# **Working with Java Data Types**

Primitives and Wrapper Classes

- Overview of primitive data types (int, double, etc.)
- Introduction to wrapper classes (Integer, Double, etc.)
- Operators and precedence
- Parentheses usage
- Type promotion and casting

Text Handling with String and StringBuilder

- String class and its methods
- StringBuilder class and its benefits
- String concatenation
- Using StringBuilder for mutable string operations

Local Variable Type Inference

- Introduction to local variable type inference (var keyword)
- Usage in lambda parameters

# **Controlling Program Flow**

Loops, if/else, and switch Statements

- Creating and using loops (for, while, do-while)
- If/else statements
- Switch statements

# **Java Object-Oriented Approach**

Declare and Instantiate Java Objects

- Creating objects and instantiating classes
- Nested class objects
- Object lifecycles (creation, dereferencing, garbage collection)

Fields and Methods

- Declaration and usage of fields
- Instance and static methods
- Overloaded methods

Initialization

- Instance and static initializer blocks
- Constructors and their usage
- Object initialization

Variable Scopes and Encapsulation

- Understanding variable scopes
- Applying encapsulation principles
- Creating immutable objects

Subclasses and Superclasses

- Creating and using subclasses
- Working with abstract classes

Polymorphism and Casting

- Utilizing polymorphism
- Casting objects and understanding reference types

Interfaces

- Creating and using interfaces
- Identifying functional interfaces
- Utilizing private, static, and default methods

Enumerations

- Introduction to enumerations
- Creating and using enums

# Exception Handling

Try/Catch/Finally and Multi-Catch Statements

- Handling exceptions using try/catch/finally
- Utilizing multi-catch statements

Custom Exceptions

- Creating custom exception classes

# Working with Arrays and Collections

## Generics and Wildcards

- Understanding generics
- Utilizing wildcards

## Arrays and Collections

- Working with Java arrays
- Using List, Set, Map, and Deque collections
- Sorting collections and arrays

# Streams and Lambda Expressions

### Functional Interfaces and Lambda Expressions

- Implementing functional interfaces with lambda expressions
- Java.util.function package

### Java Streams

- Filtering, transforming, and processing data with streams
- Decomposition and reduction
- Grouping and partitioning in sequential and parallel streams

# Java Platform Module System

- Deploying modular applications
- Automatic modules
- Declaring, using, and exposing modules
- Module services

# Concurrency

### Worker Threads and ExecutorService

- Creating worker threads with Runnable and Callable
- Managing concurrency with ExecutorService and java.util.concurrent API

### Thread Safety

- Developing thread-safe code
- Different locking mechanisms

# Java I/O API

- Reading and writing console and file data using I/O Streams
- Serialization and deserialization techniques
- Handling file system objects with java.nio.file API

## Secure Coding in Java SE Application

- Mitigating security threats
- Denial of service, code injection, input validation
- Ensuring data integrity
- Secure resource access

## Database Applications with JDBC

- Connecting to databases and performing SQL operations
- Processing query results using JDBC API

## Localization

- Implementing localization using Locale
- Resource bundles and Java APIs for parsing and formatting messages, dates, and numbers

## Annotations

- Creating, applying, and processing annotations
1.