

Customized Java-Basic to Advanced

Duration: 6 Days (8 Hours/Day)

1	Classes and objects	Define a class, Access modifiers, namely public, protected, default and private, Objects and object references, Constructors, new operator, 'this' reference, static class members.
2	Object oriented programming techniques	Encapsulation, Abstraction, Java's inheritance model and its language syntax, Super class reference variable, usage of the keyword super, Multilevel hierarchy, Method overriding, Dynamic method dispatch, or runtime polymorphism
3	Object Class & Packages in Java	Object class and its methods, Garbage collection. Describe finalization , Organize classes into Packages, Create and import packages, Built in packages in Java
4	Abstract Classes & Interfaces	Abstract classes, Interfaces, Marker Interfaces, Runtime polymorphism with interfaces, Nested classes and inner classes
5	Exception handling	Exception handling mechanism of Java, Use of the keywords that comprise Java's exception handling mechanism, Try with multiple catch, Nested try, Multiple catch involving Exception superclass and subclass, Checked and unchecked exceptions, Creating our own Exception classes, Rules governing checked exception.
6	I/O Streams	Define input and output streams in Java Define Byte streams and Character streams Define the predefined stream objects defined in the System class, namely in, out, and err Reading and writing from a file Copying images from one file to another file Object Serialization, Serializing inbuilt and user defined objects.

7	Collections & Generics	Describe the need for wrapper classes Define wrapper classes Auto boxing/unboxing Describe the context of using the Cloneable interface, Hierarchy of Collection Framework Methods of Collection interface Interfaces (Set, List, SortedSet, Queue etc.) and their methods Classes (ArrayList, Vector, LinkedList, PriorityQueue, HashSet, LinkedHashSet, TreeSet etc). Comparison between those classes Facilitate the processing of collections through the Enumeration & Iterator, Generics, Bounded Types and Unbounded Types, Object class equals and hashCode method in context of Collections, Comparable and Comparator Interface
8	Functional Interfaces and Stream API Date and Time API	FunctionalInterface Annotation java.util.function package Supplier, Consumer, Predicate, Function Stream API Sequential Streams and Parallel Streams with their implementations Date and Time API in Java8
9	Multithreading	Define multithreading Differentiate between multitasking and multithreading Describe Java's multithreading mechanism in the form of the Thread class and the Runnable interface Describe some key methods of the Thread class Describe Thread priorities Describe race conditions that are likely to occur between threads that are not synchronized Define thread synchronization through the use of the synchronized keyword Facilitate inter-thread communication with the help of wait(), notify(), and notifyAll() methods Java Concurrency Models Java Concurrency Vs Parallelism Thread Pools and Concurrency Utilities
10	JDBC	JDBC Architecture & JDBC Drivers CRUD operations, Establish a Connection Executing a Query Processing Results ResultSet Interface
11	JVM	JVM, It's Limitations, Weaknesses and Workarounds

	JUnit Tests With Mockito	Introduction to JUnit5 Test Cases & It's Annotations Assert class Mockito Framework Implementation And It's Advantages
12	Build Tool Maven	Introduction to Maven Why Use Maven? Maven Terminology Maven Lifecycle Maven Plugins Demo: Building a Project with Maven Benefits of Maven
13	Client Side and Server Side Architecture	2 Tier Architecture 3 Tier Architecture Layered Architecture
	Tomcat Servlet Engine	Servlets- Overview Servlets- Environment Setup Servlets- Life cycle, ServletConfig & ServletContext
14	Filters & Listeners	Filters, Life Cycle of Filters, Filter Pipeline, Listeners and Types of Listeners
15	Web Socket	ServerEndPoint, ClientEndPoint, OnOpen, OnClose, OnError, OnMessage
16	PostgreSQL	DML, DDL, DQL, Conditional Clauses, Group By, Having, JOINS, SubQuery
17	Introduction to SQL Server, Apache Cassandra	Introduction to Unstructured Database Cassandra
18	Spring Boot	Introduction to Spring Boot , Benefits and Implementation
19	Spring Data JPA	JpaRepository, Entity, Custom Methods in Repository
20	RestFul API Integration	Using SpringBoot to create RestFul API
21	HTML	HTML Basics, LIST, LINKS, TABLES, Image, FORMS
22	CSS	Introduction to CSS, Types of CSS, Selectors, Border, Text, Color etc
23	Java Script	JS Variables, Operators, Conditional Constructs, let ,const, var, Arrow Functions, Client Side Validation
24	Type Script	Introduction, Transpiler, Var Hoisting, Data Types, Types of Functions, Arrays & Tuples Object Oriented Concepts in TS
25	React	React .js, Core Principle(Components, State Management, Hooks, Life Cycle Methods)
26	React Styling	CSS Styling
		Dynamic Styling and Conditional Styling
		Dynamic List Content
		Managing List with Keys in React, Use Keys and Refs in React
	Custom React hooks	Building Custom React Hooks
		Managing States and Returning State Values

		Exposing Nested Functions from Custom Hooks
27	Sending Http Request & React Query / Transtack Query	Seding Http Request(GET,POST,PUT,DELETE)
		Handling Http request in React app with Ease
		Installing and using Transtack Query
28	Redux	How Redux Works?
		Redux vs React Context
		Redux Store
		Redux for State Management
		Store, Reducres, Containers, Actions, Providers
		Redux Thunk
		Finalizing the fetching Logic