CORE JAVA SYLLABUS:

1. Core Java Programming Introduction of Java
- Introduction to Java, Comparison with C and C++ and features of Java
- JVM Architecture
- Download and install JDK/JRE (Environment variables set up)
- The JDK Directory Structure
- First Java Program through command prompt
- First Java Program through Eclipse

2. Data types and Operators
- Primitive Data types, Declarations, Ranges
  - Variable Names Conventions
  - Numeric Literals, Character Literals
- Non-Primitive Data types
  - String Literals
  - Arrays(One dimensional; two- dimensional)
  - Array of Object References
  - Accessing arrays, manipulating arrays
- Enumerated Data Types
- Expressions in Java; introduction to various operators
  - Assignment Operator
  - Arithmetic Operators
  - Relational Operators
  - Logical Operators
  - Conditional Operators
  - Operator Precedence
- Defining a class, variable and method in Java
- Method Signature; method calls
- Implicit Type Conversions
- Upcasting and downcasting
- Strict typing
- Type conversion

3. Control Flow statements
- Statements and it's various categories in Java
- if, if-else, if-else-if
- switch case
- for statement (both flavours traditional and enhanced for)
- while and do-while loops
- The continue Statement; labelled continue statement
- The break Statement; labelled break statement
- return statement

4. OOPS and its application in Java
- Classes and Objects
- Defining a class;Defining instance variables and methods
- Creating objects out of a class
- Method calls via object references
- Abstraction
- Interfaces and Abstract classes
- Abstract and non-abstract methods
- Inheritance
- extends and implements keywords in Java
- Super class and Sub class
- this keyword, super keyword in Java for inheritance
- Concrete classes in Java

- ➢ Polymorphism
- ➢ Compile time polymorphism -- Overloading of methods
- ➢ Run time polymorphism -- Overriding of methods
- ➢ Method Overriding rules and method overloading rules
- ➢ Introduction to Object class and it's methods
- ➢ Encapsulation
- ➢ Protection of data
- ➢ Java Bean, POJO
- ➢ Getters/Setters
- ➢ Memory management in Java
- ➢ Heap
- ➢ Stack

5. Packages
- ➢ Need for packages
- ➢ What are packages; package declaration in Java
- ➢ Import statement in Java
- ➢ How do packages resolve name clashes?

6. Miscellaneous
- ➢ Var-Args
- ➢ Reference variables, local variables, instance variables
- ➢ Memory allocations to variables
- ➢ Double equals operator(==) operator for primitives and objects
- ➢ toString() method on an object

7. Statics
- ➢ Static variables and methods
- ➢ Static imports
- ➢ Static initialization blocks; instance intialization blocks
- ➢ Static concept in inheritance

8. Constructors
- ➢ What are Constructors?
- ➢ Properties of Constructors
- ➢ Default and Parameterized Constructors
- ➢ Rules for constructor implementation
- ➢ Constructor Chaining
- ➢ this call; super call for constructors
- ➢ Constructors for Enumerated Data Types
- ➢ Constructors concept for Abstract classes and interfaces

9. Exceptions in Java
- ➢ What are Exceptions?
- ➢ Need for exceptions
- ➢ How can Exceptions be coded in Java?
- ➢ API heirarchy for Exceptions
- ➢ Types of Exceptions
- ➢ Keywords in Exception API: try, catch, finally, throw, throws
- ➢ Rules for coding Exceptions
- ➢ Declaring Exceptions
- ➢ Defining and Throwing Exceptions
- ➢ Errors and Runtime Exceptions
- ➢ Custom Exception
- ➢ Assertions
- ➢ What are Assertions?
- ➢ Enabling and disabling assertions in development environment

10. Strings in Java

- ➢ What are Strings?
- ➢ String heap memory and Constant Pool memory
- ➢ Immutability in Strings
- ➢ String creation on heap and constant pool
- ➢ Method APIs on String; operations on Strings
- ➢ Mutability of String Objects - StringBuilder and StringBuffer
- ➢ Splitting of Strings and StringTokenizer class

11. Collection Framework in Java
- ➢ The Collections Framework
- ➢ The Set Interface
- ➢ Set Implementation Classes
- ➢ The List Interface
- ➢ List Implementation Classes
- ➢ The Map Interface
- ➢ Map Implementation Classes
- ➢ Queue Interface
- ➢ Queue Implmentation classes
- ➢ Utility classes
- ➢ Sorting collections using utility methods
- ➢ equals() and hashCode contract in Java collections
- ➢ overriding equals and hashCode methods in Java
- ➢ New Collections added in Java 1.6
- ➢ Primitive wrapper classes and all its method APIs

12. Generics
- ➢ Generics for Collections
- ➢ Generics for class
- ➢ Generics for methods

13. Input-Output in Java
- ➢ What is a stream?
- ➢ Overview of Streams
- ➢ Bytes vs. Characters
- ➢ Overview of the entire Java IO API
- ➢ Reading a file; writing to a file usinf various APIs
- ➢ Reading User input from console
- ➢ PrintWriter Class

14. Serialization
- ➢ Object Serialization
- ➢ Serializable Interface
- ➢ Serialization API
- ➢ ObjectInputStream and ObjectOutput
- ➢ Transient Fields
- ➢ readObject and writeObject

15. Inner Classes
- ➢ Inner Classes
- ➢ Member Classes
- ➢ Local Classes
- ➢ Anonymous Classes
- ➢ Static Nested Classes

16. Threads in Java
- ➢ Non-Threaded Applications
- ➢ Threaded Applications
- ➢ Process based multitasking Vs Thread based multitasking
- ➢ Thread API in Java

- ➢ Creating Threads
- ➢ States of a Thread
- ➢ Sychronization for threads; static and non-static synchronized methods; blocks; concept of object and class locks
- ➢ Coordination between threads - wait, notify and notifyAll methods for inter-thread communication

17. Applets
- ➢ What are applets?
- ➢ Need for Applets
- ➢ Different ways of running an applet program
- ➢ Applet API heirarchy
- ➢ Life Cycle of an applet
- ➢ Even Handlers for applets, mouse events, click events

18. Swing GUI
- ➢ Introduction to AWT
- ➢ Introduction to Swing GUI
- ➢ Advantages of Swing over AWT
- ➢ Swing API
- ➢ Swing GUI Components
- ➢ Event Handlers for Swing
- ➢ Sample Calculator application using Swing GUI and Swing Event handling

19. Access Modifers in Java
- ➢ What are access modifiers?
- ➢ Default
- ➢ Protected
- ➢ Private
- ➢ Public

20. Debugging of Java Programs in Eclipse.