A. V. PAREKH TECHNICAL INSTITUTE, RAJKOT
JAVA PROGRAMIMNG (3350703)
Practical List ODD 2020-21

LIST OF PROGRAMS	TITLE
CAHPTER-1: Introduction to Java	
1.1 WAP to print "Hello World" on console window.	Hello World
1.2 WAP to print the sum of two variables taken in a java program and initialized in main() and print the result in Console window.	Sum of 2 nos
1.3 WAP to print following pattern: a. 1 0 1 1 0 1 0 1 0 1 1 0 1 0 1 b. 1 2 1 2 3 2 1 2 3 4 3 2 1 2 3 4 c. 1 2 3 4 5 16 17 18 19 6 15 24 25 20 7 14 23 22 21 8 13 12 11 10 9 (Spiral Matrix)	Patterns
CAHPTER-2: Building Bblocks of the Language	
2.1 WAP to implement a simple calculator using switch statement.	Calculator using Switch
2.2 WAP to find whether the given number is Prime or not and take the number from Command Line.	Prime Number using Command Line Args
2.3 WAP to print the Fibonacci Series up to the number of terms entered by user.	Fibonacci Series
2.4 WAP to print the answer of the following series for n terms [Series: 1 + 1/2 + 1/3 + … + 1/n]	Series using Loop
2.6 WAP to print the maximum of 3 numbers which are entered by the user from command line	Max of 3 using Command Line
2.6WAP to check whether the entered string is Palindrome or not.	Palindrome String
2.7 WAP to take the average of 10 numbers taken in an array.	Average of Array
2.8 WAP to take 10 integer data from user and sort them into descending order.	Sorting in Ascending Order
2.9 WAP to use Math Class Functions.	Math Class Functions
2.10 WAP to use functions of String class.	String Class Functions
2.11 WAP to demonstrate the use of Wrapper Classes.	Wrapper Class
2.12 WAP to find out factorial of the number entered by user using recursion.	Factorial using Recursion
2.13 WAP to display the sum of the digits in a number entered by user.	Sum of Digits in a number
2.14 WAP to multiply 2 matrices of 3x3.	Matrix Multiplication

CAHPTER-3: Object Oriented Programming Concepts	
3.1 Create a class Point having x and y co-ordinates stored as members. Create a constructor to initialize the point and include methods to change the co-ordinates of the point and to print the value of co-ordinates on the screen. Create on method distance() which prints the distance between 2 points.	Point Class
3.2 Create a class StuResult and take members enr_no, marks[3] for storing Students Enrollment Number and marks of 3 subjects respectively. Create a constructor of a class to initialize the same and create a method which displays the result of student whether he is passing or not. Create 5 objects of students and take their data from student using Array of Objects.	Array of Student Object
3.3 Create a class Rectangle & create overloaded constructors to initialize length and width, where user can either enter only 1 argument or 2 arguments as length and width. Create a method to display area of the rectangle.	Overloaded Constructors in Rectangle Class
3.4 Create a class Time having members hour, min and sec. Create getters and setters to get and set the values of time. Create a method which takes 2 Time class objects as argument and gives the total time in return as another object of Time class only.	Time Class
3.5 WAP to find out the total number of objects created in program. Create method showCount() to print the total count of the objects created. Use Static member count for it.	Count number of Object using Static Members
CAHPTER-4: Inheritance, Packages & Interfaces	
4.1 Create a class Rect and have data member length and width and a method to find the area of the rectangle. Create a subclass Cube from that add a member height to it and create a method to find the volume of the Cube from that.	Simple Inheritance using Rect and Cube Class
4.2 Create a class Person that have properties like name and age, create its constructor to initialize those. Create one subclass of Person as Student having a member courseName also create a constructor. Now create another subclass of Person as Employee and create a member called salary, also initialize each member with a constructor. Now you should create a showDetails() method in each class to display details. Create objects of different classes and initialize them and call showDetails() method from each object and show the output.	Hierarchichal Inheritance using Person, Student and Employee Class
4.3 Create an interface ShapeArea having a method declared called findArea() now implement this interface to 2 classes called circle and Rectangle and implement findArea() accordingly.	Implementing interface ShapeArea
4.4 Create Interface Student with data enrno and name, extend 2 sub interfaces called Exam containing data member CPI and another Sports containg data members event and marks, now derive interfaces Student and Sports into another class StudentPerformance containing display() method to display all the info of student.	Multiple Inheritance and Dimond Formation
4.5 Create a package p1 having class A in that and create show() method to display the message to show the class name. Now create a package p2 outside package p1 and create a class B with a show method in it. Now in the main function from class B call show method of both classes one by one by using the concept of Dynamic Method Dispatch.	Packages and Dynamic Method Dispatch
CAHPTER-5: Exception Handling and Multithreaded Programming	
5.1 WAP to demonstrate the handling of exception occurred when divide by 0 operation is done and show an appropriate message. Then use the finally block to show that it is been executed when.	Divide by Zero and Finally Block
5.2 WAP to access the 11^{th} element of the array of size 10 and handle the exception.	ArrayIndexOutOfBoundsException
5.3 WAP to use the keyword throw which will throw the NullPointerException with a messeage “hello”.	throw Keyword

5.4 WAP to demonstrate multiple catch blocks that are capable of handling ArithmaticException and ArrayOutOfBoundsException. In this create an array of 5 elements with a value of 1 element set to 0 and divide a digit with the array elements one by one and loop should run between 0 to 5 values.	Multiple Catch Blocks
5.5 Create a class ThrowsDemo and create a method divide() which throws an exception ArithmaticException if required and create a program using that and use divide method to perform division operation on 2 numbers.	throws Keyword demo
5.6 WAP to create a custom Exception class called MyException of type Exception that will throw an exception if the value passed in a method named compute() is greater than 30. This will generate an Exception and should print the message “The value must be entered less than or equal to 30”	Custom Exception Class
5.7 Create a 2 Thread type classes implement Runnable interface on one and extend Thread class on other one and those threads should print 1 to 10 on the console window and run those threads from another class having main function.	Multithreading Demo
5.8 Create an example to demonstrate 2 Synchronised Thread. Each thread should have a loop to print a message "Thread 1" and "Thread 2" and thread 1 should sleep or 500 ms and thread 2 should sleep for 300 ms after each loop execution.	Thread Synchronization
CAHPTER-6: File Handling	
6.1 Create a file named Students.txt having the data of 3 students of their name, number and marks of 3 subjects. Get those data from user.	Writing in a File
6.2 Read the contents of file Student.txt and display it in a tabular form in the console window.	Reading From a File
6.3 Create a new file called StuPercentage.txt and store the data of name, number and percentage of each student. Get those data from Student.txt.	Rading from one file and writing into another
6.4 Create a file Employee.txt having employee details of name, empID and salary.	Create Employee Details
6.5 Read the file Employee.txt and get the data on console window.	Rading Employee Details
6.6 Create a file called NewSalary.txt and increase each employee’s salary by 10% which was stored in Employee.txt. Store Name, EmpId and Salary in this file also.	UpdatingEmployee Salary into another file

FACULTY NAME: **N. R. TRIVEDI & N. R. RADADIYA**

Sign
Head Of Dept
Computer Enginnering