

Class	F.Y.B.C.A.
Semester	II
Course Name	Core Java
Course Code	PUSCA201
Type of course Discipline Specific	Core
Level of the Subject	Medium
Credit Points	3 Theory+1 Practical

### Course Objectives:

- 1.Understand and apply core object-oriented principles like Encapsulation, Inheritance, Polymorphism, and Abstraction
2. Recognize the need for exception handling in software development to improve program reliability and user experience.

Unit No.	Name of Unit	Topic No.	Contents	Hours
Unit-1	Introduction to Object-Oriented Programming and Core Java Concepts	1.1	Introduction to OOPS concepts using read world problem. What is Object Oriented? Concepts of OOPS: Objects, Classes, Data Abstraction and Data Encapsulation, Inheritance, Polymorphism,	15

<b>Unit-2</b>	Java Programming: Classes, Objects, Constructors, and Exception Handling	<b>2.1</b>	Classes & Object: Defining a class, Instantiating Objects from a class, methods, accessing a method, method returning a value, method's arguments, variable arguments [Var args], static field and static methods. Constructor: uses of constructor, types of Constructor ( default, parameterized, copy), constructor overloading, this keyword.	15
		<b>2.2</b>	Exception Handling: What is Exception in Java? What is Exception Handling? Hierarchy of Java Exception classes, Types of Java Exceptions, Java Exception Keywords (try, catch, finally and throw).	
		<b>2.3</b>	Collection Framework: Introduction, util Package interfaces, List interface and its classes, ArrayList, Vectors.	
<b>Unit-3</b>	Advanced Java Concepts: Packages, Inheritance, Polymorphism, and AI-Driven Code Optimization	<b>3.1</b>	Package: what is package , Creating Packages, Default Package, Importing Package. Access specifier(public, private,protected,default)	15
		<b>3.2</b>	Inheritance & polymorphism: Inheritance and Access Control, Types of inheritance. Method overloading, Method Overriding, super keyword	
		<b>3.3</b>	Abstract Classes: Abstract Classes, Abstract methods, How is an Interface different from an Abstract class? Interfaces: Interfaces, What is an Interface? Multiple Inheritance, functional interface, Lambda Expressions. Optimize and improve the java code with AI-driven solutions(use Microsoft Copilot, ChatGPT,Jupyter Notebook Extensions)	
<b>Total Hours</b>				45

### Course Outcomes:

1. Develop expertise in writing and understanding Java code that applies fundamental programming principles effectively.
2. Explain different types of exceptions, ensuring that programs handle errors smoothly without abrupt failures.
3. Analyze how to solve complex problems by combining multiple control structures and loops for decision-making and repetition.
4. Illustrate the role of constructors in initializing object state.
5. Extend base classes to create hierarchical relationships and reuse code effectively.

6.Understand the concept of Java packages, create and import packages, and apply access specifiers to manage the visibility and organization of code effectively.

**Reference:**

- 1."Core Java" by E. Balaguruswamy
- 2.Java :The Complete Reference -Herbert Schildt, Publisher -Tata McGraw Hill ,9th Edition.
- 3.Murach's Beginning Java with Netbeans -Joel Murach , Michael Urban,Publisher- SPD,1st Edition
- 4.Core Java, Volume I:Fundamentals-Horstmann,Publisher- Pearson ,9th Edition.
- 5.<https://www.geeksforgeeks.org/>
- 6."Java: A Beginner's Guide" by Herbert Schildt

**Case Study**

1	RBI bank declares to give interest to all the bank's customers. Now each nationalized bank will give different rates of interest to its customers. Nationalized bank are SBI,PNB,BOB,IOB.RBI wants to see the rate of interest given by All the banks.RBI wants the best suitable plan to achieve this task with redundancy of data and data should be secure
2	A-Z company is a perfect service provider for all the IT and software development requirements. The team leverages their knowledge in completing the projected goals and makes it possible to build the ideas as per the market demands. Apart from focusing on just making a website, the team of experts always focus on designing a platform that can take the client's business to another level.They always blend expertise, innovation & technology to design innovative masterpieces. Client satisfaction is what matters the most for the people at the A-Z company. The company is going to develop a Java project on the Student Management System

## PRACTICALS

Practical No	Details
1	<p>a)Write a Java program that takes a number as input and prints its multiplication table up to 10.  b)Write a Java program to print the area and perimeter of a circle (use of constants).</p>
2	<p>Write a program to perform following tasks :  a)<u>Factorial</u> b)Armstrong c)Prime Number d) Palindrome  e)Create a program to print the day of the week based on a number (1 to 7)</p>
3	<p>Write a Java program to perform</p> <p>a)write a program to find the maximum and minimum of n elements.  b)Create a program that searches for a given element in an array of size n.  c)Write a program to add and multiply two matrices represented by 2D arrays.</p>
4	<p>a)Write a Java program to implement methods with variable length arguments  b)Create a class Person where this keyword is used to differentiate between class variables and method parameters.  c)Create a Counter class with a static field to count the number of objects created. Implement a static method to get the count.</p>
5	Write a Java program to work with constructors, constructor overloading.
6	Write a Java program to implement exception handling by using try...catch & finally block.
7	Create a package called com.myapp and include a class Employee inside it. Access it from another Java class.
8	<p>Write a Java program to create an <b>ArrayList</b> and perform add, remove, and iterate operations.</p> <p>Write a Java program to create a <b>Vector</b> and demonstrate its methods.</p>
9	<p>a)Write a Java program to implement different types of Inheritance.  b)Create a base class Shape with a method area(). Override this method in subclasses like Rectangle and Circle.  d).Create a class Child that calls the constructor of the parent class Parent using the super keyword.</p>
10	<p>a)Write a Java program to demonstrate the implementation of abstraction using abstract class and interface  b)Create an interface Playable with a method play(). Implement this interface in classes Song and Video.</p>
11	<p>a)Write a program to implement multiple inheritance  b)Write a Java program to work with lambda expressions.</p>
<b>Total Hours:30</b>	