FACULTY OF COMPUTER APPLICATIONS
Bachelor of Science (Information Technology)
B.Sc. (IT)

- **Sem.** :3
- **Subject Code** :05BS0302
- **Subject** : Programming Techniques-3(Java)
- **Course Objectives** :
    1. To understand concept of Object Oriented Programming.
    2. To understand class fundamentals.
    3. To describe Inheritance and Interfaces.
    4. To work with Package, Exception Handling and Multithreading.
    5. To compare different collections and understand file handling concepts.

- **Prerequisite**: Knowledge of C or C++ programming languages

Unit No	Topics Covered	No of lectures required
1	**Introduction to Java :** Java Fundamentals: The Origins of Java, The Java Buzzwords, OOP Concepts, JDK, First Simple Program, Handling Syntax Errors, The Java Keywords, Identifiers in Java, The Java Class Libraries **Introducing Data Types and Operators:** Why Data Types Are Important?, Java's Primitive Types(Integers, Floating-Point Types, Characters), The Boolean Type,Literals (Hexadecimal, Octal andBinary Literals, Character Escape Sequences, String Literals), A Closer Look at Variables(Initializing a Variable, Dynamic Initialization), The Scope and Lifetime of Variables, Operators, Arithmetic Operators (Increment and Decrement), Relational and Logical Operators, Short-Circuit Logical Operators, The Assignment Operator, Shorthand Assignments, Type Conversion in Assignments,	**10**

	Casting Incompatible Types, Operator Precedence, Expressions (Type Conversion in Expressions, Spacing and Parentheses) **Program Control Statements:** Input Characters from the Keyboard,The if Statement, Nested ifs, The if-else-if Ladder, The switchStatement,Nested switch Statements, The for Loop, Some Variations on the for Loop, Missing Pieces (The Infinite Loop), Loops with No Body, Declaring Loop Control Variables Inside the for Loop, The Enhanced for Loop, The while Loop, The dowhile Loop, Use break to Exit a Loop,Use break as a Form of goto, Use continue, Nested Loops	
2	**Class Fundamentals :** Introducing Classes, Objects, and Methods: Class Fundamentals(The General Form of a Class, Defining a Class), How Objects Are Created, Reference Variables and Assignment, Methods (Adding a Method to the Box Class), Returning from a Method, Returning a Value, Methods with Parameters,Constructors, Parameterized Constructors, The this Keyword,Instance Variable, Garbage Collection **A Closer Look at Methods and Classes:** Controlling Access to Class Members (Java's Access Modifiers), Pass Objects to Methods (How Arguments Are Passed), Returning Objects, MethodOverloading, Overloading Constructors, Recursion, Understanding static (Static Blocks), Introducing Nested and Inner Classes, Varargs: Variable-Length Arguments (Varargs Basics, Overloading Varargs Methods, Varargs and Ambiguity)	**10**
3	**Inheritance:** Inheritance Basics, Member Access and Inheritance, Constructors and Inheritance, Using super to Call Superclass Constructors, Using super to Access Superclass Members, Creating a Multilevel Hierarchy, When Are Constructors Executed?, Superclass References and Subclass Objects, Method Overriding, Overridden Methods Support Polymorphism, Why Overridden Methods?, Using Abstract Classes, Using final, (final Prevents Overriding, final Prevents Inheritance, Using final with Data	**10**

	Members), The Object Class **Interfaces:** Interfaces, Implementing Interfaces, Using Interface References, Variables in Interfaces, Interfaces Can Be Extended, Default Interface Methods (Default Method Fundamentals, A More Practical Example of a Default Method, Multiple Inheritance Issues), Use static Methods in an Interface, Private Interface Methods **Autoboxing, java.lang package :** Autoboxing, Type Wrappers, Autoboxing Fundamentals, Autoboxing and Methods, Autoboxing/Unboxing Occurs in Expressions, Java.lang package (String, String Buffer, Comparable interface)	
4	**Packages:** Packages (Defining a Package, Finding Packages and CLASSPATH, A Short Package Example), Packages and Member Access (A Package Access Example),Understanding Protected Members, Importing Packages, Java's Class Library Is Contained in Packages **Exception Handling:** The Exception Hierarchy, Exception Handling Fundamentals (Using try and catch, A Simple Exception Example), The Consequences of an Uncaught Exception (Exceptions Enable You to Handle Errors Gracefully), Using Multiple catch Statements, Catching Subclass Exceptions, Try Blocks Can Be Nested, Throwing an Exception (Rethrowing an Exception), A Closer Look at Throwable, Using finally,Using throws, Three Additional Exception Features, Java's Built-in Exceptions, Creating Exception Subclasses **Multithreaded Programming:** Multithreading Fundamentals, The Thread Class and Runnable Interface, Creating a Thread, (One Improvement and Two Simple Variations), Creating Multiple Threads, Determining When a Thread Ends, Thread Priorities, Synchronization, Using Synchronized Methods, The synchronized Statement	**12**
5	**The collection Framework:**	**08**

	Introduction, Collection framework (Collection interface, list interface, set interface, sorted set interface), The collection class, Array list and Link list classes (maintaining the capacity and the link list class), iterating elements of collection (the list iterator interface), hash set and tree set Classes **Using I/O:** Java's I/O Is Built upon Streams, Byte Streams and Character Streams, The Byte Stream Classes, The Character Stream Classes, The Predefined Streams, Using the Byte Streams (Reading Console Input, Writing Console Output), Reading and Writing Files Using Byte Streams (Inputting from a File, Writing to a File),Automatically Closing a File, Reading and Writing Binary Data	

Course Outcomes : (Students will able to)

1. Define the basic concepts of Object Oriented Programming and Java Programming.
2. Describe fundamentals of class, variable and methods.
3. Demonstrate the concept of Inheritance, Interface and java.lang packages.
4. Construct own packages and exceptions.
5. Compare and contrast different collections and use file handling concepts in Java.

Course Outcomes – Program Outcomes Mapping Table :

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8
CO1	H	H	L		L		H	L
CO2	M	H	M		L		H	L
CO3	L	H	M		L		H	L
CO4	L	H	M		L		H	L
CO5	L	H	M		L		H	L

Main Reference :

1. Java – The Complete Reference, Herbert Schildt, Oracle Press, Eleventh Edition.

FACULTY OF COMPUTER APPLICATIONS
Bachelor of Science (Information Technology)
B.Sc. (IT)

Other References :

1. Programming with Java, E Balagurusamy, McGraw Hill Education, Sixth Edition.
2. OCP Java SE 8: Programmer II Exam Guide, Kathy Sierra, Bert Bates, Elisabeth Robson, Oracle Press, Indian Edition.
3. Learn to Program with Java JDK 15.0, Smiley John, John Smiley Publishing, First Edition.

Web References :

1. https://www.w3schools.com/java/default.asp
2. https://www.javatpoint.com/java-tutorial

App References :

1. Learn Java : Ultimate Guide – coding and programming
2. Learn Java :Sololearn

Syllabus Coverage from text /reference book & web/app reference:

Unit #	Chapter Numbers
1	1,2,3,4,5
2	6,7
3	8, 9 (Interface), 12,18 (mentioned java.lang packages)
4	9 (Package), 10, 11
5	19, 21

FACULTY OF COMPUTER APPLICATIONS
Bachelor of Science (Information Technology)
B.Sc. (IT)

PRACTICALS

Unit No	List of Practicals
1	1. Write a Java program to display "Hello World" message. 2. Write a Java Program to enter two nos. and perform all 5 mathematical operations 3. Write a Java Program to enter Principle Amt., Rate of Interest and No. of Years and find Simple Interest. 4. Write a java program to get personal information from user and display on screen. 5. Write a Java Program to enter marks of 3 subjects and find total, percentage, result and class. 6. Write a Java program to enter two nos. and find maximum out of it. 7. Write a Java Program to enter three nos. and find maximum out of it. 8. Write a Java Program to enter 10 Nos. and find sum and average of it. 9. Write a java program to get the name from user and print 10 times using loop. 10. Write a java program to enter a number and find out sum of digits.
2	1. Write a Java program to demonstrate use of Class and Methods. 2. Write a Java program to demonstrate use of pass objects to methods. 3. Write a Java program to demonstrate use of constructor. 4. Write a Java program to demonstrate use of parameterised constructor. 5. Write a Java program to demonstrate use of this keyword. 6. Write a Java program to demonstrate use of Method Overloading. 7. Write a Java program to demonstrate use of Overloading of Constructor. 8. Write a Java program to demonstrate use of static block. 9. Write a Java program to demonstrate use of nested classes. 10. Write a Java program to demonstrate use of varargs.
3	1. Write a Java program to perform simple inheritance. 2. Write a Java program to demonstrate use of multilevel inheritance. 3. Write a Java program to demonstrate use of Hierarchical inheritance 4. Write a Java program to demonstrate use of method overriding. 5. Write a Java program to demonstrate use of super keyword 6. Write a Java program to demonstrate use of final keyword with class and method. 7. Write a Java program to demonstrate use of abstract class and abstract method

	8. Write a Java program to demonstrate use of interface 9. Write a Java program to demonstrate use of Multiple inheritance using interface. 10. Write a Java program to extend one interface into another interface
4	1. Write a Java program to implement simple exception handling 2. Write a Java program to implement Arithmetic Exception 3. Write a Java program to use Multiple Catch Block 4. Write a Java program to use Throw Keyword 5. Write a Java program to use Throws Keyword 6. Write a Java program to use finally Keyword 7. Write a Java program to create a thread using Thread Class 8. Write a Java program to create a thread using Runnable. 9. Write a Java program to set Thread name and priority & test it. 10. Write a Java program to create two threads and make them Synchronized (Thread Safe)
5	1. Write a Java program to demonstrate use of Array list. 2. Write a Java program to demonstrate use of Link List. 3. Write a Java program to add Book IDs and Book Names (Pairs) using a HashSet. 4. Write a Java program to store different mapped values (Key-Value) using a TreeMap Class 5. Write a Java program to add multiple elements using a SortedSet of collection. 6. Write a Java program to demonstrate use of iterator interface. 7. Write a Java program to write a simple message into a file using a FileOutputStream 8. Write a Java program to read a message (data) from a file by using FileInputStream. 9. Write a Java program to write a data into a file by using a FileWriter class of IO 10. Write a Java program to read a data from a file by using a FileReader class.