



OOP-Exercises - practice problems of Object oriented programming in java

Object Oriented Programming (COMSATS University Islamabad)

Practice Problems

1. Write a program to print the area and perimeter of a triangle having sides of 3, 4 and 5 units by creating a class named 'Triangle' without any parameter in its constructor.

```
class Triangle
{
    private int length, width, height;

    //Default Constructor
    Triangle()
    {
        Length=3; width=4; height=5;
    }

    // Parameterized Costructor////
    Triangle(int l, int w, int h)
    {
        Length=l; width=w; height=h;
    }

    int area()
    { return length*width*height;}

    int perimeter()
    { return length+width+height;}

    /////Setters/////
    Void setlength(int l)
    { length=l; }

    Void setwidth(int w)
    { width=w; }

    Void setheight(int h)
    { height=h; }

    /////Getters/////
    int getlength()
    { return length; }
```

```

    int getwidth()
    { return width; }

    int getheight()
    { return height; }

}
Public class labexercise
{
    Public Static void main( String ags...)
    {
        Triangle t=new Triangle();
        Triangle t1=new Triangle(5,6,7);

        System.out.println(" Area of Triangle= " +t.area());/// 60
        System.out.println(" Perimeter of Triangle= " +t.perimeter());

        System.out.println(" Area of Triangle= " +t1.area());
        System.out.println(" Perimeter of Triangle= " +t1.perimeter());

        t.setlength(12);

        System.out.println(" Area of Triangle= " +t.area());///240
        System.out.println(" Area of Triangle= " +t.area()+ " And its width is "+t.getwidth());

    }
}

```

2. Print the average of three numbers entered by user by creating a class named 'Average' having a method to calculate and print the average.
3. Print the sum, difference and product of two complex numbers by creating a class named 'Complex' with separate methods for each operation whose real and imaginary parts are entered by user.
4. Write a program by creating an 'Employee' class having the following methods and print the final salary.

- 1 - 'getInfo()' which takes the salary, number of hours of work per day of employee as parameter
- 2 - 'AddSal()' which adds \$10 to salary of the employee if it is less than \$500.
- 3 - 'AddWork()' which adds \$5 to salary of employee if the number of hours of work per day is more than 6 hours.

5. Create a class called 'Matrix' containing constructor that initializes the number of rows and number of columns of a new Matrix object. The Matrix class has the following information:
 - 1.number of rows of matrix
 - 2 number of columns of matrix
 - 3 - elements of matrix in the form of 2D array

The Matrix class has methods for each of the following:

- 1 - get the number of rows
- 2 - get the number of columns
- 3 - set the elements of the matrix at given position (i,j)
- 4 - adding two matrices. If the matrices are not addable, "Matrices cannot be added" will be displayed.
- 5 - multiplying the two matrices

6. Write a program that has variables to store Car data like; CarModel, CarName, CarPrice and CarOwner. The program should include functions to assign user defined values to the above mentioned variable and a display function to show the values. Write a main

that calls these functions. Now write another runner class that declares three Car objects and displays the data of all three.

7. A Student is an object in a university management System. Analyze the concept and identify the data members that a Student class should have. Also analyze the behavior of student in a university management System and identify the methods that should be included in Student class.
8. Create a class circle class with radius as data member. Create two constructors (no argument, and two arguments) and a method to calculate Circumference.

Activity 2:

9. Create a class Account class with balance as data member. Create two constructors (no argument, and two arguments) and methods to withdraw and deposit balance.

Create a class "Distance" with two constructors (no argument, and two argument), two data

Create a class "Distance" with two constructors (no argument, and two argument), two data

10. Write a class Time with three data members to store hr, min and seconds. Create two constructors and apply checks to set valid time. Also create display function which displays all data members.
11. Create a class " Distance" with two constructors (no argument, and two argument), two data
11. Implement Vehicle as outer and owner as the inner class, the vehicle class contains vehicle name, engine cc, model as data members. The inner class data members are owners name, CNIC number and phone contact of the owner.

Write down proper setters/ getters and constructors for both the classes.

1: Override the method of a class using anonymous inner class.

2: Pass an anonymous inner class as a method argument.

3: Implement the inner class as static first and then as non static nested class.

12. Create an Abstract class Student that contains a method take exam, implement the method in the child classes PhdStudent and GradStudent in which PhdStudent takes exam by giving his final defense presentation while the graduate student gives a written paper.

13. Create a class “Distance” with two constructors (no argument, and two argument), two data

14. members (feet and inches) . Create setter, getter and display method. Create a method that adds two Distance Objects and returns the added Distance Object.
15. Define a class called Fraction. This class is used to represent a ratio of two integers. Create two constructors, set, get and display function. Include an additional method, equals, that takes as input another Fraction and returns true if the two fractions are identical and false if they are not.
16. Create a SavingsAccount class. Use a static data member annualInterestRate to store the annual interest rate for each of the savers. Each member of the class contains a private data member savingsBalance indicating the amount the saver currently has on deposit. Provide member function calculateMonthlyInterest that calculates the monthly interest by multiplying the balance by annualInterestRate divided by 12; this interest should be added to savingsBalance. Provide a static member function modifyInterestRate that sets the static annualInterestRate to a new value. Write a driver program to test class SavingsAccount. Instantiate two different objects of class SavingsAccount, saver1 and saver2, with balances of \$2000.00 and \$3000.00, respectively. Set the annualInterestRate to 3 percent. Then calculate the monthly interest and print the new balances for each of the savers. Then set the annualInterestRate to 4 percent, calculate the next month's interest and print the new balances for each of the savers.

17. Assignment:

18. Create a class named Pizza that stores information about a single pizza. It should contain the following:
 - Private instance variables to store the size of the pizza (either small, medium, or large), the number of cheese toppings, the number of pepperoni toppings, and the number of ham toppings.

- Constructor(s) that set all of the instance variables.
- Public methods to get and set the instance variables.
- A public method named `calcCost()` that returns a double that is the cost of the pizza.
 Pizza cost is determined by:
 Small: \$10 + \$2 per topping
 Medium: \$12 + \$2 per topping
 Large: \$14 + \$2 per topping
- public method named `getDescription()` that returns a String containing the pizza size, quantity of each topping.

Write test code to create several pizzas and output their descriptions. For example, a large pizza with one cheese, one pepperoni and two ham toppings should cost a total of \$22. Now Create a `PizzaOrder` class that allows up to three pizzas to be saved in an order. Each pizza saved should be a `Pizza` object. Create a method `calcTotal()` that returns the cost of order. In the runner order two pizzas and return the total cost.

19.

20.

21. **CCS241 –Lab**
 Manual

22.

23. 29

24.

25.

26. **Activity 2:**

create a class "Distance"
with two constructors (no
argument, and two
argument), two data