|  |  |  |  |
| --- | --- | --- | --- |
| **Course Code** | **CSC406** | **Course Title** | **Programming in Core Java** |
| **Number of Credits** | **3 Credits** (TH)**2 Credits** (PR) | **Internal** | **20%** |
| **Total Contact Hours** | **3 HRS** (TH/Week)**4 HRS** (PR/Week) | **External** (Semester/Term Exam) | **80%** |

**Course Objective**:

The course provides students with the knowledge and skills needed to develop applications in Java for the Microsoft and Sun platform. The course focuses on fundamental concepts, designing user interfaces, program structure, and language syntax and implementation details. This is the first course in the Java Certification Exam and will serve as the entry point for other Advance Java Courses

**Prerequisite**:

Before attending this course, students must have:

• Student should know the Object oriented programming Concepts

– Or –

• Understanding of the basics of structured programming, including concepts such as flow control, variables and parameters, and function calls.

**At Course Completion**:

After completing this course, students will be able to:

• Create a simple Java Application based on the Java Foundations.

• Use of Java forms and controls to create a user interface.

• Create and use variables and arrays.

• Create and use Class, Interfaces, Packages and predefined utility classes.

• Implement decision structures and loops by using conditional expressions.

• Validate user input for fields, controls, and forms.

• Apply object‐oriented programming techniques to create classes, add methods, and add behavior to the classes

• Resolve syntax, run‐time, and logic errors by using the structured exception handling.

• Enhance the user interface by adding menus, status bars, and toolbars.

**Course Outline**

**Unit-1:**

**A. Java Introduction:** The Java Environment – Overview, Writing a Java Program, Obtaining The Java Environment, Setting up your Java Environment, Creating a Class That Can Run as a Program, The main() Method, Useful Stuff Necessary to go Further, System.out.println(), Using the Java Documentation. **B. Java Basics:** Basic Java Syntax: General Syntax Rules, Java Statements, Blocks of Code, Comments, Variables: Data types, Primitive Data Types, Object Data Types, Literal Values, Constants and the final keyword, Mathematics in Java: Expressions, Operator Precedence, Multiple Assignments, Order of Evaluation, Bitwise operators, Compound Operators, Expressions that Mix Data Types: Typecasting Creating and Using Methods, Creating Methods, Variable Scope.

**Unit-2:**

**A. Java Objects**: Objects: Object‐Oriented Languages, Object‐Oriented Programs, Encapsulation, Creating and Using an Instance of an Object, References Defining a Class, Constructors, Method Overloading, The this Keyword, static Elements, Garbage Collection, Java Packages, Dealing with Keyboard Input, String, StringBuffer, and StringBuilder, Creating Documentation, Comments and Using javadoc, Javadoc Comments. **B. Comparisons And Flow Control Structures**: Controlling Program Flow: Boolean‐Valued Expressions, Complex Boolean Expressions, Simple Branching, Two Mutually Exclusive Branches, Nested if... else Statements ‐ Comparing a Number of Mutually Exclusive Options, Comparing a Number of Mutually Exclusive Options ‐ The switch Statement, Comparing Objects, Conditional Expression, while and do. . .while Loops, for Loops, Additional Loop Control: break and continue, Breaking Out of a Loop, Continuing a Loop, Classpath, Code Libraries, and JAR files, Using CLASSPATH Creating a jar File (a Library) C. Arrays and Vectors: Arrays : Defining and Declaring Arrays, Instantiating Arrays, Initializing Arrays, Working With Arrays, Array Variables, Copying Arrays, Arrays of Objects, Enhanced for Loops ‐ the For‐Each Loop, Multi‐Dimensional Arrays, Multidimensional Arrays in Memory, Example ‐ Printing a Picture, Typecasting with Arrays of Primitives, Using Vectors: Defining Vectors and using Vectors.

**Unit-3:**

**A. Inheritance:** Inheritance: Derived Class Objects, Polymorphism, Inheritance and References Dynamic Method Invocation, Creating a Derived Class, Inheritance and Access Inheritance and Constructors ‐ the super Keyword, Derived Class Methods That Override Base Class Methods Inheritance and Default Base Class Constructors, The Instantiation Process at Runtime, Typecasting with Object References: Typecasting, Polymorphism, and Dynamic Method Invocation, More on Overriding, Object Typecasting Example, Checking an Object's Type: Using instanceof, Typecasting with Arrays of Objects, Other Inheritance‐Related Keywords: abstract, final, Methods Inherited from Object. **B. Packages and Interfaces:** Interfaces: Creating an Interface Definition, Implementing Interfaces: Implementing Interfaces – Example, Reference Variables and Interfaces, Calling an Interface Method, Interfaces and Inheritance: Some Uses for Interfaces, Interfaces and Event‐Handling Interfaces and "Pluggable Components", Packages: Creating and using packages, Access. **C. Inner Classes:** Inner Classes, Nested Classes, Inner Class Syntax, Instantiating an Inner Class Instance from Within the Enclosing Class, Inner Classes Referenced from Outside the Enclosing Class Working with Inner Classes. **D. Exceptions:** Exceptions: Handling Exceptions, Exception Objects: Attempting Risky Code ‐ try and catch, Guaranteeing Execution of Code ‐ the finally Block, Letting an Exception be Thrown to the Method Caller, Throwing an Exception, Exceptions and Inheritance, Exception Class, Constructors and Methods, Creating and Using Your Own Exception Classes, Rethrowing Exceptions, Initializer Blocks, Static Initializer Blocks, Assertions.

**Unit-4:**

**A. Utility Classes**: Collection Interfaces, Concrete collections, Collections framework, Algorithms, Legacy Collections Streams: Output Streams, Input Streams, Filter Streams, Readers and Writers. **B. Threads**: Thread Class and Runnable Interface, Thread Synchronization **C. I/O** **Package**: InputStream and OutputStream classes, Reader and Writer classes **D. Java Networking:** InetAddress, URL, URLConnection, TCP/IP Server Socket, Client Socket, User Datagram Sockets**, D. Applet and Swings**: Applet: Applet Life Cycle, Passing Parameters to Applet, Delegation Event Model, AWT Components, AWT Events, using listeners, Working with Graphics, Loading Image and Multimedia objects in applet

**Unit-5:** **Java Database Connectivity**: Java Database Connectivity Architecture, JDBC-ODBC Bridge, JDBC Drivers, JDBC API, JDBC classes, Driver Interface, DriverManager Class, Connection, Statement, ResultSet, Implementing Stored Procedures.

**Reference Books**:

* Java 2 Complete Reference by Herbert Schieldt (Sixth Edition)
* Core Java Vol 1: Sun Press, Eighth Edition
* Core Java Vol 2: Sun Press

**E-books**:

* Java 2 Complete Reference by Herbert Schieldt (Fourth Edition)

**Lab Exercise for CSC454, Practical based on CSC406: There should be minimum 10 lab assignment on the topics discussed in the course.**