

ADVANCE PROBLEM SOLVING USING JAVA PROGRAMMING

Course Number	: CSCC31/CBCS31
Credit Hours	: 3-1-4
Course Prerequisite	: CSCC11/CBCS11/CSCC12/CBCS12/CSCC21/CSCC22
Course Status	: CBCS for PG
Instructor's name	: Dr. Khalid Razz
Tel. No.	: +91-11-26980014 Extn. 30
Office Location	: Department of Computer Science
Office Hours	: 11:00 – 01:00
Class Location	: SB-002/003
Class Time	: 2:50 – 4.30 (Mon-Fri)

a) Course Description: Java is a general-purpose, object-oriented, powerful and widely used programming language that has been equipped with many modern programming features including class-based, object-oriented, structured, imperative, generic, reflective, concurrent, automatic garbage collected, and architecture neutral. As of 2016, Java has been ranked as mostly used programming language, especially for web-based application, with 9 million developers. The main principles on which Java language is created, are: simple, object-oriented, familiar, robust and secure, architecture-neutral and portable, high performance, interpreted, threaded, and dynamic.

b) Objectives: Subject-specific skills: By the end of this course, the student must be able to:

1. Understand fundamentals of Java programming constructs.
2. Design classes, objects and apply code reusability using inheritance.
3. Handle String by using powerful String classes such as StringBuffer, StringBuilder and StringTokenizer.
4. Write generic, multithreaded, GUI-based programming.
5. Handle exceptions and write programs that needs database handling.

UNIT-WISE SYLLABUS

- 1. Introduction, Environment and Programming Structure:** Java White Paper Buzzwords, History of Java, Common Misconceptions, Choosing a Development Environment: Command-Line Tools, Running a Graphical Application, Building and Running Applets; A Simple Java Program, Comments, Data Types, Variables, Operators, Input and Output, Control Flow, Big Numbers, Arrays.
- 2. Class, Objects and Inheritance:** Introduction to OOP, Predefined Classes, User Defined Classes, Static Fields and Methods, Method Parameters, Object Construction, Packages, Class Path, Documentation Comments, Class Design; Inheritance: Super-classes and Subclasses, Types of Inheritance, Polymorphism, Abstract class, Object: The Cosmic Super class, Generic Array Lists, Object Wrappers and Autoboxing, Methods with a Variable Number of Parameters, Enumeration Classes, Reflection, Inheritance Guidelines, Interfaces.
- 3. String Handling, Exception Handling and Generic Programming:** String Handling APIs: String, Immutable String, Methods of String Class, StringBuffer, StringBuilder, StringTokenizer. Exceptions: Dealing with Errors, Catching Exceptions, Guidelines for Using Exceptions, Assertions, Logging; Generic Programming: Definition, Generic Methods, Bounds for Type Variables, Generic Code and VM, Restrictions and Limitations, Inheritance Rules for Generic Types, Reflection and Generics.
- 4. Java Collections and Multithreading:** Collection Interfaces, Concrete Collections, The Collections Framework, Algorithms, Legacy Collections, Multithreading: Threads, Interrupting Threads, Thread States, Thread Properties, Synchronization, Blocking Queues, Thread-Safe Collections, Callable and Futures, Executors, Synchronizers.
- 5. Java GUI Programming and JDBC:** Introduction to Swing, Creating a Frame, Positioning a Frame, Displaying Information in a Component, Displaying Images, Event Handling, Basics of Event Handling, Actions, Mouse Events, The AWT Event Hierarchy; JDBC: Basic JDBC Programming Concepts, JDBC Drivers, Statements, Executing Queries, Result Sets.

Text Books:

1. Horstmann & Cornell: **Core Java Volume I: Fundamentals**, Pearson Education
2. Horstmann & Cornell: **Core Java Volume II: Advanced Features**, Pearson Education.

References Books

1. H. Schildt: **Java 2: The Complete Reference** (9th Ed.), 2014, Tata McGraw Hill.
2. Dietel & Dietel: **Java How To Program**, Pearson Education.
3. Bruce Eckel: **Thinking in Java**, Pearson Education, 2006
4. Balagurusamy: **Programming with Java: A Primer** (5th Ed.), 2014, Tata McGraw Hill.

c) Outline

Week	Topics
Week 1:	Java White Paper Buzzwords, History of Java, Common Misconceptions, Choosing a Development Environment: Command-Line Tools, Running a Graphical Application.
Week 2:	Building and Running Applets; A Simple Java Program, Comments, Data Types, Variables, Operators, Input and Output, Control Flow, Big Numbers, Arrays.
Week 3:	Introduction to OOP, Predefined Classes, User Defined Classes, Static Fields and Methods, Method Parameters, Object Construction.
Week 4:	Packages, Class Path, Documentation Comments, Class Design; Inheritance: Super-classes and Subclasses, Types of Inheritance, Polymorphism.
Week 5:	Abstract class, Object: The Cosmic Super class, Generic Array Lists, Object Wrappers and Autoboxing, Methods with a Variable Number of Parameters, Enumeration Classes, Reflection, Inheritance Guidelines, Interfaces.
Week 6:	String Handling APIs: String, Immutable String, Methods of String Class, StringBuffer, StringBuilder, StringTokenizer.
Week 7:	Exceptions: Dealing with Errors, Catching Exceptions, Guidelines for Using Exceptions, Assertions, Logging.
Week 8:	Generic Programming: Definition, Generic Methods, Bounds for Type Variables, Generic Code and VM, Restrictions and Limitations, Inheritance Rules for Generic Types, Reflection and Generics.
Week 9:	Collection Interfaces, Concrete Collections, The Collections Framework, Algorithms, Legacy Collections.
Week 10:	Multithreading: Threads, Interrupting Threads, Thread States, Thread Properties, Synchronization, Blocking Queues.
Week 11:	Thread-Safe Collections, Callable and Futures, Executors, Synchronizers.
Week 12:	Introduction to Swing, Creating a Frame, Positioning a Frame, Displaying Information in a Component, Displaying Images.
Week 13:	Event Handling, Basics of Event Handling, Actions, Mouse Events, The AWT Event Hierarchy.
Week 14:	JDBC: Basic JDBC Programming Concepts, JDBC Drivers, Statements, Executing Queries, Result Sets.

d) Application Pedagogy

The assessment and evaluation process will be broadly classified with the following TWO components, viz.,:

1. In-Sem Continuous Internal Assessments and Evaluations, and
2. End-Sem Final Examination

The weightage of Internal Assessments for Theory Course will be 25% and for Laboratory / practical will be 50%. However, the remaining 75% weightage for Theory Courses and 50% for Laboratory Course will be for End-Sem Final Examination of TWO hours durations for Theory and FOUR hours durations for Practical.

In-Sem Continuous Internal Evaluations:

The In-Sem Internal evaluations may be further divided into two components, viz., Two Sessional tests for 15% weightage and the remaining 10% weightage will be based on any one or combination of the various modes of evaluation mentioned below. In case of Sessional tests, Best out of TWO will be awarded for those who have appeared in both the test. NO excuse will be accepted for missing a Sessional test, unless and until the reason is Bonafide. Else only 50% of the marks obtained will be awarded for appearing in one test during the Semester.

Various Modes of Evaluations (10%):

The concerned Instructor may have the choice to select any one or combinations of the following modes of Evaluation such as, Quizzes, Assignments, Seminar presentation using ppt, etc., as per the General Guidelines mentioned below.

Quiz Schedule:

Quizzes, if any, will be MCQs usually during the class as surprise tests, the frequencies of which is left with the Instructor to be decided on the basis of nature of Course.

Assignments, if any::

There can be weightage and deadlines for Assignments to be submitted by each student. The assignment could be as such to test and evaluate the understanding of Concepts, Programming and scripting, besides diagrams etc.

Seminar presentation using ppt / multimedia, if needed:

General Guidelines:

1. No excuse are to be entertained for late submission for Quiz / Assignment/ Seminar Presentation, etc., except with notification from the Head of the Department.
2. Students are expected to work individually, on all modes of evaluations, unless and until specifically assigned in Groups.
3. Attendance and Assessments are continuous, regular and mandatory, however, Extensions may be granted only by the course instructor in consultation with Head of the Department. Extensions may be given on serious medical or compassionate grounds, or any other Bonafide reasons, with supporting documents, on the basis of formal written request to be made in advance to the Head of the Department.
4. Cheating of any sort / form / types including plagiarism is strictly prohibited may zero marks with additional penalty to debar in the next assessment mode at the discretion of Instructor Concerned.
5. Grading System and Mark sheet will be issued as per JMI rules.
