C. Syllabus Detailing and Learning objectives

Module	Chapter	Detailed Content	Syllabus Detailing	Learning Objectives
Module 1	**CH 1** Introduction to Object Oriented Programming (2-Hours)	1.1 OOP Concepts: Object, Class, Encapsulation, Abstraction, Inheritance, Polymorphism. 1.2 Features of Java, JVM 1.3 Basic Constructs /Notions: Constants, variables and data types, Operators and Expressions, Revision of Branching and looping	**Purpose**: To make students understand basic concepts of object oriented programming and the features of java and JVM. Also to make students understand the basic constructs/notions likes Constants, variables and data types, Operators and Expressions and do the revision of Branching and looping	1. To describe the features of java and JVM.**(R)** 2.To explain the fundamentals concepts of object oriented programming.**(U)** 3. To Differentiate Between various branching and looping.**(AN)**
			Scope – **1. Academic Aspects-** Understanding basic concepts of OOPS, construct/notation, branching and looping. **2. Technology Aspect-** Understand basics of JAVA and JVM. **3. Application Aspect-** Student should understand how to use various operators in programs.	
			Students Evaluation – 1. Theory Questions to be asked on Object, Class, Encapsulation, Abstraction, Inheritance, and Polymorphism. 2. Implementation of different operators and loops can be evaluated in lab.	
	CH 2 Classes, Object and Packages (5-Hours)	2.1 Class, Object, Method. 2.2 Constructor, Static members and methods 2.3 Passing and returning Objects 2.4 Method overloading	**Purpose** – This chapter gives detailed insight of classes, objects, methods, constructors, static keyword. Also focused on passing and returning objects, method overloading. It also gives a brief idea about different types of packages and access specifiers.	1. To explain class, objects and methods. **(A)** 2. To differentiate various access specifiers. **[AN]** 3. Explain different types of packages.**[U]**

		2.5 Packages in java, creating user defined packages, access specifiers.	**Scope –** **1. Academic Aspects-** Learning the insights of class, object, methods and constructors. **2. Technology Aspect-**Implement a program for passing and returning the object. **3. Application Aspect-** Application of user defined and system defined packages in a program.	4. To design a programs by passing and returning the object.**(C)** 5. Explain method overloading. **(E)**
			Students Evaluation 1. Questions on access specifiers and static members and methods can be asked. 2. Implementation of Method overloading can be evaluated in lab. 3. Students can use access specifiers in programs.	
Module 3	**Chapter 3** Array, String and Vector (Hours -4)	3.1 Arrays, Strings, String Buffer 3.2 Wrapper classes, Vector	**Purpose-** This chapter is focused on the details of the arrays, strings, string buffers. Also this chapter gives the idea about the wrapper classes and vectors.	1. Explain the concept of array. **(U)** 2. Differentiate string and string buffer concept. **(AN)** 3. Compare array and vector to determine the advantages of vector over array.**(AN)** 4. Explain the Wrapper classes. **(E)**
			Scope – **1. Academic Aspects-** Understanding concept of array, string and string buffer. **2. Technology Aspect-** implement a program using wrapper classes. **3. Application Aspect-** Students should understand where to use array and vectors in programs.	

			Students Evaluation – 1. Theory Questions to be asked on wrapper classes and array. 2. Lab experiments for String and Stringbuffer. 3. Corresponding viva questions can be asked for vectors and arrays.	
	Chapter 4 Inheritance and Interface (Hours -3)	4.1Types of Inheritance, super keyword, Method Overriding, abstract class and abstract method, final keyword 4.2 Implementing interfaces, extending interfaces	**Purpose-** This chapter gives the overview of inheritance and its types, Method Overriding, abstract class and abstract method, final keyword like concepts. Also give the detailed idea about the interfaces.	1. Describe concept of inheritance and also differentiate its various types. **(R)** 2. Explain the concept of method overloading and show how it is implemented practically. **(A)** **3.** Explain abstract method, super and final keywords. **(E)** **4.** Relate the interface with inheritance. **(A)**
			Scope - **1. Academic Aspects-** Understanding the inheritance and interface. **2. Technology Aspect-** Implement a program for method overriding. **3. Application Aspect-** Students should understand how interface can be used in program.	
			Students Evaluation – 1. Theory Questions to be asked on abstract class and abstract method. 2. Lab experiments for implementation interfaces, extending interfaces. 3. Corresponding viva questions can be asked for concepts super keyword, Method Overriding, abstract class and abstract	

			method, final keyword.	
Module 5	**Chapter 5** - Exception Handling and Multithreading (Hours -4)	5.1 Error vs Exception, try, catch, finally, throw, throws, creating own exception. 5.2 Thread lifecycle, Thread class methods, creating threads, Synchronization	**Purpose –** To make students understand basics about Error vs Exception. Also to make students understand concept of Thread.	1. To describe the fundamentals and technological aspects of Error and Exception. **(R)** 2. To list and explain different types of Exception. **(U)** 3. To draw and explain the lifecycle **(R)** 4. Discuss the need for Synchronization.**(U)**
			Scope – **1. Academic Aspects-** Understanding try, catch, finally, throw and throws keywords. **2. Technology Aspect-** Understand Thread lifecycle, synchronization. **3. Application Aspect-** Implement a program for Thread class methods.	
			Student Evaluation - 1. Theory Questions to be asked on synchronization. 2. Lab experiments for implementation of synchronization.	
Module 6	Chapter 6 - GUI programming in JAVA (Hours -8)	6.1 Applet: Applet life cycle, Creating applets, Graphics class methods, Font and Color class, parameter passing. 6.2 Event Handling: Event classes and	**Purpose** ~~0~~8 To make students understand basics of Applet life cycle, Event Handling and AWT.	1. To describe and identify use of Applets. **(AN)** 2 Describe Event classes and event listener, list the types **(R)** 3. List the AWT controls. **(R)**
			Scope – **1. Academic Aspects-** Understanding Applet, AWT and Event Handling. **2. Application Aspect-** Event classes and event listener and their use in program.	

		event listener 6.3 Introduction to AWT: Working with windows, Using AWT controls- push Buttons, Text Fields, Text Area, Box, and Radio Buttons. 6.4 Programming using JDBC: Introduction to JDBC, JDBC Drivers & Architecture.	**Student Evaluation -** 1. Theory Questions to be asked JDBC, JDBC Drivers & Architecture. 2. Explanation AWT Label, Check	4.Explain various stages of Applet life cycle.**[U]** 5. Define JDBC Drivers & Architecture.**(AN)**