Date: 19 / 06 / 2024

Practical List

Academic Year	:	2024-25	Semester	:	3
Course code	:	CSE201	Course name	:	Java Programming

Sr.No	AIM	Hours	CO
	PART-I **Data Types, Variables, String, Control Statements, Operators, Arrays**		
1.	Introduction to Object Oriented Concepts, comparison of Java with other object-oriented programming languages. Introduction to JDK, JRE, JVM, Javadoc, command line argument. Introduction to Eclipse or NetBeans IDE,or BlueJ and Console Programming.	1	1
2.	Write a program that declares one integer variable called var1. Give value 10 to this variable and then, using one println() statement, display the value on the screen like this: “10 is the value of var1.”	1	1
3.	Write a console program to declare and initialize a double variable with some value such as 1234.5678. Then retrieve the integral part of the value and store it in a variable of type long, and the first four digits of the fractional part and store them in an integer of type short. Display the value of the double variable by outputting the two values stored as integers.	1	1
4.	Write an application that creates a two-dimension array with int values. The first, second and third elements should be arrays with one, two and three numbers respectively. Display the length of each dimension.	1	1,2
5.	An electric appliance shop assigns code 1 to motor,2 to fan,3 to tube and 4 for wires. All other items have code 5 or more. While selling the goods, a sales tax of 8% to motor,12% to fan,5% to tube light,7.5% to wires and 3% for all other items is charged. A list containing the product code and price in two different arrays. Write a java program using switch statement to prepare the bill.	1	1,2
6.	Write a program in Java to reverse the digits of a number using while loop.	1	1,2
	PART-II **String**		

1.	Given a string and a non-negative int n, we'll say that the front of the string is the first 3 chars, or whatever is there if the string is less than length 3. Return n copies of the front; front_times('Chocolate', 2) → 'ChoCho' front_times('Chocolate', 3) → 'ChoChoCho' front_times('Abc', 3) → 'AbcAbcAbc'.	1	1,2
2.	Given an array of ints, return the number of 9's in the array. array_count9([1, 2, 9]) → 1 array_count9([1, 9, 9]) → 2 array_count9([1, 9, 9, 3, 9]) → 3	1	1,2
3.	Given an array of ints, return True if one of the first 4 elements in the array is a 9. The array length may be less than 4. array_front9([1, 2, 9, 3, 4]) → True array_front9([1, 2, 3, 4, 9]) → False array_front9([1, 2, 3, 4, 5]) → False	1	1,2
4.	Given a string, return a string where for every char in the original, there are two chars. double_char('The') → 'TThhee' double_char('AAbb') → 'AAAAbbbb' double_char('Hi-There') → 'HHii--TThheerree'	1	1,2
5.	Write a program that will reverse the sequence of letters in each word of your chosen paragraph. For instance, “To be or not to be” would become “oT eb ro ton ot eb”	1	1,2
6.	Perform following functionalities of the string: ● Find Length of the String ● Lowercase of the String ● Uppercase of the String ● Reverse String ● Sort the string	1	1,2
7.	Perform following Functionalities of the string: “CHARUSAT University” ● Find length ● Replace ‘H’ by ‘N’ ● Convert all character in Uppercase ● Extract and print “CHARUSAT” from given string	1	1,2
PART- III **Object Oriented Programming: Classes, Methods, Constructors**			
1	Write a java program for converting Pound into Rupees. (Accept Pounds from command line argument and using scanner class also and take 1 Pound = 100 Rupees.).	1	2
2.	Write a program that defines TriangleArea class with three constructors. The first form accepts no arguments. The second accept one double value for	1	1,2

	radius. The third form accepts any two arguments		
3.	Create a class called Employee that includes three pieces of information as instance variables—a first name (type String), a last name (type String) and a monthly salary (double). Your class should have a constructor that initializes the three instance variables. Provide a set and a get method for each instance variable. If the monthly salary is not positive, set it to 0.0. Write a test application named EmployeeTest that demonstrates class Employee's capabilities. Create two Employee objects and display each object's yearly salary. Then give each Employee a 10% raise and display each Employee's yearly salary again	1	1,2
4.	Create a class called Date that includes three pieces of information as instance variables—a month (type int), a day (type int) and a year (type int). Your class should have a constructor that initializes the three instance variables and assumes that the values provided are correct. Provide a set and a get method for each instance variable. Provide a method displayDate that displays the month, day and year separated by forward slashes (/). Write a test application named DateTest that demonstrates class Date's capabilities	1	1,2
5.	Write a program to print the area of a rectangle by creating a class named 'Area' taking the values of its length and breadth as parameters of its constructor and having a method named 'returnArea' which returns the area of the rectangle. Length and breadth of rectangle are entered through keyboard	1	1,2
6.	Print the sum, difference and product of two complex numbers by creating a class named 'Complex' with separate methods for each operation whose real and imaginary parts are entered by user	1	1,2
7.	Develop minimum 4 program based on variation in methods i.e. passing by value, passing by reference, returning values and returning objects from methods	1	2
PART-IV **File Handling & Streams**			
1.	Create a class with a method that prints "This is parent class" and its subclass with another method that prints "This is child class". Now, create an object for each of the class and call 1 - method of parent class by object of parent class 2 - method of child class by object of child class 3 - method of parent class by object of child class	1	1,2,3
2.	Create a class named 'Member' having the following members: Data members 1 - Name 2 - Age 3 - Phone number 4 - Address 5 – Salary It also has a method named 'print Salary' which prints the salary of the members. Two classes 'Employee' and 'Manager' inherits the 'Member' class. The	1	1,2,3

	'Employee' and 'Manager' classes have data members 'specialization' and 'department' respectively. Now, assign name, age, phone number, address and salary to an employee and a manager by making an object of both of these classes and print the same.		
3	Create a class named 'Rectangle' with two data members 'length' and 'breadth' and two methods to print the area and perimeter of the rectangle respectively. Its constructor having parameters for length and breadth is used to initialize length and breadth of the rectangle. Let class 'Square' inherit the 'Rectangle' class with its constructor having a parameter for its side (suppose s) calling the constructor of its parent class as 'super(s,s)'. Print the area and perimeter of a rectangle and a square. Also use array of objects	1	2,3
4.	Write an application that illustrates how to access a hidden variable. Class A declares a static variable x. The class B extends A and declares an instance variable x. display( ) method in B displays both of these variables	1	2,3
5.	Create a class 'Degree' having a method 'getDegree' that prints "I got a degree". It has two subclasses namely 'Undergraduate' and 'Postgraduate' each having a method with the same name that prints "I am an Undergraduate" and "I am a Postgraduate" respectively. Call the method by creating an object of each of the three classes	1	2,3
6.	Write a program that illustrates interface inheritance. Interface P12 inherits from both P1 and P2. Each interface declares one constant and one method. The class Q implements P12. Instantiate Q and invoke each of its methods. Each method displays one of the constants.	1	2,3
7.	Assume you want to capture shapes, which can be either circles (with a radiusand a color) or rectangles (with a length, width, and color). You also want to be able to create signs (to post in the campus center, for example), each of which has a shape (for the background of the sign) and the text (a String) to put on the sign. Create classes and interfaces for circles, rectangles, shapes, and signs. Write a program that illustrates the significance of interface default method	1	2,3
8.	Write a java program which shows importing of classes from other user define packages	1	6
PART-V **Exception Handling**			
1.	Write a java program which takes two integers x & y as input, you have to compute x/y. If x and y are not integers or if y is zero, exception will occur and you have to report it	1	4
2.	A piece of Java code is given below. You have to complete the code by writing down the handlers for exceptions thrown by the code. The exceptions the code may throw along with the handler message are listed below: Division by zero: Print "Invalid division". String parsed to a numeric variable: Print "Format mismatch". Accessing an invalid index in string: Print "Index is invalid".	1	4

	Accessing an invalid index in array: Print "Array index is invalid". MyException: This is a user defined Exception which you need to create. It takes a parameter param. When an exception of this class is encountered, the handler should print "MyException[param]", here param is the parameter passed to the exception class. Exceptions other than mentioned above: Any other exception except the above ones fall in this category. Print "Exception encountered". Finally, after the exception is handled, print "Exception Handling Completed". Example: For an exception of MyException class if the parameter value is 5, the message will look like MyException[5.		
3.	Write a java program to generate user defined exception using “throw” and “throws” keyword. Also Write a java that differentiates checked and unchecked exceptions. (Mention at least two checked and two unchecked exception in program).	1	4
	PART-VI **File Handling & Streams**		
1	Write a program that will count the number of lines in each file that is specified on the command line. Assume that the files are text files. Note that multiple files can be specified, as in "java Line Counts file1.txt file2.txt file3.txt". Write each file name, along with the number of lines in that file, to standard output. If an error occurs while trying to read from one of the files, you should print an error message for that file, but you should still process all the remaining files.	1	4,6
2.	Write an example that counts the number of times a particular character, such as e, appears in a file. The character can be specified at the command line. You can use xanadu.txt as the input file	1	4,6
3.	Write a Java Program to Search for a given word in a File. Also show use of Wrapper Class with an example	1	4,6
4.	Write a program to copy data from one file to another file. If the destination file does not exist, it is created automatically	1	4,6
5.	Write a program to show use of character and byte stream. Also show use of BufferedReader/BufferedWriter to read console input and write them into a file	1	4,6
	PART-VII **Multithreading**		
1	Write a program to create thread which display “Hello World” message. A. by extending Thread class B. by using Runnable interface..	1	5,6

2.	Write a program which takes N and number of threads as an argument. Program should distribute the task of summation of N numbers amongst number of threads and final result to be displayed on the console	1	5,6
3.	Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number	1	5,6
4.	Write a program to increment the value of one variable by one and display it after one second using thread using sleep() method.	1	5,6
5.	Write a program to create three threads 'FIRST', 'SECOND', 'THIRD'. Set the priority of the 'FIRST' thread to 3, the 'SECOND' thread to 5(default) and the 'THIRD' thread to 7.	1	4,6
6.	Write a program to solve producer-consumer problem using thread synchronization.	1	5,6
PART-VIII **File Handling & Streams**			
1	Design a Custom Stack using ArrayList class, which implements following functionalities of stack. My Stack -list ArrayList<Object>: A list to store elements. +isEmpty: boolean: Returns true if this stack is empty. +getSize(): int: Returns number of elements in this stack. +peek(): Object: Returns top element in this stack without removing it. +pop(): Object: Returns and Removes the top elements in this stack. +push(o: object): Adds new element to the top of this stack	1	5
2.	Create a generic method for sorting an array of Comparable objects	1	5
3.	Write a program that counts the occurrences of words in a text and displays the words and their occurrences in alphabetical order of the words. Using Map and Set Classes	1	5
4.	Write a code which counts the number of the keywords in a Java source file. Store all the keywords in a HashSet and use the contains () method to test if a word is in the keyword set	1	5