GEEKONIK

JAVA

Backend

Development

Training Cirriculum

GEEKONIK

J2SE:[Core Java]:

- Introduction:
  - Java History
  - Differences between java and others
  - Java Features
  - Java Naming Conventions
- First Java Application Development:
  - Java Installation
  - Editor
  - Java Application and Java File Saving
  - Compile Java File
  - Execute Java Applications
- Language Fundamentals:
  - Operators
  - Identifiers
  - Literals
  - Data Types and Type casting
  - Java Statements
  - Arrays
- OOPS
  - Class
  - Object
  - Encapsulation
  - Abstraction
  - Inheritance
  - Abstraction
  - Polymorphism
  - Message
  - Passing

- Object Based PL VS Object Oriented PL
- Object Based PL VS Object Oriented PL
- Class syntax
- Method Syntax
- Var-arg method.
- Accessor Methods VS Mutator Methods
- Syntax to create an object
- Immutable Objects VS Mutable Objects
- Object Vs Instance
- Constructors
  - Default Con.
  - User defined con.
  - 0-arg-con.
  - param-con.
- Instance Context
  - Instance variable
  - Instance block.
  - Instance method
- This keywords
  - To refer current class variable.
  - To refer current class methods.
  - To refer current class blocks.
  - To return current class objects.
- Static keyword
  - Static variable
  - Static block
  - Static method
  - Static import

- Object Based PL VS Object Oriented PL
- Object Based PL VS Object Oriented PL
- Class syntax
- Method Syntax
- Var-arg method.
- Accessor Methods VS Mutator Methods
- Syntax to create an object
- Immutable Objects VS Mutable Objects
- Object Vs Instance
- Constructors
  - Default Con.
  - User defined con.
  - 0-arg-con.
  - param-con.
- Instance Context
  - Instance variable
  - Instance block.
  - Instance method
- This keywords
  - To refer current class variable.
  - To refer current class methods.
  - To refer current class blocks.
  - To return current class objects.
- Static keyword
  - Static variable
  - Static block
  - Static method
  - Static import

- Main () method
  - Public static void main (String [] args)
  - Why public?
  - Why static?
  - Why void?
  - Why main?
  - Why String [] as parameter?
  - Is it possible to overload main (-) method?
  - Is it possible to override main (--) method?

- Is it possible to provide more than one main (--) method with in a single java appl?
- Is it possible to execute any java application without using main method?
  - Factory Method
  - Singleton classes and Doubleton classes
  - Final Keyword
    - Final variable
    - Final class
    - Final method
  - Enum keyword
  - Relationships in JAVA
    - IS-A Vs HAS-A Vs USE-A
  - Assiciations in Java
    - one-one
    - many-one
    - one-many
    - many-many
  - Inheritance and Types of inheritances

- Single
- Hierarchical
- Multipl e
- Hybrid.
- Multilevel

- Staic flow in inheritance
- Instance flow in inheritance
- Super keyword
- Class level type casting
- Poly Morphism
- Static PM
- Dynamic PM
- Method overloading
- Method overriding
- Abstract Methods Vs Concreate Methods
- Abstract class Vs concrete Class
- Class Vs Abstract class Vs interface
- "Instance of" operator
- What is Adapter class?
- What is marker interface?
- Object Cloning
  - Shallow Cloning
  - Deep Cloning
- JAVA8 features in interfaces

Inner classes

- Member Inner class
- Method local Inner class
- Static Inner class
- Anonymous Inner class

- Wrapper classes
  - Byte
  - Short
  - Integer
  - Long
  - Float
  - Double
  - Boolean
  - Character
- Packages:
  - What is a package?
  - Adv. of packages
    - Modularity
    - Reusability
    - Abstraction
    - Sharability
    - Security
  - Types of packages
    - Predefined packages
    - User defined packages
  - Jar files preparation
  - Executable Jar files
  - Batch files preparation
- String manipulations:
  - String
  - String Buffer
  - String Builder
  - String to kenizer

- Exception Handling:
- Error VS Exception
- Exception Def.
- Types of Exceptions
  - Predefined Exceptions
  - User defined Exceptions
- Checked Exception VS Unchecked Exception
  - Pure Checked Exceptions
  - Partially Checked Exceptions
- Throw Vs throws
- try-catch-finally
- Custom Exceptions
- Java7 Features in Exception Handling
  - Automatic Resource management
  - Multi catch block.
- Multi-Threading:
  - Process Vs Processor Vs Procedure
  - Single Processing Mech. Vs Multi Processing Mech.
  - Single Thread model And Multi Thread Model
  - Thread Design
    - Extending Thread class
    - Implementing Runnable interface.
  - Thread lifecycle
    - New/Born
    - Runnable
    - Running
    - Blocked
    - Dead
    - Thread class library

- Sleep ()
- Join ()
- Yield ()
- Stop ()
- Daemon Thread
- Synchronization
- Inter Thread communication
- Wait ()
- Notify ()
- Notify All()

- IOStreams:
  - What is stream?
  - Types of Streams?
  - 1Byte-oriented Stream
    - Input Streams
    - Output Streams
  - Character-Oriented Streams
    - Reader
    - Writer
  - File Input Stream Vs File Output Stream
  - File Reader Vs File Writer
  - File Vs Random Access File
  - Serialization vs Deserialization
  - Externalization
- Networking:
  - Standalone Appl. Vs Distributed Appl.
  - Client-Server Arch.
  - Socket Vs Server Socket
  - Network Appl. Arch.
  - Socket Programming.

- Reflection API:
  - Class
  - Field
  - Method
  - Constructor
- Annotations:
  - What is Annotation?
  - Adv of annotations
  - Comments Vs Annotations
  - Types Of annotations
  - Built-in Annotations
  - @Override @Inherited @Deprecated @Target @Suppress Warnings @Documented @Retention
  - User Defined Annotations
- Regular Expressions:
  - Introduction
  - Pattern
  - Character
  - Quantifiers
- Garbage Collection:
  - Introduction
  - Approaches to make an object for GC
  - Methods for requesting JVM to run GC
  - Finalization
- JVM Arch.
  - Class Loading Sub System
  - Memory Management System
  - Execution Engine
  - Java Native Interface
  - Java Native library

- Generics:
  - . Introduction
  - Generic Classes
  - Generic Methods & Wild Card Character.
  - Inter Communication with Non-Generic Code

ADV. JAVA Course Content

- Collection Framework:
  - Collection Arch.
    - List and its implementations
    - Array List3. 2.vector.LinkedList.4stack
    - Set and its implementations
      - Hash Set
      - Linked Hash Set
      - Tree Set
    - Map and its implementations
      - Hash Map
      - Hash table
      - Properties
      - TreeSet
      - Queue and its implementations
    - Priority Queue
    - Blocking Queue
    - Priority Blocking Queue
    - Linked Blocking Queue
    - Iterators
      - Iterator
      - List Iterator
      - Enumeration

JDBC:

- Storage Areas
  - Temporary Storage Areas
  - Permanent Storage Areas
- Query Processing System
  - Query Tokenization
  - Query Processing
  - Query Optimization
  - Query Execution
- Driver and Driver Types
  - Type 1 Driver
  - Type 2 Driver
  - Type 3 Driver
  - Type 4 Driver
- Steps to design JDBC Applications
  - Load and register the Driver.
  - Establish the connection between Java Application.
  - Prepare either Statement or prepared Statement or Callable Statement Objects.
  - Write and execute SQL Queries.
  - Close the connection.
- Prepared Statement
  - Prepared Statement with insert sql query
  - Prepared Statement with update sql query
  - Prepared Statement with select sql query
  - Prepared Statement with Dates Handling
  - Prepared Statement with Batch Updations

Callable Statement

- CallableStatement with procedure
- CallableStatement with function
- CallableStatement with CURSOR Type Procedure
- CallableStatement with CURSOR type function

Transaction Management

- Atomicity
- Consistency
- Isolation
- Durability
- Savepoint
- Batch Updations

SERVLETS:

- Introduction
  - Enterprise Applications
  - Web Applications b. Distributed Applications
  - Client-ServerArch
  - Client
  - Protocol
  - Server
  - Servlets Design
- Servlet interface
- Generic Servlet
- Http Servlet
- Servlet Lifecycle
- Servlet Config
- Servlet Context
- Servlet Communication

- Browser-servlet
  - SendRedirect Mechanism
  - Web-component
    - Include Mechanism
    - Forward mechanism
  - Session Tracking Mechanisms
    - HttpSession Session Tracking Mechanism
    - Coockies Session Tracking Mechanism
    - URL-Rewriting Session Tracking Mechanism
    - Hidden Form Fields Session Tracking Mechanism
- Servlets Filters

JAVA SERVER PAGES:

- Introduction
- 2SP Life Cycle
- JSP Elements
  - JSP Directives
  - Scripting Elements
  - JSP Actions
  - JSP Directives
- Page Directive
- Include Directive
- Taglib Directive
- JSP Scripting Elements
  - Declarations
  - Scriptlets
  - Expressions
- JSP implicit objects

- Out
- Request
- Response
- Config
- Application
- Session
- Exception
- Page
- Page Context

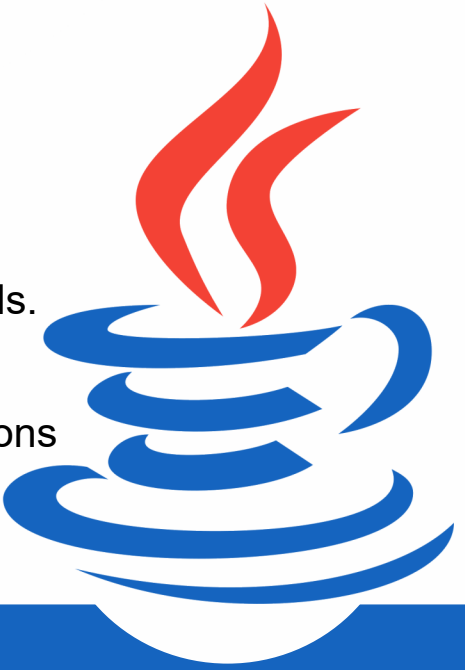
- JSP Standard Actions
  - <jsp:useBean>
  - <jsp:setProperty>
  - <jsp:getProperty>
  - <jsp:include>
  - <jsp:forward>
  - <jsp:param>
- JSP Custom Actions
  - Tag
  - IterationTag
  - BodyTags
- JSTL
  - Core Tags
  - XML Tags
  - Expression Language
  - EL operators
  - EL implicit objects.
  - EL functions.

SPRING Course Content

- Introduction :
  - Enterprise Appl
  - Enterprise Application Layers
  - Presentation Layer
  - Business Layer
  - Data Access Layer
- System Architectures
  - 1-Tier Arch.
  - 2-Tier Arch.
  - n-Tier Arch
- Types of Enterprise Applications.
  - Web Applications
  - Distributed Applications
- Modeled Arch.
  - Model-I Arch.
  - Model-II Arch.
- MVC
- Requirement to user Frameworks
- Types of Frameworks
  - Web Frameworks
  - Application Frameworks
- Differences between Spring and Struts, JSF
- Spring History

Core Module

- Introduction
- IOC Containers
  - BeanFactory
  - XmlBeanFactory
- 2ApplicationContext
  - ClassPathXmlApplicationContext
  - FileSystemXmlApplicationContext
  - WebXmlApplicationContext
- Beans in Spring Framework
  - Beans Definition
  - Beans Configuration
  - XML Based Configuration
  - Annotation Based Configuration
  - Java Based Configuration
- Bean Scopes
  - singleton Scope
  - prototype Scope
- Bean Lifecycle
  - Bean Loading
  - Bean Instantiation
    - By Constructor
    - By Static Factory Method
    - By Instance Factory Method
  - Bean Initialization and Destruction
    - By Custom initialization and destruction methods.
    - By InitializingBean and DesposableBean
    - By @PostConstruct and @Predestroy Annotations



Spring ORM

- Introduction
- Hibernate Integration with Spring
  - Hibernate Introduction
  - Hibernate Application Development
  - Spring with Hibernate Integration.

Aspect Oriented Programming [AOP]

- Introduction
- AOP Terminalogy
- Aspect
- Advice
- JoinPoint
- Pointcut
- Introduction
- Target
- Proxy
- Weaving
- Advisor
- Types of AOPs
  - Proxy Based AOP
  - Declarative Based AOP
  - Annotation Based AOP
- Advices
  - Before Advice
  - After Advice

Spring Transactions

- Introduction
- Declarative Based Transactions.
- Annotation Based Transactions

Spring web MVC Module

- Introduction
- Spring MVC Flow
- Controllers
- Handler Mappings
  - BeanNameUrlHandlerMapping
  - SimpleUrlHandlerMapping
- HandlerInterceptor
- ViewResolvers

HIBERNATE Course Content

- Introduction
  - Enterprise
  - Enterprise Appli cation
  - Enterprise Applicati on Layer
  - User Interface Layer
  - Business Processing Layer
- 3. Data Storage and Access Layer
  - Data Persistency
  - Hibernate History
  - Hibernate Features
  - Hibernate Arch.
- Persistence Class / POJO class
  - Mapping File
  - Hibernate Configuration File
  - Client Application
- Hibernate Applications
  - Hibernate Application with Main Clas s as Client.
  - Hibernate Application with Servlet as Client.
  - Hibernate Application with JSP Page as Client.
  - Hibernate Basic Annotations [Without Mapping File]
  - Hibernate Application without Configuration File
- Hibernate Persistence Object Lifecycle
  - Transient State
  - Persistent State
  - Detached State
  - Removed State

- Primary Key Generation Algorithms [XMl and Annotations]
  - Assign
  - Increment
  - Sequence

Hibernate Query Language [HQL]

- HQL Elements
  - Clauses
  - From' Clause
  - 'Select' Clause
  - 'Where' Clause
- Aggregate Functions
  - count(-)
  - sum(-)
  - min(-)
  - max(-)
  - avg(-)

Criteria API

Hibernate Mappings

- Basic 'OR' Mapping
- Component Mapping
  - Inheritance Mapping
  - Table per Class Hierarcy
  - Table per Sub-Class
- Associations Mapping
  - One-To-One Association
  - One-To-Many Association

GEEKONIK

GEEKONIK

CONTACT

www.geenonik.com
9990161025