JAVA COURSE CONTENTS

Introduction to HTML Concepts

- Form Elements, DHTML
- Nested Tables

Getting Started

- Describe the key features of Java Technology.
- Write, complete and run a simple Java Technology application.
- Describe the functions of Java Virtual Machine (JVM).
- NOTE: The terms “Java Virtual Machine” and “JVM” means a virtual machine for the Java Platform.
- Define Garbage Collection.
- List the three tasks performed by the Java Platform that handles code security.

Object Oriented Programming

- Define modeling concepts abstraction, Encapsulation and packages.
- Discuss why you can reuse Java Technology Application Code.
- Define class, member, attributer, method, constructor and package.
  Use the access modifier private and public as Appropriate for the guidelines of encapsulation.
- Invoke a method on a particular object.
- Use the Java Technology Application Programming Interface (API) online documentation.

Identifiers, Keywords & Types

- Use comments in source program.
- Distinguish between valid and Invalid Identifiers.
- Recognize Java Technology keywords.
- List the eight private types.

- Define literal values for numeric and textual Types.
- Define the terms private variable and Reference Variable.
- Declare Variable of class type.
- Construct an object using new.
- Describe default initialization.
- Describe significance of a reference Variable.
- State consequence of assigning variables of class types.

Expression & Flow Control

- Distinguish between Instance and Local Variables.
- Describe how to initialize instance variables.
- Identifiers and correct a possible reference before assignment compiler error.
- Recognize, Describe and Use Java Software Operators.
- Distinguish between legal and illegal Assignments of primitive types.
- Identify Boolean Expressions and their Requirements in control; constructs.
- Recognize assignment compatibility and require casts and fundamental types.
- Use if ,switch, for, while and do constructions And the labels forms of broke and continue as flow control structure in a program,

Arrays

- Declare and create arrays of primitive, class or array types.
- Explain why elements of and arrays are Initialized.
- Explain how to initialize the elements of an array.
- Determine the number of elements in an array. Create a multidimensional array.

Class Design

- Define inheritance, polymorphism, overloading and virtual method invocation.
- Use the access modifier protected And the default (package friendly)
- Describe the concept of constructor & method overloading.
- Describe the complete object constructor and Initialization operation.

Advance Class Features

- Create static variables, method, and Initializes.
- Create final classes, methods and Variables.
- Create and use enumerated types.
- Use the static import statement

- Create abstract classes and method.
- Create and use an interface.

Exceptions

- Define exceptions
- Use try, catch and finally statements.
- Describe exception categories.
- Identify common exception.
- Develop programs to handle your own Exception.
- Use assertions.
- Distinguish appropriate and inappropriate uses of assertions.
- Enable assertions at runtime.

Text-Based Applications

- Write a program that uses command- Line argument and system properties.
- Write a program that reads from standard input.
- Describe the C-type formatted input and output.
- Write a program that can create, Read and write files.
- Describe the basic hierarchy of collections in Java 2 Software Development kit (Java 2 SDK).
- Write a program to iterate over a collection.
- Write a program that uses generic collections.

Building Java GUIs

- Describe the Abstract Windowing Toolkit (AWT) package and its components.
- Define the terms containers, components and layout managers and Describe how they work to get to build a GUI.
- Use layout managers.
- Use the frame and panel containers appropriate.
- Describe how complex layout with nested containers work.

GUI Event Handling

- Define event and event handling.
- Write code to handling events that occurs in a GUI.
- Describe the concepts of adapter Classes, including how and when to use them.
- Determine the user action that originated the event from the event.

- Identify the appropriate listener interface for a verity of event types.
- Create the appropriate event handle methods for the verity of event types.
- Understand the use of inner classes and anonymous classes in event handling.

GUI- Based Applications

- Identify the key AWT components and the event that day trigger.
- Describe how to construct a menu and menu items in a Java GUI.
- Understand how to change a color and font of a component.

Threads

- Define a thread.
- Create Separated thread in a Java technology program, controlling the cone and data that are used by the thread.
- Control the execution of a thread and write platform independent codes with threads.
- Describe the difficulties that might arise when multiple thread share data.
- Use wait and notify the communicate between threads,
- Use synchronized to protect data from corruption.

Advance U/o Streams

- Describe the main features of the Java to Package.
- Construct node and processing Streams and use the appropriately.
- Distinguish readers and writers from Streams and select appropriately between them.

Networking

- Develop code to setup the Network Connection.
- Understand the Transmission Control Protocol/ Internet Protocol (TCP/IP).
- Use Server Socket and Socket Classes for implementation of TCP/IP clients and servers.

ADVANCE JAVA CONTENTS

Swings

- AWT and Swing
- GUI Programming
- Layout Management
- Event Handling

Database Connectivity

- JDBC Connectivity Modl
- Database Programming
- Connecting to the Database
- Executing SQL Queries
- Result Set Meta Data
- Executing SQL Updates
- Using prepared Statement
- Parameterized Statement

Remote Method Invocation

- Distributes Application
- RMI Server
- RMI Client
- Stood and skeleton
- RMI Classes and interfaces
- Create Remote Object
- RMI Registry Security Manager
- Using Policy tool.

Servlets

- Browsers, Servers and Servlets
- Servlets Basics
- Servlets life cycle
- Http Sen/let
- Http Servlet Request
- Request-Response Basics
- Session Tracking using Cookie and Http Session.

Java Server Pages

- Description
- JSP versus Server
- JSP
- Actions
- Directives
- Expressions
- Java Beans and JSP

Java Beans

- Introduction
- Types of Java Beans
- Bean Properties
- Bean Scopes

J2EE CONTENTS

- Web applications and HTTP basics
- Model 1 and Model 2 Architectures
- Interceptors
- Session Beans
- Message-Driven Beans
- Injectors
- Entities
- Entity Operations
- Java Persistence Query (JPQL)
- EJB Timer Service
- EJB Transactions
- EJB Security
- Struts Frame Work
- Web/application servers
- Netbeans & Eclips
    - Introduction
    - Project definition and configurations
    - Execution and debugging
    - Advance features Servlet

Servlet

- Servlet Basics
- Servlet API
- HTTP Basics
- Session Tracking
- Advance Servlet features (Forward, Include, Redirect)

JSP

- Introduction to JSP
- JSP Basics
- JSP Elements (Directives, Declaration, Scriplets, Actions)
- JSP and JavaBeans

- Custom Tag Library

Hibernate Syllabus
(Either hibernate or Spring Framework)

Basics of Hibernate

✓Hibernate Introduction
✓Hibernate Architecture
✓Understanding First Hibernate application

Hibernate with IDE

✓Hibernate in Eclipse
✓Hibernate in Netbeans

Hibernate Application

✓Hibernate with annotation
✓Hibernate Web application
✓Hibernate Generator classes
✓Hibernate Dialects

Hibernate Logging

✓Hibernate with Log4j 1
✓Hibernate with Log4j 2

Inheritance Mapping

✓Table Per Hierarchy
✓Table Per Hierarchy using Annotation
✓Table Per Concrete
✓Table Per Concrete using Annotation
✓Table Per Subclass
✓Table Per Subclass using Annotation

Collection Mapping

✓Mapping List
✓One-to-many by List using XML
✓Many to Many by List using XML

- ✓ One To Many by List using Annotation
- ✓ Mapping Bag
- ✓ One-to-many by Bag
- ✓ Mapping Set
- ✓ One-to-many by Set
- ✓ Mapping Map
- ✓ Many-to-many by Map
- ✓ Bidirectional
- ✓ Lazy Collection

Component Mapping

Association Mapping

- ✓ One-to-one using Primary Key
- ✓ One-to-one using Foreign Key

- Transaction Management
- HQL
- HCQL
- Named Query

Hibernate Caching

- ✓ First Level Cache
- ✓ Second Level Cache

- Integration

- ✓ Hibernate and Struts
- ✓ Hibernate and Spring

Spring Syllabus

(Either Spring or Hibernate Framework)

Basics of Spring

- ✓ What is Spring
- ✓ Spring Modules
- ✓ Spring Application
- Spring with IDE

- ✓ Spring in Myeclipse
- ✓ Spring in Eclipse
- IOC container

Dependency Injection

- ✓ Constructor Injection
- ✓ CI Dependent Object
- ✓ CI with collection
- ✓ CI with Map
- ✓ CI Inheriting Bean
- ✓ Setter Injection
- ✓ SI Dependent Object
- ✓ SI with Collection
- ✓ SI with Map
- ✓ CI vs SI
- ✓ Autowiring
- ✓ Factory Method

Spring AOP

- ✓ AOP Terminology
- ✓ AOP Implementations
- ✓ Pointcut
- ✓ Advices

Spring JDBC

- ✓ JdbcTemplate Example
- ✓ PreparedStatement
- ✓ ResultSetExtractor
- ✓ RowMapper

- ✓ NamedParameter
- ✓ SimpleJdbcTemplate

- Spring with ORM

  - ✓ Spring with Hibernate
  - ✓ Spring with JPA

SpEL

- ✓ SpEL Examples
- ✓ Operators in SpEL
- ✓ variable in SpEL

- Spring 3 MVC

Remoting with Spring

- ✓ Spring with RMI
- ✓ Http Invoker
- ✓ Hessian
- ✓ Burlap
- ✓ Spring with JMS

- OXM Frameworks

  - ✓ Spring with JAXB
  - ✓ Spring with Xstream
  - ✓ Spring with Castor

- Spring Java Mail

- Web Integration

  - ✓ Spring with Struts2
  - ✓ Login and Logout Application

JDBC

- JDBC Basics
- Advance JDBC

- Hibernate & JPA
- Hibernate Configuration
- Entity definitions and O/R Mapping
- Working with Persistent Objects
- Hibernate/JPA Queries
- Advance Concepts

EJB (Enterprise Java Beans)

- Stateless session beans
- Stateful session beans
- Entity beans
- Message Driven Beans
- EJB Architecture.