

**IDE:** Eclipse;      **JDK:** 1.7.0;

**Backend:** MYSQL

**Project:** Using SWINGS+MYSQL

## 1. Introduction To Java

## 2. Java Virtual Machine

- a. Java Platform
- b. Java Virtual Machine
  - i. Architecture of JVM
  - ii. Method Area
  - iii. Java Stack
  - iv. Heap Memory
  - v. Pc Register
  - vi. Native method Stack
- c. Java API
- d. Class Loader
  - i. Physical Loading
  - ii. Linking
  - iii. Initialization
- e. Types of Class Loader
  - i. Bootstrap Class Loader
  - ii. Extension Class Loader
  - iii. Application Class Loader
- f. Delegation Hierarchy
- g. Class File
- h. JDK

## 3. Operators and Casting

- a. Assignment Operator
- b. Type Casting

## 4. Control Flow Statement

- a. Selection Statement

## 5. Array

- a. Introduction
- b. Declaration of Array Variable
- c. Creating an array
- d. Consigning Declaration and Creation

- e. Array Initialize List
- f. Passing Array arguments
- g. Anonymous Array
- h. Multidimensional array
- i. Passing and returning multidimensional array
- j. Jagged array

## 6. Inheritance and interface

- a. Introduction
- b. Types of inheritance
- c. Methods of overriding
- d. Method of overloading
- e. Difference b/w Overriding and overloading
- f. Using keyword super
- g. Abstract method and class
- h. Difference b/w Abstract class Concrete Class
- i. Final Method
- j. Final Class
- k. Static Member
- l. Interface

## 7. Object and Variable

- a. Creation of an Object
- b. Types of Reference variable
- c. Passing object and returning Objects
- d. Array of Objects

## 8. Polymorphism

- a. Binding/Linking
- b. Overriding private methods
- c. Redeclaring files
- d. Overriding static methods
- e. Polymorphic Arguments and Return files
- f. Polymorphic method and constructor

**9. Garbage collection and object class**

- a. What is garbage collection?
- b. The universal Class (java.lang.object)
  - i. Colove ()
  - ii. Equals ()
  - iii. ToString()
  - iv. Getclass()
  - v. Finalize ()

**10.Exceptions**

- a. Introduction to Exceptions handling
- b. Runtime stack
- c. The stack trace
- d. Java.lang. throwable Class
- e. Java.lang.Exception Class
- f. Java.lang.error Class
- g. Finally Block
- h. Using try without getch
- i. Multiple catch block
- j. Using keyword throw
- k. Checked Exceptions
- l. Unchecked Exceptions
- m. Using throw clause
- n. User define Exceptions

**11.Packages**

- a. Introduction
- b. Name and unnamed Packages
- c. Java API Package Hierarchy
- d. Creating Packages
- e. Import Declarations
- f. Creating .Jar Files
- g. Accessibility Specifier
  - i. Public
  - ii. Protected
  - iii. Default
  - iv. Private

**12.Threads**

- a. Introduction
- b. Creating Threads
  - i. Java.lang. Threads Class
  - ii. Java.lang.runnable Interface
- c. Threads Priority
- d. Joining Threads
  - i. Synchronization
  - ii. Synchronize method
- e. Synchronized block
- f. Waiting and notifying
- g. Daemon Threads
- h. Deadlocks
- i. Life cycle of a thread
- j. Thread pool

**13.File and Streams**

- a. Java.io.file Class
- b. Introduction to Streams
- c. Byte Streams: Input and Output Streams
  - i. Java.io.input Stream class
  - ii. Java.io.output Stream class
  - iii. Java.io.Fileinput Stream class
  - iv. Java.io.Fileoutput Streamclass
  - v. Java.io.datainput Stream class
  - vi. Java.io.dataoutput Stream class
- d. Character Stream: readers And Writers
  - i. Java.io. reader class
  - ii. Java.io.writer class
  - iii. Java.io. filereader class
  - iv. Java.io. FileWriter class
  - v. Java.io. Printstream class
  - vi. Java.io. Bufferreader class
  - vii. Java.io.Bufferwriter class
  - viii. Java.io.RandomAccessFile Class
  - ix. Java,io.Scanner Class
- e. Object Input and Output Stream Class

## 14. Serialization

- a. Keyword Transient
- b. Inheritance in Serialization

## 15. Network/ Socket Programming

- a. Introduction
- b. Ports
- c. Java.net.Package
  - i. Java.net.InetAddress Class
  - ii. Java.net.Socket Class
  - iii. Java.net.ServerSocket Class
- d. Socket Programming and Serialization

## 16. JDBC (Java Database Connectivity)

- a. Introduction
- b. JDBC Driver
- c. Types of JDBC Drivers
- d. Versions of JDBC
- e. Writing- Reading Images (BLOB)
- f. JDBC API

## 17. The Collection Framework

- a. Introduction
- b. Retrieving Elements From Collection
  - i. Using for-each loop
  - ii. Using Iterator Interface
  - iii. Using List Iterator Interface
  - iv. Using Enumeration Interface
- c. Lists
  - i. ArrayList, LinkedList and Vector Classes
- d. Sets
  - i. HashSet and LinkedHashSet Classes
- e. Maps
  - i. Map Interface
  - ii. Hashmap, LinkedHashMap and HashTable Classes