



Fusion Software  
Institute

WWW.FUSION-INSTITUTE.COM

# JAVA DEVELOPMENT



# Java

# TABLE OF CONTENTS

## INTRODUCTION TO JAVA

- *History of Java*
- *Important Features of Java*
- *JVM Architecture*
- *JDK 1.8, JRE and JVM*
- *Java Keywords*
- *Simple Hello World Program*
- *Java Flow Control*
- *Source File*
- *Class and Objects*
- *Data Types*
  - *Primitive Data Types*
  - *Non - PrimitiveData Types*
- *Constructor*

## PACKAGES

- *Instance & Static Variable*
- *Static Block*
- *Has-A Relation (secondary Reference)*
- *Setter and Getter*
- *Method return type (primitive type and class type)*

# TABLE OF CONTENTS

## OBJECT ORIENTED PROGRAMMING (OOPS)

- **Inheritance (IS-A)**
- *Constructor and Inheritance*
- *Inheritance with parent reference and Child constructor*
- *Covariant Return Types*
- *Inheritance and type casting*
- *This/Super Keyword*
- **Polymorphism**
- *Method Overloading*
- *Constructor Overloading*
- *Compile time Polymorphism*
- *Overloading and Narrowing Concept*
- *Method Overriding*
- *@override Annotation*
- *Overriding and toString () method*
- **Abstraction**
- *Abstract class*
- *Interface*
- *Marker Interface*
- *Interface and Multiple Inheritance*
- *Interface Uses and Benefit with example*
- *Difference of Interface and Abstract class*

# TABLE OF CONTENTS

## • **Encapsulation**

- Access Modifier
- Data Hiding
- protected Access Modifier Implementation

## **EXCEPTION**

- Exception Handling:
- What and Why? (Try, Catch, Finally, Throw ThrowsTry with Resources)
- try and catch block
- Multiple catch blocks
- Nested try
- finally block
- finally block with returnstatement
- throw keyword
- Exception Propagation
- throws keyword
- throws keyword with MethodOverriding
- throws keyword with Constructor
- Custom Exception
- throw keyword and CustomException
- jdk 1.7 Feature AutoCloseable Interface
- jdk1.7 Feature try with Resources

# TABLE OF CONTENTS

## **STRING HANDLING**

- *String: What and Why?*
- *Immutable String*
- *String Comparison*
- *String Concatenation*
- *Substring*

## **MULTITHREADING**

- *Introduction*
- *Extends Thread Class*
- *Implementing Runnableinterface and Callableinterface*
- *Thread Life Cycle*
- *Demon Thread & Non-Demon Thread*
- *Locking System*
- *Inter Thread Communication (Producer-Consumer)*
- *Dead Lock System*
- *Thread Methods (sleep, join, yield etc.)*

## **SYNCHRONIZATION**

- *Synchronization: What and Why?*
- *Synchronized method*
- *synchronized block*
- *static synchronization*
- *Deadlock*
- *Inter-thread Communication*

# TABLE OF CONTENTS

## INPUT AND OUTPUT

- *FileOutputStream&File Input Stream*
- *ByteArrayOutputStream*
- *SequenceInputStream*
- *BufferedOutputStream & BufferedInputStream*
- *FileWriter & FileReader*
- *CharArrayWriter*
- *Input from keyboard by InputStreamReader*
- *Methods of String class*
- *String Buffer class*
- *String Builder class*
- *Creating Immutable class*
- *String Tokenizer*
- *Input from keyboard by Console*
- *Input from keyboard by Scanner Class*
- *PrintStream class*
- *PrintWriter class*
- *Compressing and UncompressingFile*
- *ReadingandWritingdatasimultaneously*
- *DataInputStream and DataOutputStream*
- *StreamTokenizer class*

## SERIALIZATION

- *Serialization & Deserialization*
- *Serialization with IS-A and Has-A*
- *transient keyword*

# TABLE OF CONTENTS

## COLLECTION FRAMEWORK

- *Java Collection*
- *Java List*
- *ArrayList*
- *Vector*
- *LinkedList*
- *Nested List*
- *List and Sorting*
- *List and Duplicate*
  - *Java Set*
  - *HashSet*
  - *TreeSet*
  - *LinkedHashSet*
- *Nested Set*
- *Java Map*
  - *HashMap*
  - *TreeMap*
  - *LinkedHashMap*
  - *Weak Hash Map*
- **Cursors**
  - *Iterator*
  - *List Iterator*
  - *Enumeration*
- **Enumeration**
- *Java Collections (Utility Class)*
- *Comparable & Comparator*

# TABLE OF CONTENTS

## JDBC

- *JDBC Drivers*
- *Steps to connect to the database*
- *Connectivity with Oracle*
- *Connectivity with MySQL*
- *Driver Manager*
- *Connection interface*
- *Statement interface*
- *Result Set interface*
- *PreparedStatement*
- *ResultSetMetadata*
- *Database Metadata*
- *Statement*
- *Prepared Statement*
- *Callable Statement*
- *CRUD Operations*

## LOGIC BUILDING

- *Operator*
- *java ternary operator*
- *if, else*
- *if, if else*
- *switch statement*
- *Conditional Related Problem*
- *Looping Control*
- *java for loop*



# TABLE OF CONTENTS

- *java while loop*
- *java do while loop*
- *java break Statement*
- *java continue statement*
- *Looping Related Problems*
- *Array*
- *Array Related Problem*
- *Numeric Logical Problems*
- *String Logical Problems (String Palindrome, DuplicateRemoval)*
- *Sorting Problems*
- *Searching Problems*
- *File Related Logical Problems*

## **SQL**

- *Database Introduction*
- *SQL Introduction*
- *MYSQL Introduction*
- *MYSQL Installation*
- *MYSQL Workbench Installation*
- *Features*
- *Data Types*
- *DDL, DML, TCL*
- *Create Table*

# TABLE OF CONTENTS

- *Select Statement*
- *Insert into Query*
- *Delete Query*
- *Update Query*
- *Constraints*
- *DISTINCT Clause*
- *WHERE Clause*
- **MYSQL Conditions**
  - *AND*
  - *OR*
  - *BOOLEAN*
  - *LIKE*
  - *IN*
- **MYSQL Functions**
  - *MIN*
  - *MAX*
  - *AVG*
  - *SUM*
  - *COUNT*
- *ORDER BY Clause GROUP BY Clause*
- *SQL JOINS*
- *SQL Outer Join*
- *SQL Inner Join*
- *SQL Left Join*
- *SQL Right Join*
- *SQL Full Join*
- *Stored Procedure*

# TABLE OF CONTENTS

## “FULLSTACK AND ADVANCEFRAMEWORK ”

- **Hibernate (Hibernate 5.X)**

- *Introduction*
- *ORM (Object Relational Mapping)*
- *Advantage of ORM*
- *ORM Tolls List*
- *Hibernate architecture*
- *JPA*
- *Hibernate with JPA*
- *Installation and Directory Structure*
- *Hibernate Advantages over JDBC*
- *Hibernate and Annotation*
- *Hibernate Configuration File*
- *Hibernate with Java Based(Zero XML file)*
- *Hibernate Utilityfile*
- *Load and get methoddifference*
- *save,saveorUpdate,persist method*
- *Inheritance in Hibernate( IS-A )*
- *Default Inheritance SingleTable Joined*
- *Table Per class*
- *Hibernate Mapping(Has-A)*
- *One-To-One One-To-Many Many-To-One Many-To-Many*
- *Hibernate QueryLanguage (HQL)*
- *Hibernate Annotations*
- *Caching in Hibernate*
  - *First Level Cache*

# TABLE OF CONTENTS

- *Scalar queries*
  - *Native SQL*
  - *Criteria Builder*
  - *Name Queries*
  - *Procedures*
- *Criteria Builder*
- *Hibernate Transaction Management*
- *CRUD Operations*

## **SPRING:-**

### ***Spring IOC***

- *What is Spring*
- *Spring Modules*
- *What is IOC*
- *Spring IOC Container*
- *Bean Factory/Core Container*
- *Application Context/Advance /J2EE Container*
- *Spring Bean life Cycle*
- *Bean Scope*
- *Singleton Scope Prototype Scope*
- *Lazy and Eagar LoadingConcept*
- *Scope Combination*
- *Dependency Injection Setter Based Constructor Based*
- *Spring IOC Container*
- *Setter Based XML & Java*

# TABLE OF CONTENTS

## **Spring Web MVC:**

- *Spring MVC*
- *Features Of Web MVC*
- *Life Cycle of Web MVC*
- *Execution Flow of MVC*
- *Create the Request and view Pages  
CreateController*
- *Configure DispatcherServlet Configure  
viewResolver*
- *Stereotype Annotation in MVC @Component  
@Controller*
- *@Service @Repository*

## **Spring MVC –Multiple Controllers**

- *RequestMapping –Working with Parameters  
@RequestParam  
@PathVariable*
- *Handler Mapping*
- *Controller Class*
- *View Resolvers*
- *Form Handling*
- *Spring Form Validation*
- *Spring Hibernate Integration (XML and Java  
Based)*
- *Mini Project*

# TABLE OF CONTENTS

## **Spring Boot:**

- Spring Boot Introduction
- Spring Boot Features
- Advantages Over SpringWeb MVC
- Creating Spring Boot Application Using Maven
- Using Spring Initializer (<http://start.spring.io/>) Using Spring STS IDE
- Removal Of XML Files
- Simple Application Using Spring Boot
- Spring Boot Starters
- SpringBootApplication Annotation
- Spring Application Class
- Spring Boot Actuator.
- Embaded Servletcontainers
- Spring Boot Profiles DEV
- PROD UAT
- Spring Boot With Data JPA (CRUD Repository)
- Mini Project Using Spring Boot

## **Spring Boot JPA:**

- What is Spring Boot JPA?
- JPA Annotations
- Curd Repository
- JPA Repository
- Custom Queries
  - Using Method Name
  - JPA Named Queries
  - Query Annotation

# TABLE OF CONTENTS

## **User Interface**

### **HTML**

- What is html?
- Html paragraphs
- html headings
- Links
- Html forms
- Use of get()method
- Use of post()method
- Html table
- how to use radio buttons and check box

### **CSS**

- What is CSS and its Syntax
- Ways to insert CSS in html page
- Internal CSS
- External CSS
- Inline CSS
- Types of Selector
- id selector
- class selector
- element selector

### **JavaScript**

- what is JavaScript
- ways to insertJavaScript in html page
  - in head
  - in body

# TABLE OF CONTENTS

## Ajax

- What is Ajax?
- How Ajax works
- Xml Http request Object methods
- Xml Http request Objectproperties
- What is Synchronous requestand Asynchronous request
- What is Gson?And its methodsto convert objectinto String.
- What is Jackson?

## Angular :-

- Introduction of Angular10
- Basic Architecture of AngularApplication
- Angular SemanticVersioning
- Environment Setup
- Creating First Angular Application
- Folder Structure
- Angular CLI
- Typescript Sessions
- Angular Building

## “Projects Topics”

### Maven

- Introduction
- POM.XML Structure
- Maven Installation and Configuration



# TABLE OF CONTENTS

## **SDLC (Software Development Life Cycle)**

- Waterfall model
- Agile Methodology

## **Log4j**

- Introduction of Log4J
- Components of Log4J
- Basic Program
- Log4j Properties file
- Console Properties and FileProperties

## **Junit**

- Introduction of Junit
- Annotation
  - Beforeclass
  - Afterclass
  - After
  - Before
  - Test
- Assertion class

## **2 Real Time Project As Per Industry Standard**