TW1

1. Write a Java program to accept IA marks obtained by five students in three subjects. The program should accept marks obtained by each student and display the total marks and the average marks. The average marks is computed as the average of best two marks obtained.
2. A company has 10 zonal sales offices in four zones namely, North, East, West and South. The company wants to organize the sales data of each of the office in each zone and find answers to queries such as,
3. Which office has performed the highest sales in each zone?
4. What is the average sales done by all the offices in each zone?
5. Which office among each zone is the poorly office?

You are required to answer the following:

1. How do you organize the above data?
2. How do you provide answers to the above queries?

Design a Java application for the same and demonstrate the correctness of the solution.

TW2

1. Design a class called Rectangle having two methods. First method named as setDim() takes length and breadth of rectangle as parameters and the second method named as getArea() returns the area of the rectangle. Length and breadth of rectangle are entered through keyboard.
2. Design a class called Triangle. Include methods:
   * 1. getSides() to initialize the three sides of triangle.
     2. checkType() to determine the type of triangle represented by the three sides (Equilateral/ Isosceles/ Scalene triangle).
     3. computeArea() to compute and return the area of the triangle.

TW3

* 1. A certain small bank intends to automate few of its banking operations for its customers. Design a class by name mybankAccount to store the customer data having following details: 1.accountNumber 2. acctType 3. Name 4. Address 5. accountBalance.

The class must have both default and parameterized constructors. Write appropriate method to compute interest accrued on accountBalance based on accountType and time in years. Assume 5% for S/B account 6.5% for RD account and 7.65 for FD account. Further, add two methods withdrawAmount/depositAmount with amount as input to withdraw and deposit respectively. The withdrawAmount method must report in-sufficient balance if accountBalance falls below Rs. 1000.

TASK 1: Build the class with appropriate member variables, constructors and methods.

TASK 2: Instantiate three objects of above type and perform different operations on the same.

TASK 3: Write a function to display all the three customer details in a tabular form with appropriate column headings.

* 1. Define a class to represent a rectangle in which constructors and parameterized methods are used. It also has a method to compute area of rectangle.
  2. First make a class rectangle in which we declare the parameterized constructor.
  3. Then demonstrate the use of parameterized method.
  4. Use a method to compute area of rectangle.
  5. Create a class to demonstrate the call of the methods in previously created class rectangle holding constructors, parameterized methods and method to compute area of rectangle.

TW4

1. A company has two types of employees – FullTime and Partime. The company records for each employee his/her name, age, address, salary and gender. Given the basic salary of the FullTime employee the components of his/her gross salary are: Dearness allowance – 75% of basic salary, HRA – 7.5% of basic salary, IT – 10% of basic. The salary of a Partime employee is dependent on the qualification, experience, number of working hours and the rate per hour, as below:

|  |  |  |  |
| --- | --- | --- | --- |
|  | Qualification | | |
| Experience | B.E. | M.Tech | PhD |
| 1-5 Years | 300 Rs. | 500 Rs. | 800 Rs. |
| 6-10 Years | 400 Rs. | 700 Rs. | 1200 Rs. |
| >10 Years | 500 Rs. | 1000 Rs. | 1500 Rs. |

Model this as a problem of hierarchical inheritance by:

1. Identifying the super class with its data members and member functions.
2. Identify the sub-class/sub-classes and their associated data members and member functions.

Test the program by creating objects of the classes that are so identified.

1. The class Cylinder inherits all the instance variables (radius and color) and methods (getRadius(), getArea(), among others) from its superclass Circle. It further defines a variable called height, three methods getHeight(), setHeight() and getVolume() and its own constructors. Implement the hierarchy as shown below.

Cylinder()

Cylinder(height:double)

Cylinder(height:double, radius:double)

Cylinder(height:double, radius:double,

color:String)

getHeight():double

setHeight(height:double):void

getVolume():double

TW5 (METHOD OVERLOADING)

1. Create a Stack class having an integer array say elem and top\_of\_stack as instance variables. Define three overloaded methods having the following signatures:
   * 1. initStack(int size) to create an array of specified size and initialize the top\_of\_stack
     2. initStack(Stack another) to intialize the Stack object with state of the Stack object "another"
     3. initStack(int [] a) to initialize contents of a[] to the instance variable elem.

Write following methods:

1. push(): Pushes the element onto the stack,
2. pop(): Returns the element on the top of the stack, removing it in the process, and
3. peek(): Returns the element on the top of the stack, but does not remove it.

Also write methods that check whether stack is full and stack is empty and return boolean value true or false appropriately.

1. Implement a linear search function by using method overloading concept for an array of integers, double and character elements.

TW5 (METHOD OVERRIDING)

1. Implement the following class hierarchy. In the Cuboid class, override the method computeArea() and computePerimeter() of Rectangle class to compute the surface area and perimeter of a rectangle cuboid. Add a method computeVolume() in Cuboid class to compute volume of the cuboid. Assuming length, width and height as l, w and h respectively,

* formula to find the surface area = 2(lw) + 2(hl) + 2(hw)
* formula to find the perimeter = 2l + 2w
* formula to find the volume = l x w x h

Rectangle

length:double = 1.0

width:double = 1.0

Rectangle()

Rectangle(length, width)

computeArea():double

computePerimeter():double

Cuboid

height:double = 1.0

Cuboid()

Cuboid(length, width, height)

computeArea():double

computePerimeter():double

computeVolume():double

1. Design a base class ArrayStack that uses array to hold the elements and has 3 methods namely, push, pop and display. Derive a class LinkedStack that overrides these 3 methods and uses linked list to implement stack. Demonstrate the working of both the classes by performing push, pop and display operations on the objects of the above to classes.

TW6

1. Design an abstract class Car to have carName, chassiNum, modelName as member variables and add two abstract methods, startCar and operateSteering . Inherit MarutiCar and BmwCar from Car class and override the two abstract methods in their own unique way.

Design a driver class to have driver name, gender and age as data members and add a method driveCar with abstract class reference variable as argument and invoke the two basic operations namely, startCar and operateStearing and demonstrate run-time polymorphism.

1. Implement the following inheritance hierarchy.

Shape (Abstract class)Circle

area,perimeter: float

type: Stringradius:double = 1.0

color:String = “Red”

abstract computeArea()

abstract computePerimeter()Circle()

Circle(radius:double)

Circle(radius:double, color:String)

getRadius():double

setRadius(radius:double):void

getColor():String

getColor(color:String):void

getArea():double

Rectangle

(Concrete class)

length, width: floatCylinder

height:double = 1.0

Circle

(Concrete class)

radius: float

Triangle

(Concrete class)

base, height: float

TW 7

1. Write a Java application to implement the following UML diagram.
   1. PrimeTester class implements isPrime() method by iterating from 2 to n-1 for a given number n
   2. ImprPrimeTester class implements isPrime() method by iterating from 2 to n/2
   3. FasterPrimeTester class implements isPrime() method by iterating from 2 to
   4. FastestPrimeTester class implements isPrime() method using Fermat’s Little theorem. o Fermat’s Little Theorem:

If n is a prime number, then for every a, 1 < a < n-1, an-1 % n = 1

1. Design an interface IAnimal that has walk, and sleep methods, an interface IBird that has fly, and peck methods, an interface IHuman that has eat and speak methods. Construct a Bird class that implements IAnimal and IBird interfaces and also construct Human class that implements IAnimal and IHuman interfaces. Demonstrate the working of these methods by invoking the methods using appropriate reference variables.

DemoClass

Human

Bird

IHuman

+speak( ) +eat( )

IBird

+fly( ) + peck( )

IAnimal

+sleep( ) +walk( )

TW 8

1. Assume that you have received a request from the transport authority for automating the task of issuing the permanent license for two wheelers. The mandatory condition to issue the license are: 1) the applicant must over 18 years of age and 2) holder of a valid learner’s license and 3) no accident cases in the last one year.

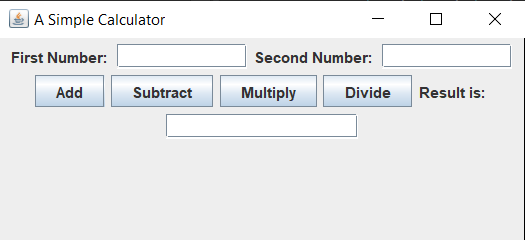
Write a Java program that reads user details as required (use the Scanner class). Create user defined exceptions to check for the three conditions imposed by the transport authority.

Based on the inputs entered by the user, decide and display whether or not a license has to be issued or an error message as defined by the user exception.

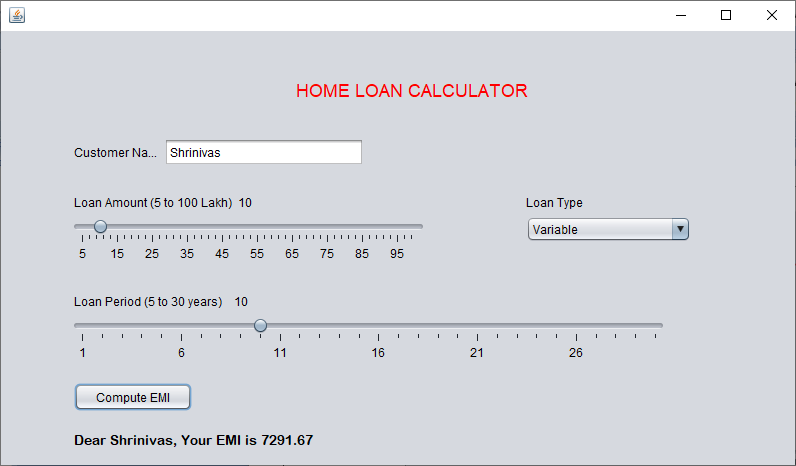
1. Write java program that takes the value of num variable and checks it is odd, then the throw keyword will raise the user defined exception and the catch block will get execute. OddNumberException class is derived from the Exception class. To implement user defined exception throw an exception object explicitly.

TW9

1. Read a string containing 3 to 4 words using Scanner class object. Split it into words and for each word check if it's palindrome by writing a function isPalindrome(String the myWord, int s, int e) which return true if its palindrome else return false. Where s is start index and e is end index of the input myWord. Print it in uppercase if it is palindrome else reverse the string and print it in lowercase. Use appropriate string functions to implement the above problem statement.
2. Two strings will be anagram to each other if and only if they contain the same number of characters (order of the characters doesn't matter). That is, If the two strings are anagram to each other, then one string can be rearranged to form the other string. For Example: creative and reactive are anagrams. Write a Java program to test whether two strings are anagrams or not. (listen and silent, stressed and desserts, dusty and study)

TW 10

1. Design and develop a GUI application as shown below. Assume the two numbers to be integers. The application must check for invalid division condition and throw an appropriate exception.
2. Design and implement a Home loan Emi calculator using appropriate Swing components. The GUI should like as under:



The formula to compute Home loan EMI for a given Principal amount PA and interest rate IR for a period of T years is

EMI = ( PA + (PA \* IR \* T))/12\*T ;

Hint : Use SlideBars for Amount and Loan Period LoanType - ComboBox