1.1) Write a program to list all, even numbers less than or equal to the number n. Take the value of n as input from the user.

package basics;

import java.util.Scanner;

public class Even {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a value for n: ");

int n = sc.nextInt();

if (n >= 0) {

System.out.println("Even numbers less than or equal to " + n + ":");

for (int i = 2; i <= n; i += 2) {

System.out.print(i +" ");

}

} else {

System.out.println("Please enter a non-negative integer.");

}

sc.close();

}

}

1.2) Define a class Rectangle with its length and breadth. Follow the below steps,

a. Provide appropriate constructor(s), which gives facility of constructing Rectangle object with

default values of length and breadth as 0 or passing value of length and breadth externally to

constructor.

b. Provide appropriate accessor & mutator methods to Rectangle class.

c. Provide methods to calculate area & to display all information of Rectangle.

d. Design different class TestRectangle class in a separate source file, which will contain main

method. From this main method, create a Rectangle object by taking all necessary information

from the user and calculate respective area of rectangle objects and display it.

package basics;

public class Rectangle {

private double length;

private double breadth;

public Rectangle() {

this.length = 0;

this.breadth = 0;

}

public Rectangle(double length, double breadth) {

this.length = length;

this.breadth = breadth;

}

public double getArea() {

return length \* breadth;

}

public double getLength() {

return length;

}

public void setLength(double length) {

this.length = length;

}

public double getBreadth() {

return breadth;

}

public void setBreadth(double breadth) {

this.breadth = breadth;

}

public void display() {

System.out.println("Length: " + length);

System.out.println("Breadth: " + breadth);

System.out.println("Area: " + getArea());

}

}

package basics;

import java.util.Scanner;

public class TestRectangle {

public static void main(String[] args) {

Scanner scanner= new Scanner(System.in);

System.out.println("enter number a length");

double length=scanner.nextDouble();

System.out.println("enter number a breadth");

double breadth=scanner.nextDouble();

Rectangle rc=new Rectangle(length,breadth);

rc.display();

}

}

1.3) 1.3 Create a class Book which describes its book\_title and book\_price. Follow the below steps,

a. Use getter and setter methods to get & set the Books description.

b. Implement createBooks and showBooks methods to create n objects of Book in an array.

c. Display the books along with its description as follows.

d. Note: createBooks & showBooks should not be member functions of Book class.

package basics;

import java.util.ArrayList;

import java.util.Scanner;

public class Book {

private String bookTitle;

private double bookPrice;

public String getBookTitle() {

return bookTitle;

}

public void setBookTitle(String bookTitle) {

this.bookTitle = bookTitle;

}

public double getBookPrice() {

return bookPrice;

}

public void setBookPrice(double bookPrice) {

this.bookPrice = bookPrice;

}

public static ArrayList<Book> createBooks(int n) {

ArrayList<Book> bookList = new ArrayList<>();

Scanner sc = new Scanner(System.in);

for (int i = 0; i < n; i++) {

Book book = new Book();

System.out.println("Enter the title of book " + (i + 1) + ": ");

String title = sc.nextLine();

book.setBookTitle(title);

System.out.println("Enter the price of book " + (i + 1) + ": ");

double price = sc.nextDouble();

sc.nextLine();

book.setBookPrice(price);

bookList.add(book);

}

return bookList;

}

public static void showBooks(ArrayList<Book> bookList){

System.out.println("Book title Price");

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

for (Book book : bookList) {

System.out.printf("%-25s Rs %.2f\n", book.getBookTitle(), book.getBookPrice());

}

}

}

package basics;

import java.util.ArrayList;

import java.util.Scanner;

public class BookDemo extends Book {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("How many books would you like to create?");

int n = sc.nextInt();

sc.nextLine();

ArrayList<Book> books = createBooks(n);

showBooks(books);

}

}

1.4) Modify the program, which is created in sub problem 1.2 as follows,

a. The class has attributes length and width, each of which defaults to 1.

b. It should have member functions that calculate the perimeter and area of the rectangle.

c. It should have set and get functions for both length and width.

d. The set functions should verify that length and width are each floating-point number larger than 0.0 and less than 20.0.

package basics;

public class Rectangle {

private double length=1.0;

private double width=1.0;

public Rectangle(double length, double width) {

this.length = length;

this.width = width;

}

public double getLength() {

return length;

}

public void setLength(double length) {

if (length > 0.0 && length < 20.0) {

this.length = length;

} else {

System.out.println("Length must be greater than 0.0 and less than 20.0.");

}

}

public double getWidth() {

return width;

}

public void setWidth(double width) {

if (width > 0.0 && width < 20.0) {

this.width = width;

} else {

System.out.println("Width must be greater than 0.0 and less than 20.0.");

}

}

public double calculateArea() {

return length \* width;

}

public double calculatePerimeter() {

return 2 \* (length + width);

}

public void displayInfo() {

System.out.println("Length: " + length);

System.out.println("Width: " + width);

System.out.println("Area: " + calculateArea());

System.out.println("Perimeter: " + calculatePerimeter());

}

}

package basics;

import java.util.Scanner;

public class TestRectangle extends Rectangle {

public TestRectangle(double length, double width) {

super(length, width);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the length of the rectangle: ");

double length = sc.nextDouble();

System.out.print("Enter the width of the rectangle: ");

double width = sc.nextDouble();

Rectangle rectangle= new Rectangle(length, width);

rectangle.displayInfo();

sc.close();

}

}

1.5) 1.5 Create a class Date with day, month, and year attributes for manipulating dates.

Follow the below steps,

a. Provide a constructor that enables an object of this class to be initialized when it is declared (You can select any default values for the day, month & year, e.g., your birth date).

b. Provide the necessary functionality to perform error checking on the initializer values for data

members day, month, and year.

c. Provide a member function to add an integer in a date to obtain a new date.

d. Design separate class Employee which will have following information. Refer below table.

e. Provide appropriate constructor(s)& methods to this class. Provide main function which will create 5 objects of Employee class and display employee information.

package basics;

import java.util.Date;

import java.util.Scanner;

public class Day {

private int day;

private int month;

private int year;

public void Date(int day, int month, int year) {

if (isValidDate(day, month, year)) {

this.day = day;

this.month = month;

this.year = year;

} else {

System.out.println("Invalid date. Setting default date.");

this.day = 1;

this.month = 1;

this.year = 2000; }

}

private boolean isValidDate(int day, int month, int year) {

if (year < 0 || month < 1 || month > 12) {

return false;

}

int[] daysInMonth = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

if (isLeapYear(year)) {

daysInMonth[1] = 29;

}

return day > 0 && day <= daysInMonth[month - 1];

}

private boolean isLeapYear(int year) {

return (year % 4 == 0 && year % 100 != 0) || (year % 400 == 0);

}

public void addDays(int days) {

while (days > 0) {

int[] daysInMonth = {31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31};

if (isLeapYear(year)) {

daysInMonth[1] = 29;

}

if (day + days > daysInMonth[month - 1]) {

days -= (daysInMonth[month - 1] - day + 1);

day = 1;

month++;

if (month > 12) {

month = 1;

year++;

}

} else {

day += days;

days = 0;

}

}

}

@Override

public String toString() {

return String.format("%02d/%02d/%04d", day, month, year);

}

}

class Employee {

private int employeeNumber;

private String employeeName;

private Date joiningDate;

public Employee(int employeeNumber, String employeeName, Date joiningDate) {

this.employeeNumber = employeeNumber;

this.employeeName = employeeName;

this.joiningDate = joiningDate;

}

public void displayEmployeeInfo() {

System.out.println("Employee Number: " + employeeNumber);

System.out.println("Employee Name: " + employeeName);

System.out.println("Joining Date: " + joiningDate);

System.out.println();

}

}

package basics;

import java.util.Date;

import java.util.Scanner;

public class DayDemo extends Day {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

Employee[] employees = new Employee[5];

for (int i = 0; i < 5; i++) {

System.out.println("Enter details for employee " + (i + 1) + ":");

System.out.print("Employee Number: ");

int employeeNumber = sc.nextInt();

sc.nextLine();

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

String employeeName = sc.nextLine();

System.out.print("Joining Date (dd mm yyyy): ");

int day = sc.nextInt();

int month = sc.nextInt();

int year = sc.nextInt();

Date joiningDate = new Date(day, month, year);

employees[i] = new Employee(employeeNumber, employeeName, joiningDate);

System.out.println();

}

for (Employee employee : employees) {

employee.displayEmployeeInfo();

}

}

}

2) Design a Java program to manage different types of vehicles (Car, Motorcycle, Truck) with the following features:

Vehicle Class:

package basics;

public class Vehicle {

private String manufacturer;

private String model;

private int year;

public Vehicle() {}

public Vehicle(String manufacturer, String model, int year) {

this.manufacturer = manufacturer;

this.model = model;

this.year = year;

}

public String getManufacturer() {

return manufacturer;

}

public void setManufacturer(String manufacturer) {

this.manufacturer = manufacturer;

}

public String getModel() {

return model;

}

public void setModel(String model) {

this.model = model;

}

public int getYear() {

return year;

}

public void setYear(int year) {

this.year = year;

}

public void displayDetails() {

System.out.println("Manufacturer: " + manufacturer);

System.out.println("Model: " + model);

System.out.println("Year: " + year);

}

}

Subclasses:

Car Class:

package basics;

public class Car extends Vehicle {

private int seatingCapacity;

public Car() {}

public Car(String manufacturer, String model, int year, int seatingCapacity) {

super(manufacturer, model, year);

this.seatingCapacity = seatingCapacity;

}

public int getSeatingCapacity() {

return seatingCapacity;

}

public void setSeatingCapacity(int seatingCapacity) {

this.seatingCapacity = seatingCapacity;

}

public void displayDetails() {

super.displayDetails();

System.out.println("Seating Capacity: " + seatingCapacity);

}

public void accelerate() {

System.out.println("The car is accelerating.");

}

public void brake() {

System.out.println("The car is braking.");

}

}

Motorcycle Class:

package basics;

public class Motorcycle extends Vehicle {

private double engineCapacity;

public Motorcycle() {}

public Motorcycle(String manufacturer, String model, int year, double engineCapacity) {

super(manufacturer, model, year);

this.engineCapacity = engineCapacity;

}

public double getEngineCapacity() {

return engineCapacity;

}

public void setEngineCapacity(double engineCapacity) {

this.engineCapacity = engineCapacity;

}

public void displayDetails() {

super.displayDetails();

System.out.println("Engine Capacity: " + engineCapacity + "cc");

}

public void startEngine() {

System.out.println("The motorcycle engine is starting.");

}

public void stopEngine() {

System.out.println("The motorcycle engine is stopping.");

}

}

Truck Class:

package basics;

public class Truck extends Vehicle {

private double cargoCapacity;

public Truck() {}

public Truck(String manufacturer, String model, int year, double cargoCapacity) {

super(manufacturer, model, year);

this.cargoCapacity = cargoCapacity;

}

public double getCargoCapacity() {

return cargoCapacity;

}

public void setCargoCapacity(double cargoCapacity) {

this.cargoCapacity = cargoCapacity;

}

public void displayDetails() {

super.displayDetails();

System.out.println("Cargo Capacity: " + cargoCapacity + " tons");

}

public void loadCargo() {

System.out.println("Loading cargo into the truck.");

}

public void unloadCargo() {

System.out.println("Unloading cargo from the truck.");

}

}

Main Class:

package basics;

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

//Creating and using a Car object

System.out.println("Enter details for Car:");

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

String carManufacturer = scanner.nextLine();

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

String carModel = scanner.nextLine();

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

int carYear = scanner.nextInt();

System.out.print("Seating Capacity: ");

int carSeatingCapacity = scanner.nextInt();

scanner.nextLine();

Car car = new Car(carManufacturer, carModel, carYear, carSeatingCapacity);

car.displayDetails();

car.accelerate();

car.brake();

// Creating and using a Motorcycle object

System.out.println("\nEnter details for Motorcycle:");

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

String motoManufacturer = scanner.nextLine();

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

String motoModel = scanner.nextLine();

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

int motoYear = scanner.nextInt();

System.out.print("Engine Capacity (cc): ");

double engineCapacity = scanner.nextDouble();

scanner.nextLine();

Motorcycle motorcycle = new Motorcycle(motoManufacturer, motoModel, motoYear, engineCapacity);

motorcycle.displayDetails();

motorcycle.startEngine();

motorcycle.stopEngine();

// Creating and using a Truck object

System.out.println("\nEnter details for Truck:");

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

String truckManufacturer = scanner.nextLine();

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

String truckModel = scanner.nextLine();

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

int truckYear = scanner.nextInt();

System.out.print("Cargo Capacity (tons): ");

double cargoCapacity = scanner.nextDouble();

Truck truck = new Truck(truckManufacturer, truckModel, truckYear, cargoCapacity);

truck.displayDetails();

truck.loadCargo();

truck.unloadCargo();

scanner.close();

}

}

3) Design a Java program to calculate areas of different shapes (Circle, Rectangle, Triangle):

• Define an abstract class Shape with an abstract method calculateArea().

• Implement classes for each shape extending Shape and provide necessary attributes (radius and

sideLength for circle, rectangle, and triangle respectively).

• Use abstraction to ensure that each shape class implements its own logic to calculate the area based on its specific attributes.

package assignment;

abstract class Shape {

public abstract double calculateArea();

}

class Circle extends Shape {

private double radius;

public Circle(double radius) {

this.radius = radius;

}

@Override

public double calculateArea() {

return Math.PI \* radius \* radius;

}

}

class Rectangle extends Shape {

private double width;

private double height;

public Rectangle(double width, double height) {

this.width = width;

this.height = height;

}

@Override

public double calculateArea() {

return width \* height;

}

}

class Triangle extends Shape {

private double base;

private double height;

public Triangle(double base, double height) {

this.base = base;

this.height = height;

}

@Override

public double calculateArea() {

return 0.5 \* base \* height;

}

}

package assignment;

public class Main {

public static void main(String[] args) {

// Creating instances of different shapes

Shape circle = new Circle(5.0);

Shape rectangle = new Rectangle(4.0, 6.0);

Shape triangle = new Triangle(3.0, 7.0);

// Calculating and printing areas

System.out.println("Area of Circle: " + circle.calculateArea());

System.out.println("Area of Rectangle: " + rectangle.calculateArea());

System.out.println("Area of Triangle: " + triangle.calculateArea());

}

}

4) VastMindz Company is creating a performance rating system to calculate the performance of each employee. Design the below classes to achieve the same.

package assignment;

public class EmployeePerformance {

private String name;

private int point;

public EmployeePerformance(String name,int point) {

this.name=name;

this.point=point;

}

public String getName() {

return name;

}

public void setname(String name) {

this.name=name;

}

public int getPoint() {

return point;

}

public void setPoint(int point) {

this.point=point;

}

class PerformanceRating{

public static final int OUTSTANDING=5;

public static final int GOOD=4;

public static final int AVERAGE=3;

public static final int POOR=2;

public static int caluclatePerformance(int point) {

if(point>=80 && point<=100) {

return OUTSTANDING;

}

else if(point>=60 && point<=79) {

return GOOD;

}

else if(point>=50 && point <=59) {

return AVERAGE;

}

else if(point>=1 && point<=49) {

return POOR;

}

else {

throw new IllegalArgumentException("Invalid ponit value:" +point);

}

}

}

}

package assignment;

import assignment.EmployeePerformance.PerformanceRating;

public class PerformanceCalculator{

private static int employeeCount=0;

public static void main(String[] args) {

EmployeePerformance employee1=new EmployeePerformance("Oliver", 95);

EmployeePerformance employee2=new EmployeePerformance("jayden", 45);

EmployeePerformance employee3=new EmployeePerformance("Daniel", 75);

EmployeePerformance [] employees = {employee1,employee2,employee3};

employeeCount = employees.length;

System.out.println("Toatal Number of Employee: " +employeeCount +" and Their ratings are");

for(EmployeePerformance employee : employees) {

int rating = PerformanceRating.caluclatePerformance(employee.getPoint());

System.out.println(employee.getName() +" has Performed with a Rating "+ rating);

}

}

}