

Programme Name: **BCS(HONS.)**

Course Code: **CSC 1513**

Course Name: **PROGRAMMING FUNDAMENTALS LAB**

Assignment / Lab Sheet / Project / Case Study No. \_\_**1**\_\_

Date of Submission: \_\_**24th April 2020**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Submitted By: Submitted To:**

Student Name: **Dipesh Tha Shrestha**  Faculty Name: **Pradip Baskota**

IUKL ID:

Semester: **Second**

Intake: **September 2019**

**1. Write a program that implements Interest class which stores three variables principal, rate and time and calculates simple interest. Create constructor(s) to initialize the variables. Write main method to test the class.**

Ans:

import java.util.Scanner;

public class FindInterest

{

double principal;

float rate;

float time;

// constructor

FindInterest(double principal, float rate, float time) {

this.principal = principal;

this.rate = rate;

this.time = time;

}

// method

public double calcInterest()

{

double SI = ((this.principal \* this.rate \* this.time) / 100);

return SI;

}

// main program

public static void main(String[] args)

{

Scanner sc = new Scanner(System.*in*);

System.*out*.println("Enter the principal, rate and time to calculate simple interest: ");

System.*out*.print("Principal: ");

double p = sc.nextDouble();

System.*out*.print("Rate(in %): ");

float r = sc.nextFloat();

System.*out*.print("Time(in year): ");

float t = sc.nextFloat();

// creating object

FindInterest obj1 = new FindInterest(p,r,t);

System.*out*.println("The simple interest for obj1 is " +obj1.calcInterest());

}

}

**2. Design a class to represent a bank account. Include the following members:**

**Fields/Data members**

**Name of the depositor**

**Account number**

**Type of account**

**Balance amount in the account**

**Methods**

**Constructor(s)**

**To assign initial values**

**To deposit an amount**

**To withdraw an amount after checking balance**

**To display the name and balance**

**Answer:**

import java.util.Scanner;

public class Account

{

String name, accountNumber, typeOfAccount;

double balance;

// constructor

Account(String name, String accountNumber, String typeOfAccount, double balance)

{

this.name = name;

this.accountNumber = accountNumber;

this.typeOfAccount = typeOfAccount;

this.balance = balance;

}

// method to deposit

double deposit(Scanner khoj)

{

System.out.println("Enter the following credentials: ");

System.out.print("Deposit Amount(in double): ");

double amount = khoj.nextDouble();

this.balance = this.balance + amount;

System.out.println("You have deposited " + amount + " in your account");

System.out.println("Total Balance you have in your account: " + this.balance);

return this.balance;

}

// method to withdraw

double withdraw(Scanner khoj)

{

System.out.println("Enter the following credentials: ");

System.out.print("Withdraw Amount(in double): ");

double withdraw = khoj.nextDouble();

if (withdraw > this.balance) {

System.out.println("Sorry, you don't have enough balance to withdraw");

}

else

{

this.balance = this.balance - withdraw;

System.out.println("You have withdrawen " +withdraw + " from your account.");

}

System.out.println("Total Balance you have in your account: " +this.balance);

return this.balance;

}

// method to display

void display() {

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Your Statement\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Account Holder's Name: " + this.name);

System.out.println("Total Balance you have in your account: " +this.balance);

System.out.println("Thank You for visiting us!! ");

}

public static void main(String[] args) {

String name, accountNumber, typeOfAccount;

double balance;

Scanner khoj = new Scanner(System.in);

System.out.println("Enter the following credentials: ");

System.out.print("Account Holder's Name(in string): ");

name = khoj.next();

System.out.print("Account Number(in string): ");

accountNumber = khoj.next();

System.out.print("Account Type(in string): ");

typeOfAccount = khoj.next();

System.out.print("Initial Balance(in double): ");

balance = khoj.nextDouble();

// creating objects

Account obj1 = new Account(name, accountNumber, typeOfAccount, balance);

obj1.deposit(khoj);

System.out.println("==========================================");

obj1.withdraw(khoj);

System.out.println("==========================================");

obj1.display();

}

}

3. **Define a class Student as described below:**

**Instance variables:**

**name, age, marks in three subjects (m1, m2, m3), maximum and average.**

**Methods:**

**i. A parameterized constructor to initialize the instance variables.**

**ii. To accept the details of a student.**

**iii. To compute the average and minimum out of the three marks.**

**iv. To display the name, age, marks in the three subjects, minimum and average.**

**v. Write a main method to create an object of the class and call the above methods.**

**Answer:**

import java.util.Scanner;

public class StudentData {

String name;

int age;

String subjects[];

double marks[];

double maximum;

double average;

// constructor

StudentData(String name, int age, int len) {

this.name = name;

this.age = age;

this.subjects = new String[len];

this.marks = new double[len];

}

// accepting details

void setMarks(Scanner sc) {

System.out.println("Enter the following credentials: ");

for (int i = 0; i < subjects.length; i++) {

System.out.print("Subject [" + (i + 1) + "](string): ");

subjects[i] = sc.next();

System.out.print("Marks in Subject [" + (i + 1) + "](double):");

marks[i] = sc.nextDouble();

}

}

// methods to get total

double getTotal() {

double sum = 0;

for (int i = 0; i < subjects.length; i++) {

sum = sum + marks[i];

}

return sum;

}

// method to get average

double getAverage(double sum) {

double average = sum / subjects.length;

return average;

}

// method to get minimum

double getMinimum() {

double minimum = marks[0];

for (int i = 1; i < marks.length; i++) {

if (marks[i] < minimum) {

minimum = marks[i];

}

}

return minimum;

}

// method to get minimum subject

String getMinimumSubject() {

String subject = null;

double minimum = marks[0];

for (int i = 1; i < marks.length; i++) {

if (marks[i] < minimum) {

subject = subjects[i];

}

}

return subject;

}

// method to display details

public void display() {

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Students Records\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

System.out.println("Name of the student: " + this.name);

System.out.println("Age of the student: " + this.age);

System.out.println("=====================================================");

System.out.println("Subjects\t \tMarks");

for (int i = 0; i < this.marks.length; i++) {

System.out.println(this.subjects[i] + "\t\t\t" + this.marks[i]);

}

double total = this.getTotal();

double average = this.getAverage(total);

System.out.println("====================================================");

System.out.println("Total marks obtained by Student: " + total);

System.out.println("Average marks of the Student: " + average);

System.out.println("Minimum marks obtained by the Student in " +getMinimumSubject()+ " with marks " + getMinimum());

}

public static void main(String[] args) {

Scanner khoj = new Scanner(System.in);

String name;

int age;

int len;

System.out.println("Enter the following instances: ");

System.out.print("Name(String): ");

name = khoj.next();

System.out.print("Age(int): ");

age = khoj.nextInt();

System.out.print("Number of subjects(int): ");

len = khoj.nextInt();

// creating object

StudentData std1 = new StudentData(name, age, len);

std1.setMarks(sc);

std1.display();

}

}

**4. Define a constructor for ShowDate class that initializes the ShowDate objects with given**

**initial values. In case the initial values are not provided, it should initialize the object with**

**default 0 values.**

**Answer:**

**i**mport java.util.Scanner;

public class Date {

int age;

// parameterized constructor

Date(int age) {

this.age = age;

}

// default constructor

Date() {

this.age = 0;

}

// method to display

void display() {

System.out.println("Display Age: " + this.age);

}

public static void main(String[] args) {

Scanner khoj = new Scanner(System.in);

System.out.println("Enter the following credentials: ");

System.out.print("Age(int): ");

int age = khoj.nextInt();

System.out.println("=========================================");

Date obj1 = new Date(age);

obj1.display();

System.out.println("=========================================");

Date obj2 = new Date();

obj2.display();

}

}

**5. Write a class Circle with a parameterized constructor. If no parameters are passed then its default constructor should be invoked parameterized constructor with default values. A circle is defined using radius and circumference.**

**Answer:**

import java.util.Scanner;

public class CircleMath {

static double pi = 3.141592654;

double radius;

// default constructor

CircleMath(){

this.radius = 2;

}

// parameterized constructor

CircleMath(double radius){

this.radius = radius;

}

// method to calculate circumference

public double calcCircumference(){

double circumference = 2\*(pi)\*this.radius;

return circumference;

}

// main function

public static void main(String[] args) {

Scanner khoj = new Scanner(System.in);

System.out.println("Enter the radius of a circle: ");

System.out.print("Radius: ");

double r = khoj.nextDouble();

// creating objects

CircleMath obj1 = new CircleMath(r);

double circumference = obj1.calcCircumference();

System.out.println("The circumference of a circle with radius " +r+ " is: " +circumference

);

// another object

CircleMath obj2 = new CircleMath();

circumference = obj2.calcCircumference();

System.out.println("The circumference of a circle with default radius is: " +circumference);

}

}

**6. Write a program to implement a Book class that stores the details of a book namely, bookcode, name of the book, name of the author(s) and price. The class has methods to display any of the details individually.**

**Answer:**

import java.util.Scanner;

public class BookDetail {

String bookCode;

String bookName;

String authorName;

double price;

// constructor

BookDetail(String bookCode, String bookName, String authorName, double price) {

this.bookCode = bookCode;

this.bookName = bookName;

this.authorName = authorName;

this.price = price;

}

// method to display book code

void displayBookCode() {

System.out.println("Book code is " + this.bookCode);

}

// method to display book name

void displayBookName() {

System.out.println("Book name is " + this.bookName);

}

// method to display book author name

void displayAuthorName() {

System.out.println("Book author name is " + this.authorName);

}

// method to display book price

void displayPrice() {

System.out.println("Book price is " + this.price);

}

// main function

public static void main(String[] args) {

Scanner khoj = new Scanner(System.in);

System.out.println("Enter the following credentials: ");

System.out.print("Book Code(in string): ");

String bookCode = khoj.nextLine();

System.out.print("Book Name(in string): ");

String bookName = khoj.nextLine();

System.out.print("Book Author Name(in string): ");

String authorName = khoj.nextLine();

System.out.print("Book Price(in string): ");

double price = khoj.nextDouble();

System.out.println("==============================================");

// creating objects

BookDetail type1 = new BookDetail(bookCode, bookName, authorName, price);

type1.displayBookCode();

System.out.println("==============================================");

type1.displayBookName();

System.out.println("==============================================");

type1.displayAuthorName();

System.out.println("==============================================");

type1.displayPrice();

System.out.println("==============================================");

}

}

**7. Implement a class called Dimension based on the following information:**

**Constructors**

**Dimension(double length, double width, double height)**

**Dimension(double side)**

**Methods**

**double volume() // length\*width\*height**

**double area() // 2\*(length\*width+width\*height+height\*length)**

**Make all the instance variables private so that they can be accessed only by the methods defined within the class. Make the methods public. Test your program.**

**Answer:**

import java.util.Scanner;

public class AllDimension {

private double length;

private double width;

private double height;

private double side;

// constructor

AllDimension(double length, double width, double height) {

this.length = length;

this.width = width;

this.height = height;

}

AllDimension(double side) {

this.length = side;

this.width = side;

this.height = side;

}

// method to calculate volume

public double calcVolume() {

double volume = this.length \* this.width \* this.height;

return volume;

}

// method to calculate area

public double calcArea() {

double area = 2 \* (this.length \* this.width + this.width \* this.height + this.height \* this.length);

return area;

}

// main function

public static void main(String[] args) {

Scanner khoj = new Scanner(System.in);

System.out.println("Enter the folloing instances: ");

System.out.print("Length: ");

double l = khoj.nextDouble();

System.out.print("Width: ");

double w = khoj.nextDouble();

System.out.print("Height: ");

double h = khoj.nextDouble();

System.out.print("Side: ");

double s = khoj.nextDouble();

// creating object

AllDimension cuboid1 = new AllDimension(l, w, h);

System.out.println("The volume of cuboid having length, width, height of " +l+", " +w+", " +h+" respectively is " +cuboid1.calcVolume());

System.out.println("The area of cuboid having length, width, height of " +l+", " +w+", " +h+" respectively is " +cuboid1.calcArea());

AllDimension cube1 = new AllDimension(s);

System.out.println("The volume of cube having side of " +s+" is " +cube1.calcVolume());

System.out.println("The area of cube having side of " +s+" is " +cube1.calcArea());

}

}

**8. Modify the implementation of area() given in the previous question using private**

**methods, faceArea(), topArea() and sideArea(). [Often private methods are helping methods that public methods use, but are not to be used outside the class.] Test your program.**

**Answer:**

import java.util.Scanner;

public class AreaDimension {

private double length;

private double width;

private double height;

private double side;

// constructor

AreaDimension(double length, double width, double height) {

this.length = length;

this.width = width;

this.height = height;

}

AreaDimension(double side) {

this.length = side;

this.width = side;

this.height = side;

}

// method to calculate volume

public double calcVolume() {

double volume = this.length \* this.width \* this.height;

return volume;

}

// method to calculate face area

private double faceArea() {

double faceArea = this.width \* this.height;

return faceArea;

}

// method to calculate top area

private double topArea() {

double topArea = this.length \* this.width;

return topArea;

}

// method to calculate side area

private double sideArea() {

double sideArea = this.length \* this.height;

return sideArea;

}

// method to calculate area

public double calcArea() {

double area = 2 \* (faceArea() + topArea() + sideArea());

return area;

}

// main function

public static void main(String[] args) {

Scanner khoj = new Scanner(System.in);

System.out.println("Enter the folloing instances: ");

System.out.print("Length: ");

double l = khoj.nextDouble();

System.out.print("Width: ");

double w = khoj.nextDouble();

System.out.print("Height: ");

double h = khoj.nextDouble();

System.out.print("Side: ");

double s = khoj.nextDouble();

// creating object

AreaDimension cuboid1 = new AreaDimension(l, w, h);

System.out.println("The volume of cuboid having length, width, height of " + l + ", " + w + ", " + h + " respectively is " + cuboid1.calcVolume());

System.out.println("The area of cuboid having length, width, height of " + l + ", " + w + ", " + h+ " respectively is " + cuboid1.calcArea());

AreaDimension cube1 = new AreaDimension(s);

System.out.println("The volume of cube having side of " + s + " is " +cube1.calcVolume());

System.out.println("The area of cube having side of " + s + " is " + cube1.calcArea());

}

}

**9. Add a new constructor to the Dimension class created in question 1 as**

**Dimension(Dimension dim)**

**This constructor creates a new Dimension object with identical dimensions as the old**

**Dimension object. The old object is not changed.**

**Answer:**

import java.util.Scanner;

public class Dimension3 {

private double length;

private double width;

private double height;

private double side;

// constructor

Dimension3(double length, double width, double height) {

this.length = length;

this.width = width;

this.height = height;

}

Dimension3(double side) {

this.length = side;

this.width = side;

this.height = side;

}

Dimension3(Dimension3 dim) {

this.length = dim.length;

this.width = dim.width;

this.height = dim.height;

}

// method to calculate volume

public double calcVolume() {

double volume = this.length \* this.width \* this.height;

return volume;

}

// method to calculate area

public double calcArea() {

double area = 2 \* (this.length \* this.width + this.width \* this.height + this.height \* this.length);

return area;

}

// main function

public static void main(String[] args) {

Scanner khoj = new Scanner(System.in);

System.out.println("Enter the folloing instances: ");

System.out.print("Length: ");

double l = khoj.nextDouble();

System.out.print("Width: ");

double w = khoj.nextDouble();

System.out.print("Height: ");

double h = khoj.nextDouble();

System.out.print("Side: ");

double s = khoj.nextDouble();

// creating object

Dimension3 cuboid1 = new Dimension3(l, w, h);

System.out.println("The volume of cuboid1 having length, width, height of " + l + ", " + w + ", " + h

+ " respectively is " + cuboid1.calcVolume());

System.out.println("The area of cuboid1 having length, width, height of " + l + ", " + w + ", " + h

+ " respectively is " + cuboid1.calcArea());

Dimension3 cube1 = new Dimension3(s);

System.out.println("The volume of cube1 having side of " + s + " is " + cube1.calcVolume());

System.out.println("The area of cube1 having side of " + s + " is " + cube1.calcArea());

System.out.println("========================================================================================");

Dimension3 cuboid2 = new Dimension3(cuboid1);

System.out.println("The volume of cuboid2 having length, width, height of " + l + ", " + w + ", " + h

+ " respectively is " + cuboid2.calcVolume());

System.out.println("The area of cuboid2 having length, width, height of " + l + ", " + w + ", " + h + " respectively is " + cuboid2.calcArea());

Dimension3 cube2 = new Dimension3(cube1);

System.out.println("The volume of cube2 having side of " + s + " is " + cube2.calcVolume());

System.out.println("The area of cube2 having side of " + s + " is " + cube2.calcArea());

}

}

**10. Design a class to overload a function num\_calc() as follows:**

**i. void num\_calc (int num) with one integer argument, computes the square of integer**

**argument.**

**ii. void num\_calc (int a, int b) with two integer arguments. It computes the product of**

**integer arguments.**

**iii. void num\_calc (String s1, String s2) with two string arguments, which concatenates**

**and prints the combined strings.**

**Answer:**

import java.util.Scanner;

public class FunctionOverLoad {

int num;

int a, b;

String s1, s2;

// method

public void num\_calc(int num) {

int sqr = num \* num;

System.out.println("The square of " +num+ " is " +sqr);

}

public void num\_calc(int a, int b) {

int product = a \* b;

System.out.println("The product of " +a+ " & " +b+ " is " +product);

}

public void num\_calc(String s1, String s2) {

String concatString = s1.concat(s2);

System.out.println("The concatenation of " +s1+ " & " +s2+ " is " +concatString);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter the following instances: ");

System.out.print("num(int): ");

int num = sc.nextInt();

System.out.print("a(int): ");

int a = sc.nextInt();

System.out.print("b(int): ");

int b = sc.nextInt();

System.out.print("s1(String): ");

String s1 = sc.next();

System.out.print("s2(String): ");

String s2 = sc.next();

// creating object

FunctionOverLoad obj1 = new FunctionOverLoad();

obj1.num\_calc(num);

obj1.num\_calc(a, b);

obj1.num\_calc(s1, s2);

}

}

**11. Write a program that uses the function power() to raise a number m to power n. The**

**function takes integer values for m and n and returns the result correctly. Use a default**

**value of 2 for n to make the function calculate squares when this argument is omitted.**

**Write a function main() to pass the value of m and n and display the calculated result.**

**Answer:**

import java.util.Scanner;

public class PowerFunction {

int a, b;

// parameterized constructor

PowerFunction(int a, int b) {

this.a = a;

this.b = b;

}

// default constructor

PowerFunction(int a) {

this.a = a;

this.b = 2;

}

// method to find power

public int power() {

int power = (int) Math.pow(this.a, this.b);

return power;

}

// main method

public static void main(String[] args) {

Scanner khoj = new Scanner(System.in);

System.out.println("Enter the following credentials: ");

System.out.print("a: ");

int a = khoj.nextInt();

System.out.print("b: ");

int b = khoj.nextInt();

System.out.println("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Printing\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

// creating objects

PowerFunction obj1 = new PowerFunction(a, b);

int power = obj1.power();

System.out.println("The result of 'b raised to power of a' is " + power);

System.out.println("================================================");

PowerFunction obj2 = new PowerFunction(a);

power = obj2.power();

System.out.println("The result of 'default b raised to power of a' is " + power);

}

}