

**Object Oriented Programming**

**Lab Task 5**

**SUBMITTED BY:**

Hasaan Ahmad SP22-BSE-017

**SUBMITTED TO: Sir Muzaffar Iqbal**

**Activity 1:**

package LAB5;

class studentRecord {

    private String degree;

    public studentRecord() {

    }

    public void setDegree(String deg) {

        degree = deg;

    }

    public String getDegree() {

        return degree;

    }

}

class employeeRecord {

    private int emp\_id;

    private double salary;

    public employeeRecord() {

    }

    public void setEmp\_id(int id) {

        emp\_id = id;

    }

    public int getEmp\_id() {

        return emp\_id;

    }

    public void setSalary(int sal) {

        salary = sal;

    }

    public double getSalary() {

        return salary;

    }

}

class Manager {

    private String title;

    private double dues;

    private employeeRecord emp;

    private studentRecord stu;

    public Manager(String t, double d, employeeRecord e, studentRecord s) {

        title = t;

        dues = d;

        emp = e;

        stu = s;

    }

    public void display() {

        System.out.println("Title is : " + title);

        System.out.println("Dues are : " + dues);

        System.out.println("Emplyoyee record is as under:");

        System.out.println("EmployeeId is : " +

                emp.getEmp\_id());

        System.out.println("EmployeeId is : " + emp.getSalary());

        System.out.println("Student record is as under: ");

        System.out.println("Degree is : " + stu.getDegree());

    }

}

public class Runner {

    public static void main(String args[]) {

        studentRecord s = new studentRecord();

        s.setDegree("MBA");

        employeeRecord e = new employeeRecord();

        e.setEmp\_id(1);

        e.setSalary(25000);

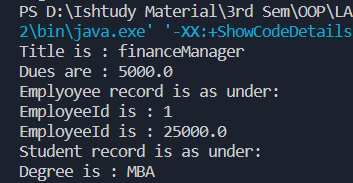
        Manager m1 = new Manager("financeManager", 5000, e, s);

        m1.display();

    }

}

**Output:**

****

**Activity 2:**

package LAB5;

class Date {

    private int day;

    private int month;

    private int year;

    public Date(int theMonth, int theDay, int theYear) {

        day = checkday(theDay);

        month = checkmonth(theMonth);

        year = theYear;

    }

    private int checkmonth(int testMonth) {

        if (testMonth > 0 && testMonth <= 12) {

            return testMonth;

        } else {

            System.out.println("Invalid month " + testMonth + " set to 1");

            return 1;

        }

    }

    private int checkday(int testDay) {

        int daysofmonth[] = { 0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };

        if (testDay > 0 && testDay <= daysofmonth[month]) {

            return testDay;

        } else if (month == 2 && testDay == 29 &&

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

            return testDay;

        } else {

            System.out.println("Invalid date " + testDay + " set to 1");

        }

        return 1;

    }

    public int getDay() {

        return day;

    }

    public int getMonth() {

        return month;

    }

    public int getYear() {

        return year;

    }

    public void display() {

        System.out.println(day + " " + month + " " + year);

    }

}

class employee {

    private String name;

    private String fname;

    private Date birthdate;

    private Date hiredate;

    employee() {

    }

    employee(String x, String y, Date birthofDate, Date dateofHire) {

        name = x;

        fname = y;

        birthdate = birthofDate;

        hiredate = dateofHire;

    }

    public void setname(String x) {

        name = x;

    }

    public String getname() {

        return name;

    }

    public void setfname(String x) {

        fname = x;

    }

    public String getfname() {

        return fname;

    }

    public void setbirthdate(Date b) {

        birthdate = b;

    }

    public Date getbirthdate() {

        return birthdate;

    }

    public void sethiredate(Date h) {

        hiredate = h;

    }

    public Date gethiredate() {

        return hiredate;

    }

    public void display() {

        System.out.println("Name: " + name + "  Father Name: " + fname);

        birthdate.display();

        hiredate.display();

    }

}

public class Runner1 {

    public static void main(String[] args) {

        Date b = new Date(1, 12, 1990);

        Date h = new Date(5, 6, 2016);

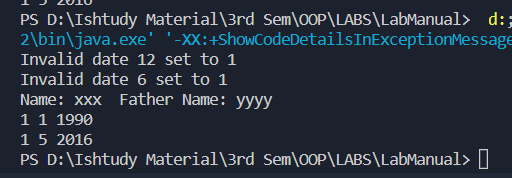
        employee e1 = new employee("xxx", "yyyy", b, h);

        e1.display();

    }

}

**Output:**

****

**Graded Lab Task 1:**

package LAB5;

import java.sql.PseudoColumnUsage;

class Adrress {

    private String street;

    private String city;

    private String house;

    private String code;

    public String getStreet() {

        return street;

    }

    public void setStreet(String street) {

        this.street = street;

    }

    public String getCity() {

        return city;

    }

    public void setCity(String city) {

        this.city = city;

    }

    public String getHouse() {

        return house;

    }

    public void setHouse(String house) {

        this.house = house;

    }

    public String getCode() {

        return code;

    }

    public void setCode(String code) {

        this.code = code;

    }

    public Adrress(String street, String city, String house, String code) {

        this.street = street;

        this.city = city;

        this.house = house;

        this.code = code;

    }

}

class Person {

    private String name;

    private String fname;

    private Adrress adrress;

    public Person(String name, String fname, Adrress adrress) {

        this.name = name;

        this.fname = fname;

        this.adrress = adrress;

    }

    public String getName() {

        return name;

    }

    public void setName(String name) {

        this.name = name;

    }

    public String getFname() {

        return fname;

    }

    public void setFname(String fname) {

        this.fname = fname;

    }

    public Adrress getAdrress() {

        return adrress;

    }

    public void setAdrress(Adrress adrress) {

        this.adrress = adrress;

    }

    void display() {

        System.out.println("Name: " + name);

        System.out.println("Father Name: " + fname);

        System.out.println("Street: " + adrress.getStreet());

        System.out.println("City: " + adrress.getCity());

        System.out.println("House: " + adrress.getHouse());

        System.out.println("Code: " + adrress.getCode());

    }

}

public class PersonRunner {

    public static void main(String[] args) {

        Adrress add1 = new Adrress("Street 5", "Islamabad", "B101", "44000");

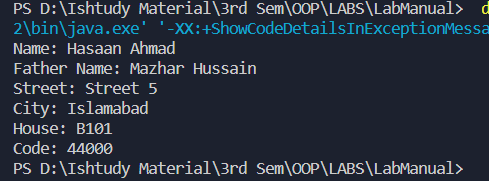
        Person Hasaan = new Person("Hasaan Ahmad", "Mazhar Hussain", add1);

        Hasaan.display();

    }

}

**Output:**

****

**Graded Lab Task 2:**

package LAB5;

class Book {

    private Person author;

    private String bookName;

    private String publisher;

    public Person getAuthor() {

        return author;

    }

    public void setAuthor(Person author) {

        this.author = author;

    }

    public String getBookName() {

        return bookName;

    }

    public void setBookName(String bookName) {

        this.bookName = bookName;

    }

    public String getPublisher() {

        return publisher;

    }

    public void setPublisher(String publisher) {

        this.publisher = publisher;

    }

    public Book(Person author, String bookName, String publisher) {

        this.author = author;

        this.bookName = bookName;

        this.publisher = publisher;

    }

    void display() {

        System.out.println("Book Name: " + bookName);

        System.out.println("Publisher: " + publisher);

        System.out.println("----------------Author's Information------------------");

        System.out.println("Author Name: " + author.getName());

        System.out.println("Author Father Name: " + author.getFname());

        System.out.println("Author Address: " + author.getAdrress().getStreet());

        System.out.println("Author Address: " + author.getAdrress().getCity());

        System.out.println("Author Address: " + author.getAdrress().getHouse());

        System.out.println("Author Address: " + author.getAdrress().getCode());

    }

}

public class BookRunner {

    public static void main(String[] args) {

        Adrress adrress = new Adrress("Street 1", "Islamabad", "B202", "40400");

        Person author = new Person("Hasaan Ahmad", "Mazhar Hussain", adrress);

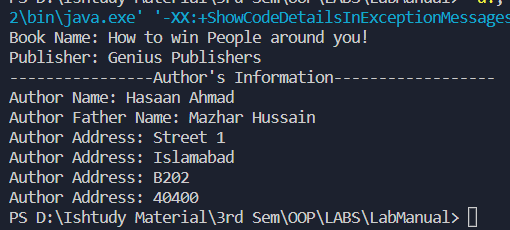
        Book book = new Book(author, "How to win People around you!", "Genius Publishers");

        book.display();

    }

}

**Output:**

****

**Graded Lab Task 3:**

package LAB5;

class point {

    private double xCord;

    private double yCord;

    public double getxCord() {

        return xCord;

    }

    public void setxCord(double xCord) {

        this.xCord = xCord;

    }

    public double getyCord() {

        return yCord;

    }

    public void setyCord(double yCord) {

        this.yCord = yCord;

    }

    public point(double xCord, double yCord) {

        this.xCord = xCord;

        this.yCord = yCord;

    }

}

class Line {

    private point p1;

    private point p2;

    public point getP1() {

        return p1;

    }

    public void setP1(point p1) {

        this.p1 = p1;

    }

    public point getP2() {

        return p2;

    }

    public void setP2(point p2) {

        this.p2 = p2;

    }

    public Line(point p1, point p2) {

        this.p1 = p1;

        this.p2 = p2;

    }

    public double getLength() {

        return Math.sqrt(Math.pow((p2.getxCord() - p1.getxCord()), 2) + Math.pow((p2.getyCord() - p1.getyCord()), 2));

    }

    void display() {

        System.out.println("Length of line is: " + getLength());

    }

}

public class PointRuner {

    public static void main(String[] args) {

        point p1 = new point(23.5, 12.4);

        point p2 = new point(45.4, 53.32);

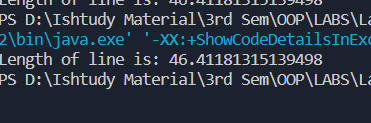
        Line l1 = new Line(p1, p2);

        l1.display();

    }

}

**Output:**

****

**Graded Lab Task 4:**

package LAB5;

class Pizza {

    private String size;

    private int cheeseToppings;

    private int pepperoniToppings;

    private int hamToppings;

    public Pizza(String size, int cheeseToppings, int pepperoniToppings, int hamToppings) {

        this.size = size;

        this.cheeseToppings = cheeseToppings;

        this.pepperoniToppings = pepperoniToppings;

        this.hamToppings = hamToppings;

    }

    public String getSize() {

        return size;

    }

    public void setSize(String size) {

        this.size = size;

    }

    public int getCheeseToppings() {

        return cheeseToppings;

    }

    public void setCheeseToppings(int cheeseToppings) {

        this.cheeseToppings = cheeseToppings;

    }

    public int getPepperoniToppings() {

        return pepperoniToppings;

    }

    public void setPepperoniToppings(int pepperoniToppings) {

        this.pepperoniToppings = pepperoniToppings;

    }

    public int getHamToppings() {

        return hamToppings;

    }

    public void setHamToppings(int hamToppings) {

        this.hamToppings = hamToppings;

    }

    public double calcCost() {

        double cost = 0.0;

        if (size.equalsIgnoreCase("small")) {

            cost = 10 + (2 \* (cheeseToppings + pepperoniToppings + hamToppings));

        } else if (size.equalsIgnoreCase("medium")) {

            cost = 12 + (2 \* (cheeseToppings + pepperoniToppings + hamToppings));

        } else if (size.equalsIgnoreCase("large")) {

            cost = 14 + (2 \* (cheeseToppings + pepperoniToppings + hamToppings));

        }

        return cost;

    }

    public String getDescription() {

        return "Size: " + size + ", Cheese Toppings: " + cheeseToppings + ", Pepperoni Toppings: " + pepperoniToppings

                + ", Ham Toppings: " + hamToppings;

    }

}

class PizzaOrder {

    private Pizza[] pizzas;

    private int numPizzas;

    public PizzaOrder() {

        pizzas = new Pizza[3];

        numPizzas = 0;

    }

    public void addPizza(Pizza pizza) {

        if (numPizzas < 3) {

            pizzas[numPizzas] = pizza;

            numPizzas++;

        } else {

            System.out.println("Maximum pizzas per order is 3.");

        }

    }

    public double calcTotal() {

        double totalCost = 0.0;

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

            totalCost += pizzas[i].calcCost();

        }

        return totalCost;

    }

}

public class PizzaRunner {

    public static void main(String[] args) {

        Pizza pizza1 = new Pizza("small", 1, 1, 1);

        Pizza pizza2 = new Pizza("medium", 2, 2, 2);

        Pizza pizza3 = new Pizza("large", 3, 3, 3);

        PizzaOrder order = new PizzaOrder();

        order.addPizza(pizza1);

        order.addPizza(pizza2);

        order.addPizza(pizza3);

        System.out.println(pizza1.getDescription());

        System.out.println("Cost: $" + pizza1.calcCost());

        System.out.println(pizza2.getDescription());

        System.out.println("Cost: $" + pizza2.calcCost());

        System.out.println(pizza3.getDescription());

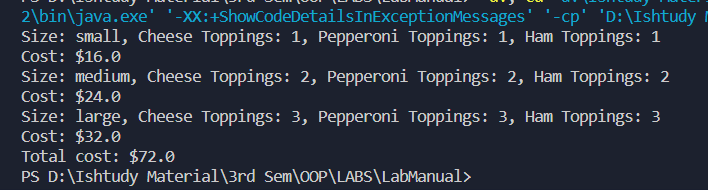
        System.out.println("Cost: $" + pizza3.calcCost());

        System.out.println("Total cost: $" + order.calcTotal());

    }

}

**Output:**

****