U.S.N.										

B.M.S. College of Engineering, Bengaluru-560019

Autonomous Institute Affiliated to VTU

June 2025 Semester End Main Examinations

Programme: B.E. **Semester: IV**
Branch: Information Science and Engineering **Duration: 3 hrs.**
Course Code: 23IS4PCJAV / 19IS4PCJAV **Max Marks: 100**
Course: Java Programming

Instructions: 1. Answer any FIVE full questions, choosing one full question from each unit.
2. Missing data, if any, may be suitably assumed.

Important Note: Completing your answers, compulsorily draw diagonal cross lines on the remaining blank pages. Revealing of identification, appeal to evaluator will be treated as malpractice.

		UNIT - I	*CO*	*PO*	Marks
1	a)	Create a class Book with data members: ISBN, title, author, and price. Include accessor, mutator methods and constructors. Write the main method to create and display details of N Book objects.	*CO2*	*PO1*	**10**
	b)	Create a class Movie with data members: movieId, title, director, genre, releaseYear, and rating. Write methods to: (i) count movies of a particular genre, and (ii) find and display the movie with the highest rating.	*CO2*	*PO1*	**10**
		OR			
2	a)	Design a Java program with an Employee class having employeeId, name, and salary. Write methods i)PrintEmployeeDetails() to display employee details ii) max(Employee emp[]) method to find the name of the employee with highest salary. Write a driver class to create an array of 'N' Employee objects and print all their details.	*CO2*	*PO1*	**10**
	b)	Class DiscountCalculator{ calculateDiscount(string Purchaseamount) {---} calculateDiscount(string Purchaseamount,String membership) {---} calculateDiscount(string Purchaseamount,String membership, int offer){ --- }} Identify the concept in the above snippet and complete the code by defining the above methods and the main method.	*CO2*	*PO1*	**10**

			UNIT - II			
	3	a)	Create a package **geometryutils** with an interface **ShapeOperations** having methods area(double a, double b); and perimeter(double a, double b). Implement this interface in a class Rectangle. In the default package create a class TestRectangle which has main method to create an object of class Rectangle and invoke methods to print the area and perimeter for given dimensions.	***CO2***	***PO1***	**10**
		b)	Create three interfaces—LightControl (with methods to turn lights on/off), TemperatureControl (method to set temperature), and SecurityControl (methods to activate/deactivate alarm). Implement all the three interfaces in a class SmartHome with appropriate method definitions. In TestSmartHome class, create a SmartHome object and invoke all the methods.	***CO2***	***PO1***	**10**
			OR			
	4	a)	Create a class Employee with a constructor (name, salary) and a method showDetails(). Create an interface BonusEligible with a method checkBonusEligibility(). Create a class PermanentEmployee that extends class Employee and implementsthe interface BonusEligible. The checkBonusEligibility() method should check for eligibility for bonus. Emplyoee is eligibile for bonus if his salary is greater than ₹50,000. Write the main method to create a PermanentEmployee object and invoke the methods.	***CO2***	***PO1***	**10**
		b)	Create three interfaces— AudioControl (methods: play(), stop()), VideoControl (methods: startVideo(), stopVideo()), and RecordingControl (methods: startRecording(), stopRecording()). Implement these interfaces in a class MediaDevice with method implementations. In a TestMediaDevice class, write a main method to create an object of MediaDevice and invoke all its methods.	***C02***	***PO1***	**10**
			UNIT - III			
	5	a)	Create a class userregistration system which has member variable age and method validateage(int age). If the age entered, is less than 18 throw an IllegalArrgumentException and if age entered is a character throw InputMismatchException. Print the message "Validation complete" in the finally block.	***CO3***	***PO2***	**8**
		b)	Write a Java program to simulate a restaurant with a Kitchen class that has a chef as member variables and waiter class with	***CO4***	***PO3***	**8**

			place_order as member variables. Multiple waiters objects can place orders, but the chef can prepare only one order at a time. Ensure that only one waiter's order is processed at a time.			
		c)	Create a thread that prints numbers from 1 to 5 with a short delay. use isAlive() to check if the thread is still running once the thread is started and also after thread terminates. Print appropriate messages.	***CO2***	***PO1***	**4**
			OR			
	6	a)	Imagine you are designing a banking application where withdrawal amount should not exceed the available balance. Create a custom exception called InsufficientBalanceException. Write a Java program that throws this exception if the withdrawal amount is greater than the balance and handles it in main().	***CO3***	***PO2***	**8**
		b)	Create a Java program to simulate multiple employees printing documents using a shared printer that as only one method printDocument(String name, int pages). Create multiple threads for employees to use this printDocument method. Ensure that only one employee 's document is printed at a time.	***CO4***	***PO3***	**8**
		c)	Create a thread that prints the "hello from the thread" n times. Ensure that the main thread waits for the thread to terminate.	***CO2***	***PO1***	**4**
			UNIT - IV			
	7	a)	Write a Java program that writes the text "All the best" to a file named message.txt. Then, read the content from the same file and display it on the screen.	***CO3***	***PO2***	**10**
		b)	Imagine you are developing a simple document analyzer for a text editor. Write a Java program that performs the following operations on a string "WelcomeToJavaWorld": 1. extract characters from index 0 to 6 into a character array and display it. 2. compare the original string with WelcomeToPythonWorld". 3. check if the original string is equal to WelcomeToJavaWorld". 4. find the last occurrence of the character 'o'. 5. check if the string ends with "World".	***CO3***	***PO2***	**10**
			OR			
	8	a)	Write a Java program that uses Print Writer to write three lines of text to a file named output.txt. After writing, read the content of the file and display it on the screen.	***CO3***	***PO2***	**10**
		b)	You're building a basic string processor for a note-taking app. Using the StringBuffer class in Java, perform the following operations after creating an initial string "Hello": 1. Append the text " World" to the string. 2. Insert "Java " at index 6. 3. Replace the word "Java" with "Awesome".	***CO3***	***PO2***	**10**

			4. Delete the word "Hello". 5. Reverse the final string. 6. Set the character at index 0 to 'X'. 7. Get the length of the final modified string and display it. 8. Convert the string to uppercase and display it.			
			UNIT - V			
	9	a)	Write a program to create a generic class that performs sum of all the elements in an array.	***CO4***	***PO3***	**10**
		b)	Create a LinkedList object to store 4 student names. Perform the following operations 1. Add "Raju" to the list at index 1. 2. Remove at index 2 3. Insert names at the beginning 4. Insert at the end of the list. 5. Display the list before and after modifications.			**5**
		c)	Write a Java program that creates two HashSet objects storing city names. Add some cities to both objects (with duplicates between them). Add all elements from the second object into the first object using an appropriate method. Display the contents of both sets before and after the addition.	***CO4***	***PO3***	**5**
			OR			
	10	a)	Create a generic class ProductInfo<K, V> with a constructor, getters, and a method to print key-value pairs. Use it to store a product ID and price (Integer, Double), and a product name and availability status (String, Boolean).In main(), create objects with these pairs and display their info using the method.	***CO4***	***PO3***	**10**
		b)	Create an ArrayList to store city names. Add at least seven city names to the list. Display the list and its size. Remove the city at index 2 and 4. Display the updated list and its size after removals.	***CO4***	***PO3***	**5**
		c)	Explain Autoboxing and unboxing with an example	***CO4***	***PO3***	**5**

****