tree Traversal: Inorder, Preorder, Postorder
- Special types of Binary Trees: Binary Search Trees (BST), Balanced Binary Trees (AVL Trees)

### **Session 22: Sorting Algorithms**

- Introduction to Sorting: Importance, Terminologies (Stable Sorting, Comparison Sorting)
- Bubble Sort: Algorithm, Implementation, Time Complexity
- Selection Sort: Algorithm, Implementation, Time Complexity
- Insertion Sort: Algorithm, Implementation, Time Complexity
- Comparison of sorting algorithms

### **Session 23: Searching Algorithms and Conclusion**

- Introduction to Searching: Linear Search, Binary Search
- Linear Search: Algorithm, Implementation, Time Complexity
- Binary Search: Algorithm, Implementation, Time Complexity
- Comparison of searching algorithms
- Review of course content and key takeaways
- Discussion on further learning resources and career opportunities in DSA

**Session 24: Introduction to REST (Representational State Transfer)**

- Overview of Web Services
- Evolution of Web Architecture: From RPC to REST
- Principles of RESTful Architecture
- Understanding Resources, URIs, and Representations
- HTTP Methods: GET, POST, PUT, DELETE
- Status Codes and Headers in REST
- Demonstration: Basic CRUD Operations using RESTful APIs

**Session 25: Introduction to Spring Boot**

- Overview of Spring Framework
- Introduction to Spring Boot
- Advantages of Spring Boot
- Setting up development environment (IDE, JDK, Maven/Gradle)
- Creating a basic Spring Boot application
- Building and running the application

**Session 26: Spring Boot Configuration and Dependency Injection**

- Spring Boot auto-configuration
- Externalizing configuration properties
- Dependency Injection in Spring Boot
- Introduction to Spring annotations (e.g., @Autowired, @Value)
- Bean lifecycle in Spring Boot

**Session 27: RESTful Web Services with Spring Boot**

- Introduction to RESTful architecture
- Creating RESTful services with Spring Boot
- Handling HTTP methods (GET, POST, PUT, DELETE)
- Request mapping and URL patterns
- Consuming RESTful services using RestTemplate

**Session 28: Data Access with Spring Boot**

- Introduction to Spring Data
- Setting up data sources in Spring Boot
- CRUD operations using Spring Data JPA
- Working with repositories and entities
- Transaction management in Spring Boot

### **Session 29: Security in Spring Boot Applications**

- Introduction to Spring Security
- Authentication and authorization in Spring Boot
- Implementing security features (login, logout, session management)
- Securing RESTful APIs
- Role-based access control

### **Session 30: Advanced Topics in Spring Boot**



- Introduction to Spring Boot Actuator
- Monitoring and managing Spring Boot applications
- Spring Boot Profiles
- Spring Boot Testing
- Best practices and tips for developing Spring Boot applications
- Q&A and discussion on real-world applications of Spring Boot