Week1::Day2::LabCodingChallenges

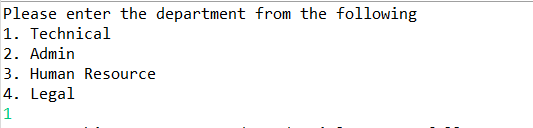
# Problem Statement:

You are an IT Support Administrator and are charged with the task of creating credentials for new hires

Yourapplicationshoulddothefollowing:

1. Generate an email with the following syntax [firstNamelastName@department.company.com](mailto:firstNamelastName@department.company.com)
2. Determine the department(Technical,Admin,HumanResource,Legal)
3. Generate a random password which will contain(number,capitalletter,smallletter& special character)
4. Display the generatedcredentials
5. Use parameterized constructor of class Employee,topassfirstName,lastName.
6. Create a separate CredentialService which will have generatePassword, generateEmailAddress, & showCredentials method.

# Sample output



Dear Harshit your generated credentials are as follows:

Email ---> [harshitchoudary@tech.gl.com](mailto:harshitchoudary@tech.gl.com) Password ---> 181E@wFT

# Concepts to be used

Must use Abstraction, Inheritance, Encapsulation, AccessModifiers.

Follow appropriate package structure and namingconventions throughout the application.

Question 2: Write a program to pass your friends names as an array to a constructor. create a method sort which can sort the names in ascending order based on the first letter of their names and return the sorted array. Print the sorted array. You can either hardcode the names or you can accept it from the user.

Sample output:

Array of names after sorting

Anirudh

Anirudh

Mukesh

Ramesh

Raju

Sridhar

Suresh

Zarina

Question 3:

Write a program in java to illustrate the constructor overloading. (cube and cuboid volume

calculation)

i) Create a class ThreeDimensionShape with three variables width, height, depth of type

double;

ii) Create three constructors

One which Doesn’t accept any parameter

Another one which Accepts 1 parameter of type double(length)

Another one which Accepts 3 parameters of type double(width, height, depth)

iii) Create a method calculateVolume of return type double that returns a product of width,

height, and depth.

iv) in case of No argument constructor make length=width=depth = 0;

v) in case of 1 argument constructor make length=width=depth = value passed;

Stub Code:

public class ThreeDimensionShape {

double width, height, depth;

// constructor used when all dimensions are defined

ThreeDimensionShape (double w, double h, double d) {

}

// constructor used when one dimension is defined

ThreeDimensionShape (double l) {

}

// constructor used when no dimension is specified

ThreeDimensionShape() {

}

// compute and return volume

double calculateVolume() {

return width \* height \* depth;

}

}

public class ConstructorImplementation {

public static void main(String args[]) {

// create boxes using the various

// constructors

ThreeDimensionShape shape1 = new ThreeDimensionShape(5, 6, 7);

ThreeDimensionShape shape2 = new ThreeDimensionShape();

ThreeDimensionShape shape3 = new ThreeDimensionShape(8);

double volume;

// get volume of first box

volume = shape1.calculateVolume();

System.out.println(" Volume of shape1 is " + volume);

// get volume of second box

volume = shape2.calculateVolume();

System.out.println(" Volume of shape2 is " + volume);

// get volume of cube

volume = shape3.volume();

System.out.println(" Volume of shape3 is " + volume);

}

}

Sample output:

Volume of shape1 is 210.0

Volume of shape2 is 0.0

Volume of shape3 is 512.0

**Question 4:**

**Write an algorithm to create classes InheritanceA, InheritanceB and InheritanceC to find the area of circleArea, rectangleArea and triangleArea respectively.**

**Use multi level inheritance and execute all methods using the object of class InheritanceC.**

Note: create methods circleArea, rectangleArea and triangleArea in the classes InheritanceA,

InheritanceB and InheritanceC respectively.

Stub Code

1. InheritanceA.java

package com.glca.week1.day4.selfpractice;

public class InheritanceA {

// 1. Declare required variables

// 2. Create the parameterized method for Circle Area

public void findcCircleArea(float rad) {

// your code here

}

}

2. InheritanceB.java

package com.glca.week1.day4.selfpractice;

public class InheritanceB extends InheritanceA {

// 1. Declare required variables

// 2. Create the parameterized method for Rectangle Area

public void findRectangleArea(int l, int b) {

// your code here

}

}

3. InheritanceC.java

package com.glca.week1.day4.selfpractice;

public class InheritanceC extends InheritanceB {

// 1. Declare required variables

// 2. Create the parameterized method for triangle Area

public void findTriangleArea(int b, int h) {

}

public static void main(String[] args) {

// 3. Create object and call required methods

}

}

Sample Output 1:

The area of the circle is: 1518.75 cm

The area of the rectangle is: 30.0 cm square

The area of the triangle is: 6.0 cm square

**Question-5: Developing a Banking Application with Polymorphism in Java**

You are tasked with developing a banking application in Java that can handle different types of accounts, such as Savings Accounts and Checking Accounts. The application should demonstrate the use of polymorphism to provide a common interface for various account types while allowing each type to have its own specific functionality.

Requirements:

a) Implement a common interface, BankAccount, with methods for displaying the account balance (displayBalance) and performing a transaction (performTransaction).

b)Create a SavingsAccount class that extends the BankAccount interface. This class should include an additional method, applyInterest, which represents the specific logic for applying interest to a savings account.

c)Implement a CheckingAccount class that also extends the BankAccount interface. Override the performTransaction method to allow for overdrafts and include an appropriate overdraft limit.

d) In the main class (PolymorphismExample), demonstrate polymorphic behavior by creating instances of both SavingsAccount and CheckingAccount using the BankAccount interface.

e) Display the initial balances, perform transactions, and showcase specific behaviors such as applying interest to savings accounts and handling overdrafts in checking accounts.

f) Finally, display the updated balances after the transactions.

**Card Details**

Write a program to read and display the card details. A card can be any one of the two types, either Payback or Membership.

Based on the type of card, the kind of details to be displayed varies. Refer details below:

Payback Card:

1. Card Number

2. Points Earned

3. Total Amount

Membership Card:

1. Card Number

2. Rating

Create an abstract class named Card with the following protected attributes / member variables.

· String holderName;

· String cardNumber;

· String expiryDate;

Include appropriate getters and setters.

Include 3-argument constructor, the order of the arguments is holderName, cardNumber, expiryDate.

  Create a class named MembershipCard derived from Card. Include the following private attributes / member variables.

· Integer rating

Include appropriate getters and setters.

Include 4-argument constructor, the order of the arguments is holderName, cardNumber, expiryDate, rating.

  Create a class named PaybackCard derived from Card. Include the following private attributes / member variables.

· Integer pointsEarned;

· Double totalAmount;

Include appropriate getters and setters.

Include a 5-argument constructor, the order of the arguments is holderName, cardNumber, expiryDate, pointsEarned, totalAmount.

  Create another class called Main. In the method, create instances of the above classes and test the above classes.

Note: The card details are entered separated by a ‘|’.

Input and Output Format:

  Refer sample input and output for formatting specifications.

All text in bold corresponds to input and the rest corresponds to output.

Note: Code submission not required. To be implemented using eclipse IDE and manually verified by the SME.

Sample Input and Output 1:

  Select the Card

1.Payback Card

2.Membership Card

1

Enter the Card Details:

Anandhi|12345|14/01/2020

Enter points in card

1000

Enter Amount

50000

Anandhi's Payback Card Details:

Card Number 12345

Points Earned 1000

Total Amount 50000.0

Sample Input and Output 2:

  Select the Card

1.Payback Card

2.Membership Card

2

Enter the Card Details:

Collin|45678|20/11/2021

Enter rating in card

10

Collin's Membership Card Details:

Card Number 45678

Rating 10