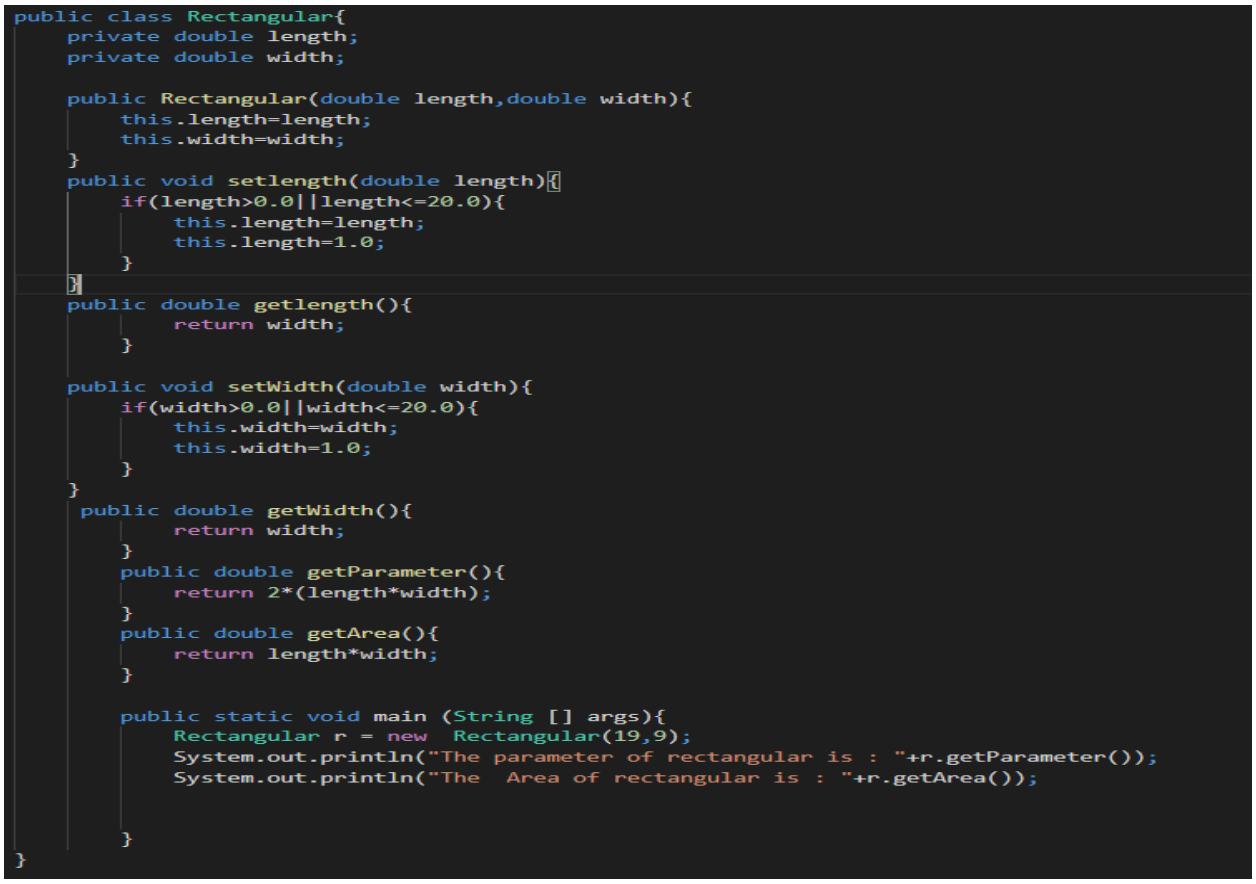
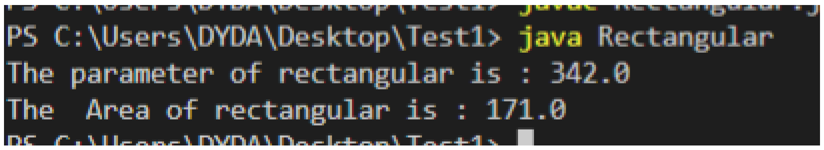
BITA/7/23/025/TZ KHADIJA HAJI ALI

1. (Rectangle Class) Create a class Rectangle with attributes length and width, each of which defaults to 1. Provide member functions that calculate the perimeter and the area of the rectangle. Also, provide set and get functions for the length and width attributes. The set functions should verify that length and width are each floating-point numbers larger than 0.0 and less than 20.0.



```java
public class Rectangular{
    private double length;
    private double width;

    public Rectangular(double length,double width){
        this.length=length;
        this.width=width;
    }
    public void setlength(double length){
        if(length>0.0||length<=20.0){
            this.length=length;
            this.length=1.0;
        }
    }
    public double getlength(){
            return width;
        }

    public void setWidth(double width){
        if(width>0.0||width<=20.0){
            this.width=width;
            this.width=1.0;
        }
    }
     public double getWidth(){
            return width;
        }
        public double getParameter(){
            return 2*(length*width);
        }
        public double getArea(){
            return length*width;
        }

        public static void main (String [] args){
            Rectangular r = new  Rectangular(19,9);
            System.out.println("The parameter of rectangular is : "+r.getParameter());
            System.out.println("The  Area of rectangular is : "+r.getArea());


        }
}
```

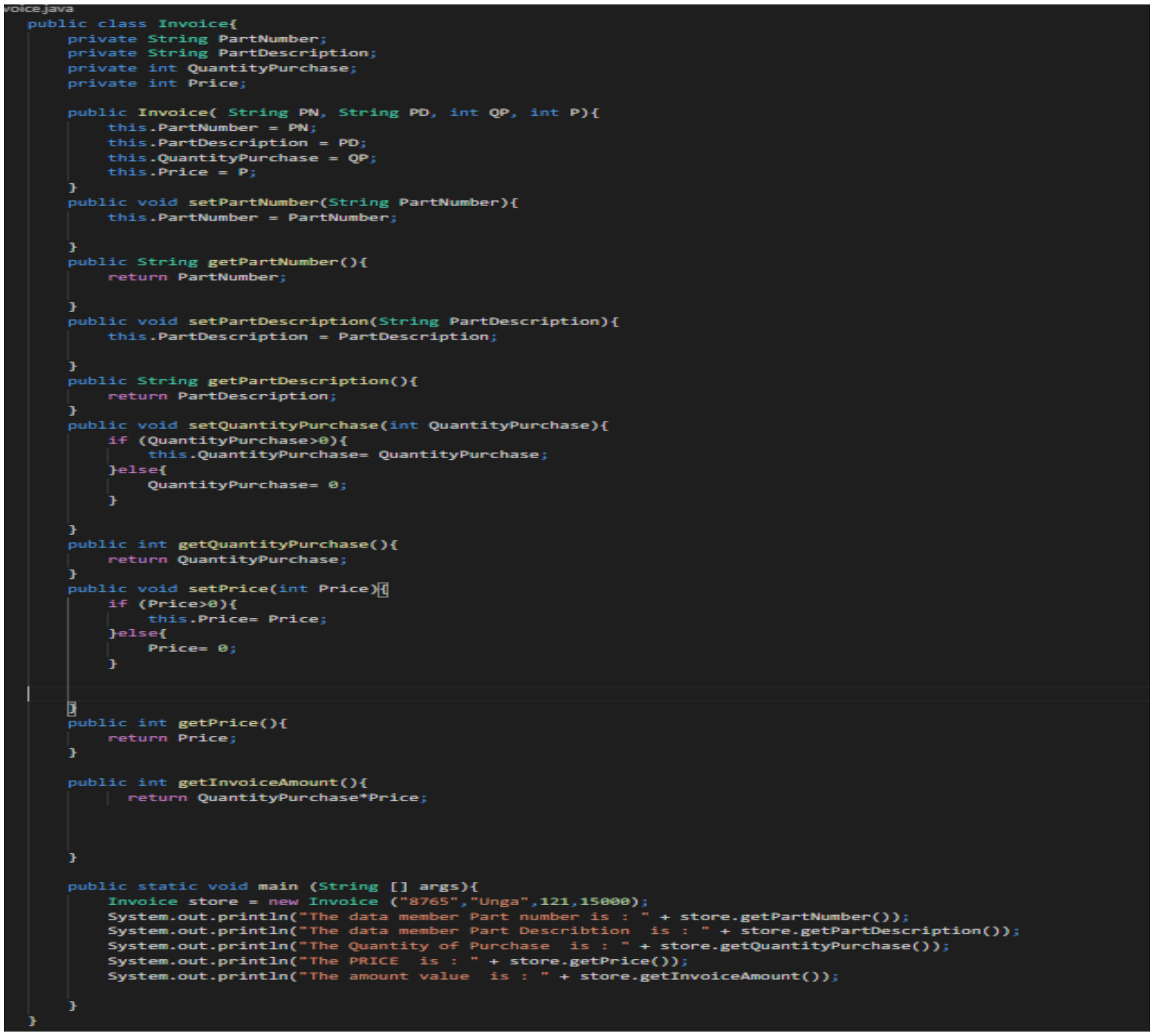
ANSWER



```
PS C:\Users\DYDA\Desktop\Test1> java Rectangular
The parameter of rectangular is : 342.0
The  Area of rectangular is : 171.0
```

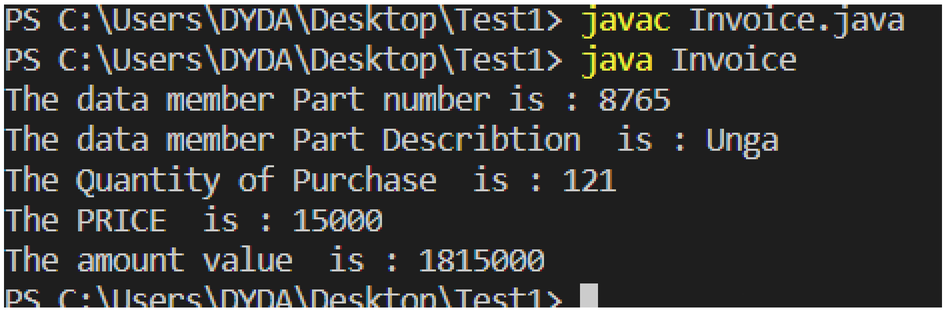
2. (Invoice Class) Create a class called Invoice that a hardware store might use to represent an invoice for an item sold at the store. An Invoice should include four pieces of information as data members a part number (type string), a part description (type string), a quantity of the item being purchased (type int) and a price per item (type int). Your class should have a constructor that initializes the four data members. Provide a set and a get function for each data member. In addition, provide a member function named getInvoiceAmount that calculates the invoice amount (i.e., multiplies the quantity by the price per item), then returns the amount as an int

value. If the quantity is not positive, it should be set to 0. If the price per item is not positive, it should be set to 0. Write a test program that demonstrates class Invoice's capabilities.



```
voice.java
public class Invoice{
    private String PartNumber;
    private String PartDescription;
    private int QuantityPurchase;
    private int Price;

    public Invoice( String PN, String PD, int QP, int P){
        this.PartNumber = PN;
        this.PartDescription = PD;
        this.QuantityPurchase = QP;
        this.Price = P;
    }
    public void setPartNumber(String PartNumber){
        this.PartNumber = PartNumber;

    }
    public String getPartNumber(){
        return PartNumber;

    }
    public void setPartDescription(String PartDescription){
        this.PartDescription = PartDescription;

    }
    public String getPartDescription(){
        return PartDescription;
    }
    public void setQuantityPurchase(int QuantityPurchase){
        if (QuantityPurchase>0){
            this.QuantityPurchase= QuantityPurchase;
        }else{
            QuantityPurchase= 0;
        }

    }
    public int getQuantityPurchase(){
        return QuantityPurchase;
    }
    public void setPrice(int Price){
        if (Price>0){
            this.Price= Price;
        }else{
            Price= 0;
        }


    }
    public int getPrice(){
        return Price;
    }

    public int getInvoiceAmount(){
            return QuantityPurchase*Price;



    }

    public static void main (String [] args){
        Invoice store = new Invoice ("8765","Unga",121,15000);
        System.out.println("The data member Part number is : " + store.getPartNumber());
        System.out.println("The data member Part Describtion  is : " + store.getPartDescription());
        System.out.println("The Quantity of Purchase  is : " + store.getQuantityPurchase());
        System.out.println("The PRICE  is : " + store.getPrice());
        System.out.println("The amount value  is : " + store.getInvoiceAmount());

    }
}
```

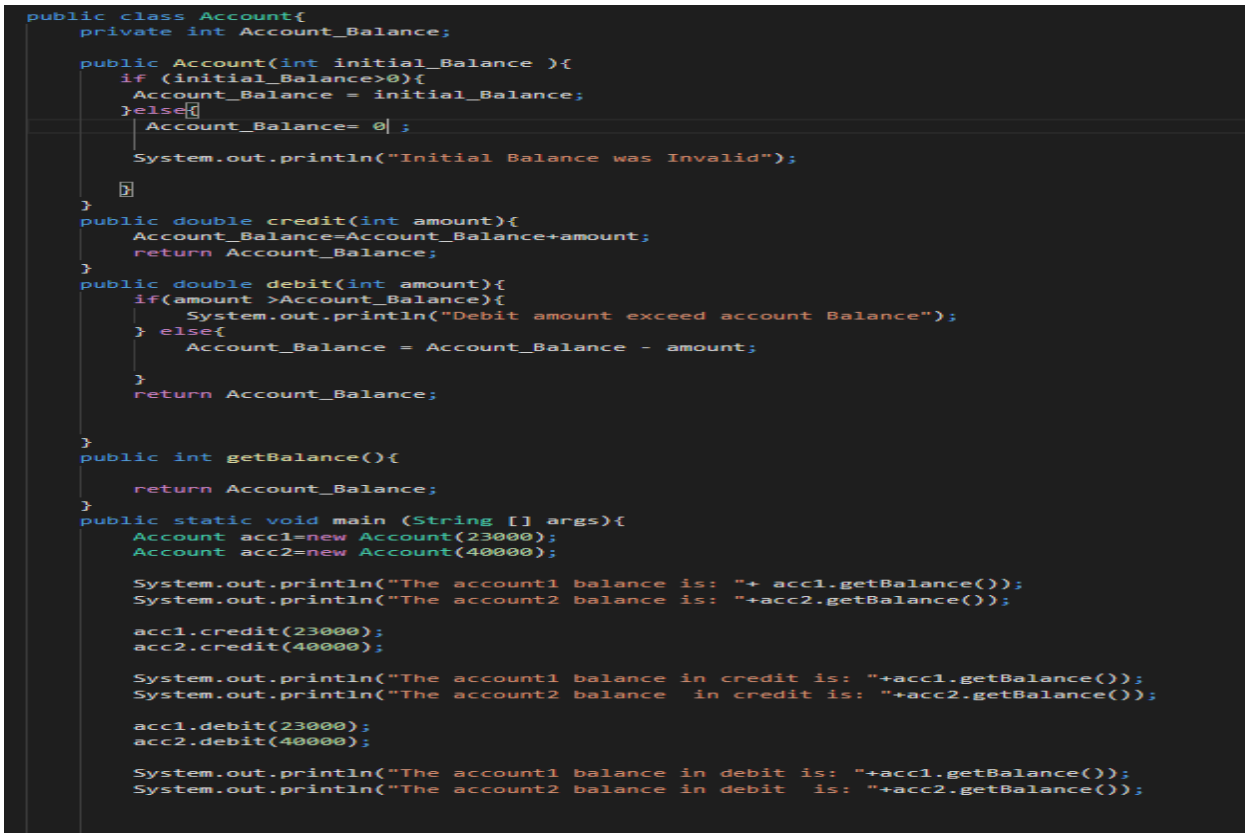
ANSWER



```
PS C:\Users\DYDA\Desktop\Test1> javac Invoice.java
PS C:\Users\DYDA\Desktop\Test1> java Invoice
The data member Part number is : 8765
The data member Part Describtion  is : Unga
The Quantity of Purchase  is : 121
The PRICE  is : 15000
The amount value  is : 1815000
PS C:\Users\DYDA\Desktop\Test1>
```

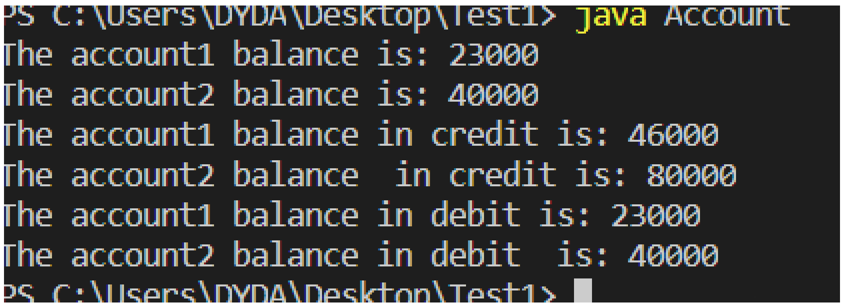
3. (Account Class) Create a class called Account that a bank might use to represent customers' bank accounts. Your class should include one data member of type int to represent the account balance. Your class should provide a constructor that receives an initial balance and uses it to initialize the data member. The constructor should validate the initial balance to ensure that it is

greater than or equal to 0. If not, the balance should be set to 0 and the constructor should display an error message, indicating that the initial balance was invalid. The class should provide three member functions. Member function credit should add an amount to the current balance. Member function debit should withdraw money from the Account and should ensure that the debit amount does not exceed the Account's balance. If it does, the balance should be left unchanged and the function should print a message indicating "Debit amount exceeded account balance." Member function getBalance should return the current balance. Create a program that creates two Account objects and tests the member functions of class Account.



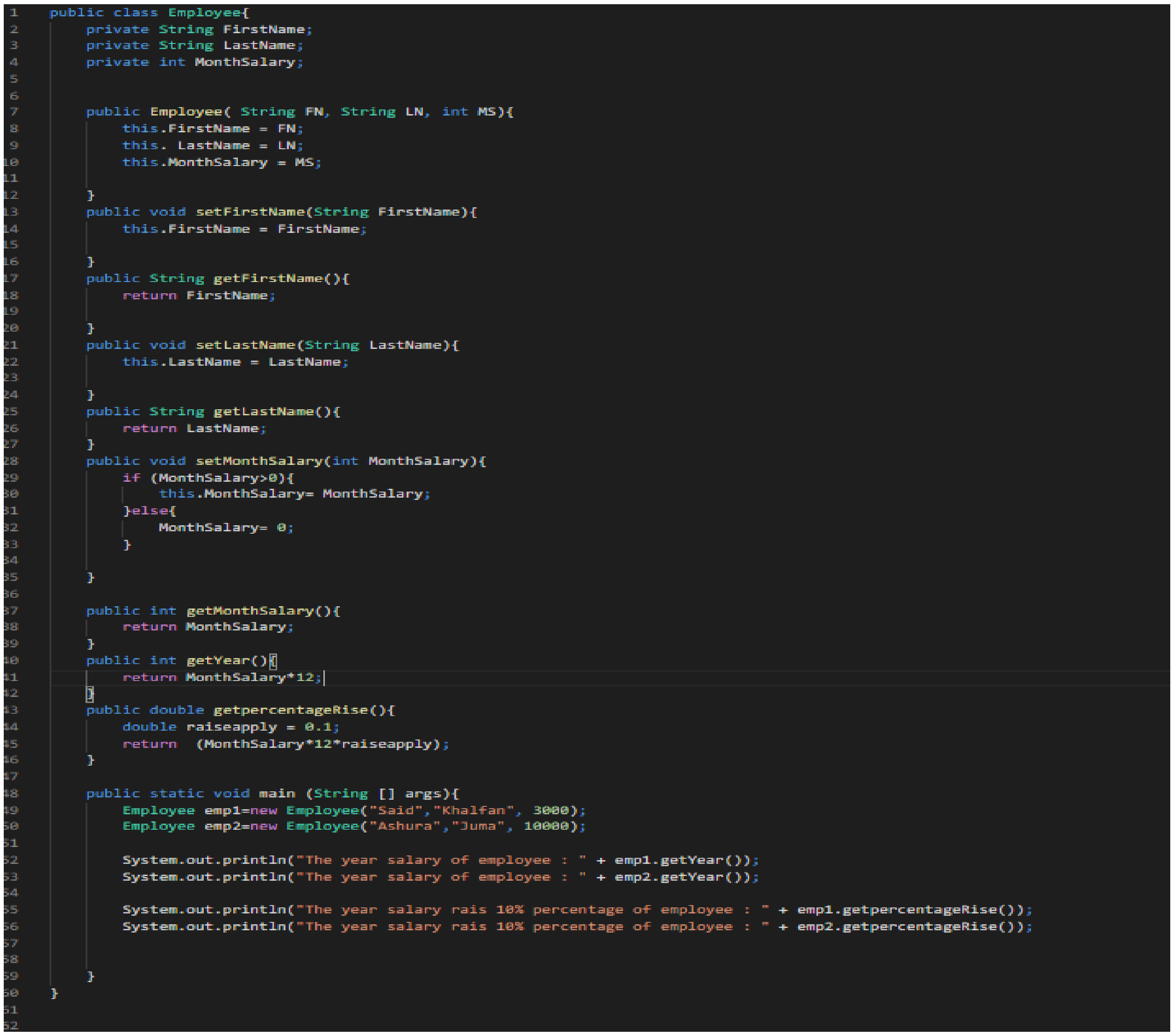
```java
public class Account{
    private int Account_Balance;

    public Account(int initial_Balance ){
       if (initial_Balance>0){
        Account_Balance = initial_Balance;
       }else{
         Account_Balance= 0 ;

        System.out.println("Initial Balance was Invalid");

       }
    }
    public double credit(int amount){
        Account_Balance=Account_Balance+amount;
        return Account_Balance;
    }
    public double debit(int amount){
        if(amount >Account_Balance){
            System.out.println("Debit amount exceed account Balance");
        } else{
            Account_Balance = Account_Balance - amount;

        }
        return Account_Balance;


    }
    public int getBalance(){

        return Account_Balance;
    }
    public static void main (String [] args){
        Account acc1=new Account(23000);
        Account acc2=new Account(40000);

        System.out.println("The account1 balance is: "+ acc1.getBalance());
        System.out.println("The account2 balance is: "+acc2.getBalance());

        acc1.credit(23000);
        acc2.credit(40000);

        System.out.println("The account1 balance in credit is: "+acc1.getBalance());
        System.out.println("The account2 balance  in credit is: "+acc2.getBalance());

        acc1.debit(23000);
        acc2.debit(40000);

        System.out.println("The account1 balance in debit is: "+acc1.getBalance());
        System.out.println("The account2 balance in debit  is: "+acc2.getBalance());
```

ANSWERS



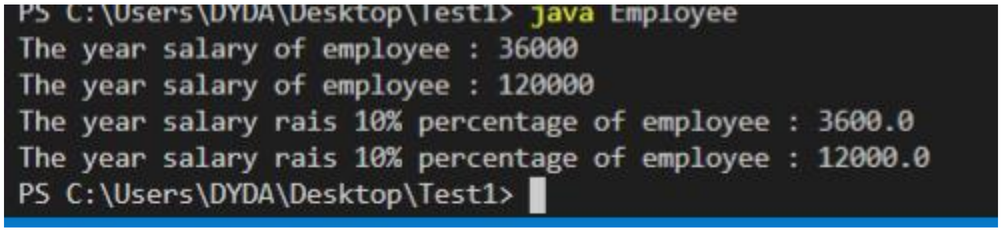
```
PS C:\Users\DYDA\Desktop\Test1> java Account
The account1 balance is: 23000
The account2 balance is: 40000
The account1 balance in credit is: 46000
The account2 balance  in credit is: 80000
The account1 balance in debit is: 23000
The account2 balance in debit  is: 40000
PS C:\Users\DYDA\Desktop\Test1>
```

4. (Employee Class) Create a class called Employee that includes three pieces of information as data members a first name (type string), a last name (type string) and a monthly salary (type int). Your class should have a constructor that initializes the three data members. Provide a set and a get function for each data member. If the monthly salary is not positive, set it to 0. Write a test program that demonstrates class Employee's capabilities. Create two Employee objects and display each object's yearly salary. Then give each Employee a 10 percent raise and display each Employee's yearly salary again.



```java
public class Employee{
    private String FirstName;
    private String LastName;
    private int MonthSalary;


    public Employee( String FN, String LN, int MS){
        this.FirstName = FN;
        this. LastName = LN;
        this.MonthSalary = MS;

    }
    public void setFirstName(String FirstName){
        this.FirstName = FirstName;

    }
    public String getFirstName(){
        return FirstName;

    }
    public void setLastName(String LastName){
        this.LastName = LastName;

    }
    public String getLastName(){
        return LastName;
    }
    public void setMonthSalary(int MonthSalary){
        if (MonthSalary>0){
            this.MonthSalary= MonthSalary;
        }else{
            MonthSalary= 0;
        }

    }

    public int getMonthSalary(){
        return MonthSalary;
    }
    public int getYear(){
        return MonthSalary*12;
    }
    public double getpercentageRise(){
        double raiseapply = 0.1;
        return  (MonthSalary*12*raiseapply);
    }

    public static void main (String [] args){
        Employee emp1=new Employee("Said","Khalfan", 3000);
        Employee emp2=new Employee("Ashura","Juma", 10000);

        System.out.println("The year salary of employee : " + emp1.getYear());
        System.out.println("The year salary of employee : " + emp2.getYear());

        System.out.println("The year salary rais 10% percentage of employee : " + emp1.getpercentageRise());
        System.out.println("The year salary rais 10% percentage of employee : " + emp2.getpercentageRise());


    }
}
```

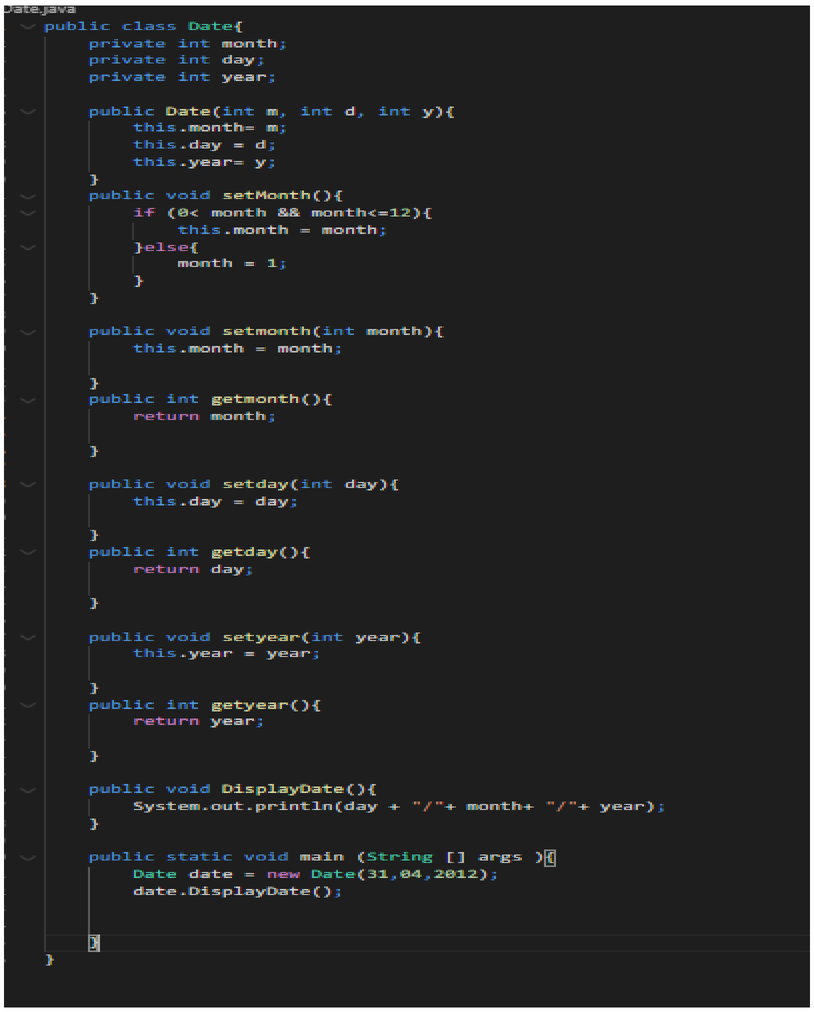
ANSWER



```
PS C:\Users\DYDA\Desktop\Test1> java Employee
The year salary of employee : 36000
The year salary of employee : 120000
The year salary rais 10% percentage of employee : 3600.0
The year salary rais 10% percentage of employee : 12000.0
PS C:\Users\DYDA\Desktop\Test1>
```

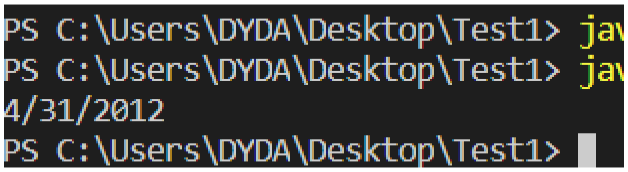
BITA/7/23/025/TZ KHADIJA HAJI ALI

5. (Date Class) Create a class called Date that includes three pieces of information as data members a month (type int), a day (type int) and a year (type int). Your class should have a constructor with three parameters that uses the parameters to initialize the three data members. For the purpose of this exercise, assume that the values provided for the year and day are correct, but ensure that the month value is in the range 1-12; if it is not, set the month to 1. Provide a set and a get function for each data member. Provide a member function displayDate that displays the month, day and year separated by forward slashes (/). Write a test program that demonstrates class Date's capabilities.



```
Date.java
public class Date{
    private int month;
    private int day;
    private int year;

    public Date(int m, int d, int y){
        this.month= m;
        this.day = d;
        this.year= y;
    }
    public void setMonth(){
        if (0< month && month<=12){
            this.month = month;
        }else{
            month = 1;
        }
    }

    public void setmonth(int month){
        this.month = month;

    }
    public int getmonth(){
        return month;

    }

    public void setday(int day){
        this.day = day;

    }
    public int getday(){
        return day;

    }

    public void setyear(int year){
        this.year = year;

    }
    public int getyear(){
        return year;

    }

    public void DisplayDate(){
        System.out.println(day + "/"+ month+ "/"+ year);
    }

    public static void main (String [] args ){
        Date date = new Date(31,04,2012);
        date.DisplayDate();


    }
}
```

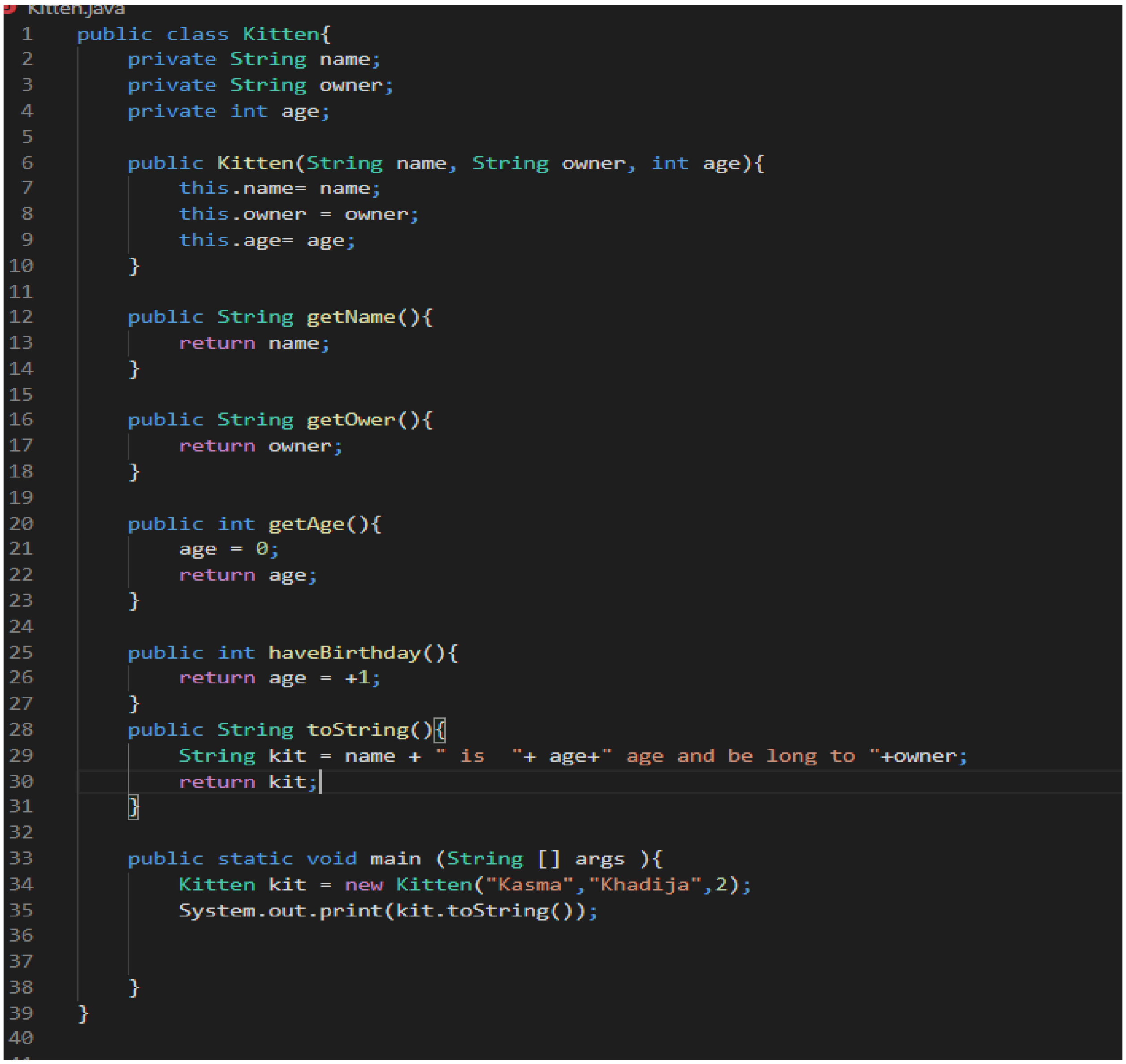
ANSWER



```
PS C:\Users\DYDA\Desktop\Test1> jav
PS C:\Users\DYDA\Desktop\Test1> jav
4/31/2012
PS C:\Users\DYDA\Desktop\Test1>
```

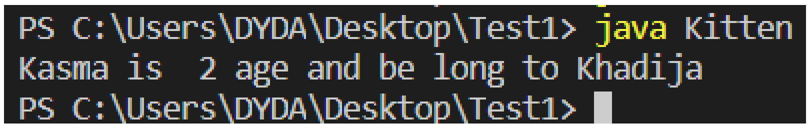
6. Create a class called Kitten that has three fields: String name, String owner, and int age. Create a constructor for Kitten that takes a String name and owner for the Kitten and uses them

for initialization. Have the age for a Kitten start at 0; Implement a method for age which return a Kitten's age. Implement a method called haveBirthday that does not return anything and simply increases a Kitten's age by one. Finally, write a method called toString that returns a string of the form: " is and belongs to " e.g. "Bob is 87 and belongs to Gregor Samsa". Create an object which initialize three data members and print out the string above



```
Kitten.java
public class Kitten{
    private String name;
    private String owner;
    private int age;

    public Kitten(String name, String owner, int age){
        this.name= name;
        this.owner = owner;
        this.age= age;
    }

    public String getName(){
        return name;
    }

    public String getOwer(){
        return owner;
    }

    public int getAge(){
        age = 0;
        return age;
    }

    public int haveBirthday(){
        return age = +1;
    }
    public String toString(){
        String kit = name + " is  "+ age+" age and be long to "+owner;
        return kit;
    }

    public static void main (String [] args ){
        Kitten kit = new Kitten("Kasma","Khadija",2);
        System.out.print(kit.toString());


    }
}
```

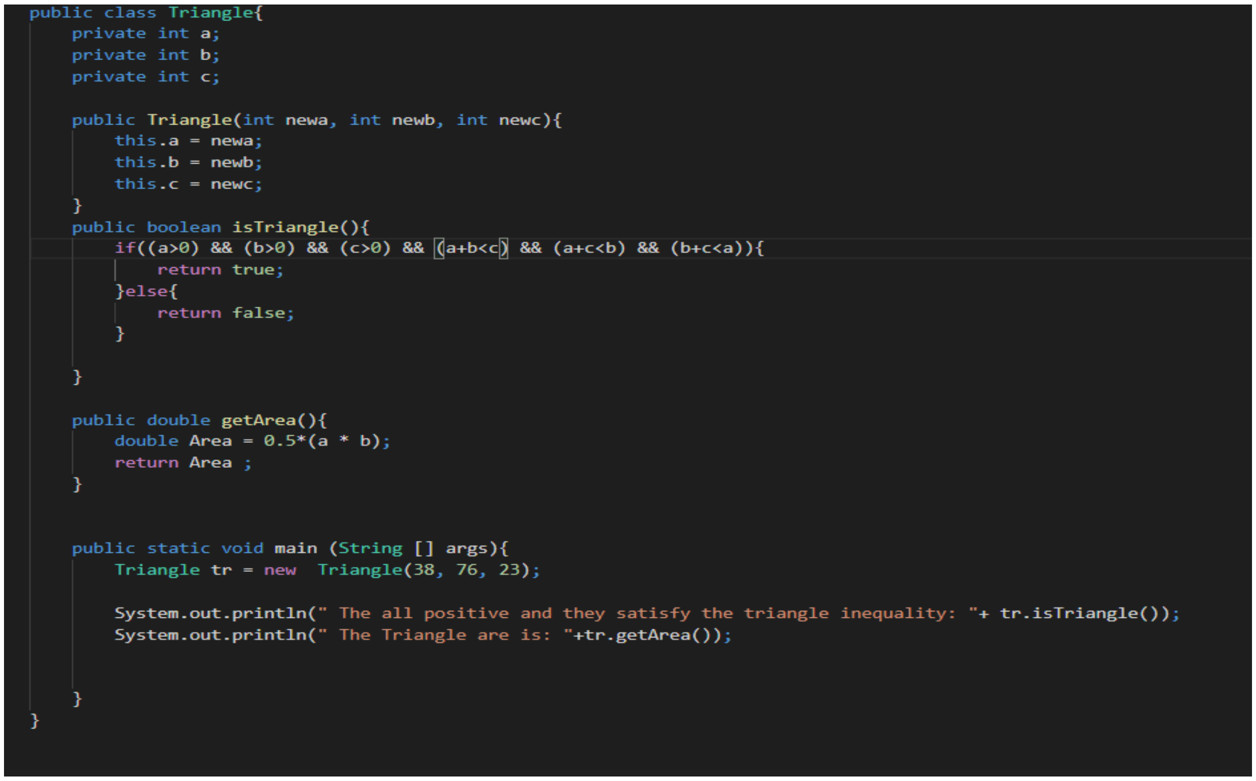
ANSWER



```
PS C:\Users\DYDA\Desktop\Test1> java Kitten
Kasma is  2 age and be long to Khadija
PS C:\Users\DYDA\Desktop\Test1>
```

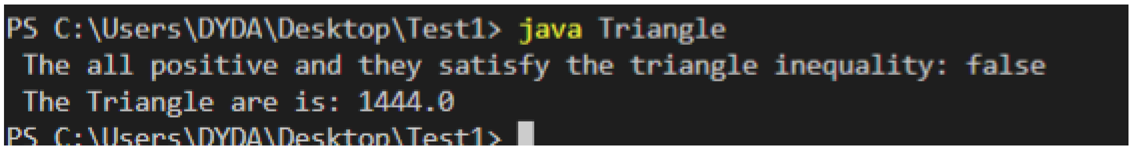
7. Define a class called Triangle with three integer data members a, b, and c as the lengths of its three edges. This class should have the following methods: (a)a constructor with 3 parameters

representing the 3 edges (b)a method isTriangle() which returns true if the 3 edges are all positive and they satisfy the triangle inequality where a+b > c, a+c > b, b+c > a. (c)a method getArea() which returns the area of triangle. NB: Area=1/2(a*b) The signature of these methods are given below: public Triangle(int newa, int newb, int newc) public boolean isTriangle() public double getArea() Note: getAngle() should return zero if the triangle is not really a triangle.



```
public class Triangle{
    private int a;
    private int b;
    private int c;

    public Triangle(int newa, int newb, int newc){
        this.a = newa;
        this.b = newb;
        this.c = newc;
    }
    public boolean isTriangle(){
        if((a>0) && (b>0) && (c>0) && (a+b<c) && (a+c<b) && (b+c<a)){
            return true;
        }else{
            return false;
        }

    }

    public double getArea(){
        double Area = 0.5*(a * b);
        return Area ;
    }


    public static void main (String [] args){
        Triangle tr = new  Triangle(38, 76, 23);

        System.out.println(" The all positive and they satisfy the triangle inequality: "+ tr.isTriangle());
        System.out.println(" The Triangle are is: "+tr.getArea());


    }
}
```

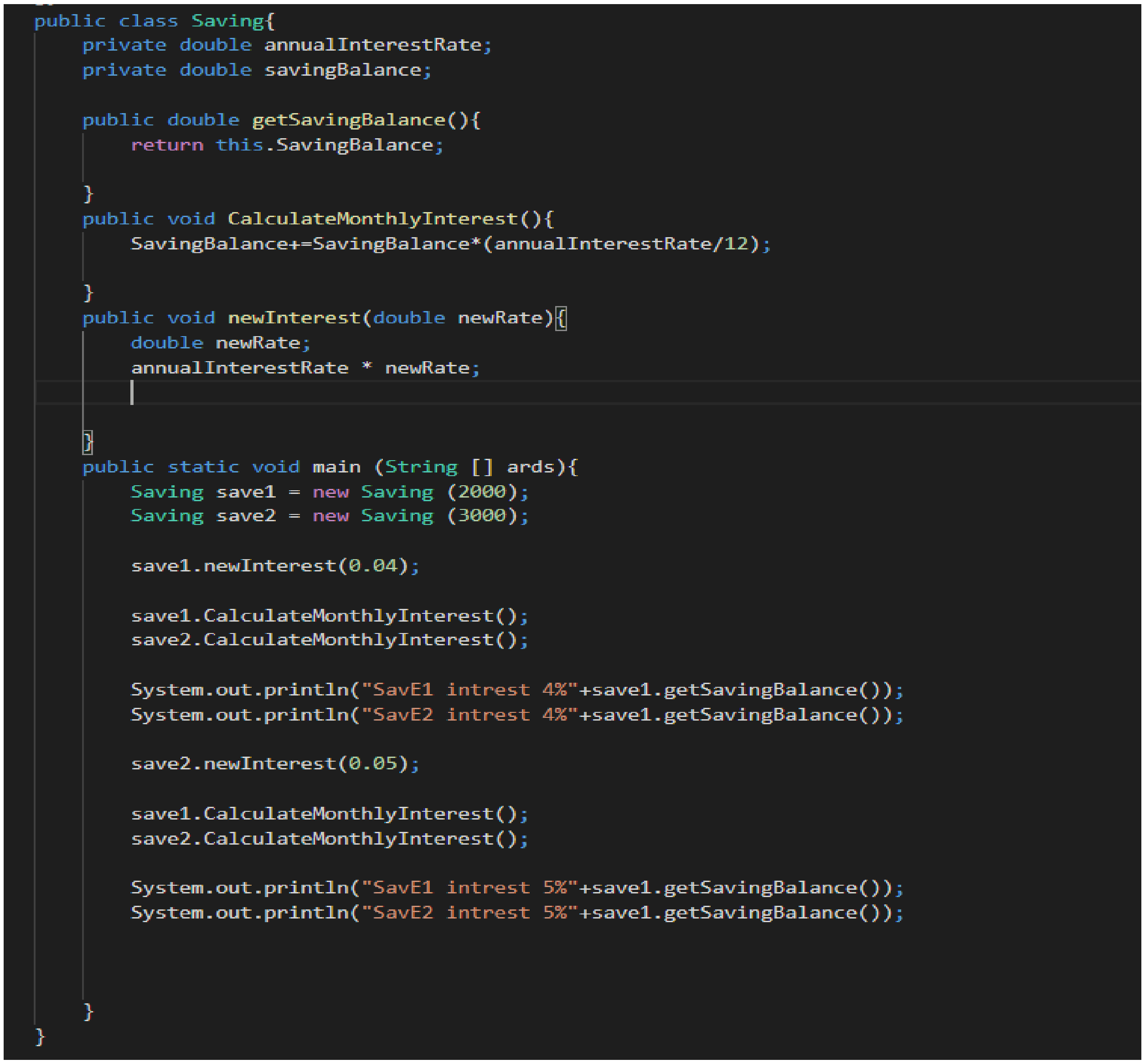
ANSWER



```
PS C:\Users\DYDA\Desktop\Test1> java Triangle
 The all positive and they satisfy the triangle inequality: false
 The Triangle are is: 1444.0
PS C:\Users\DYDA\Desktop\Test1>
```

8. Create class SavingsAccount. Usea static variable annualInterestRate to store the annual interest rate for all account holders. Each object of the class contains a private instance variable savingsBalance indicating the amount the saver currently has ondeposit. Provide method calculateMonthlyInterest to calculate the monthly interest by multiplying the savingsBalance by annualInterestRate divided by 12 this interest should be added to savingsBalance. Provide a static method modifyInterestRate that sets the annualInterestRate to a new value. Write a program to test class SavingsAccount. Instantiate two savingsAccount objects, saver1 and saver2, with balances of $2000.00 and $3000.00, respectively. Set annualInterestRate to 4%, then calculate the monthly interest and print the new balances for both savers. Then set the annualInterestRate to 5%, calculate the next month’s interest and print the new balances for both savers

BITA/7/23/025/TZ KHADIJA HAJI ALI

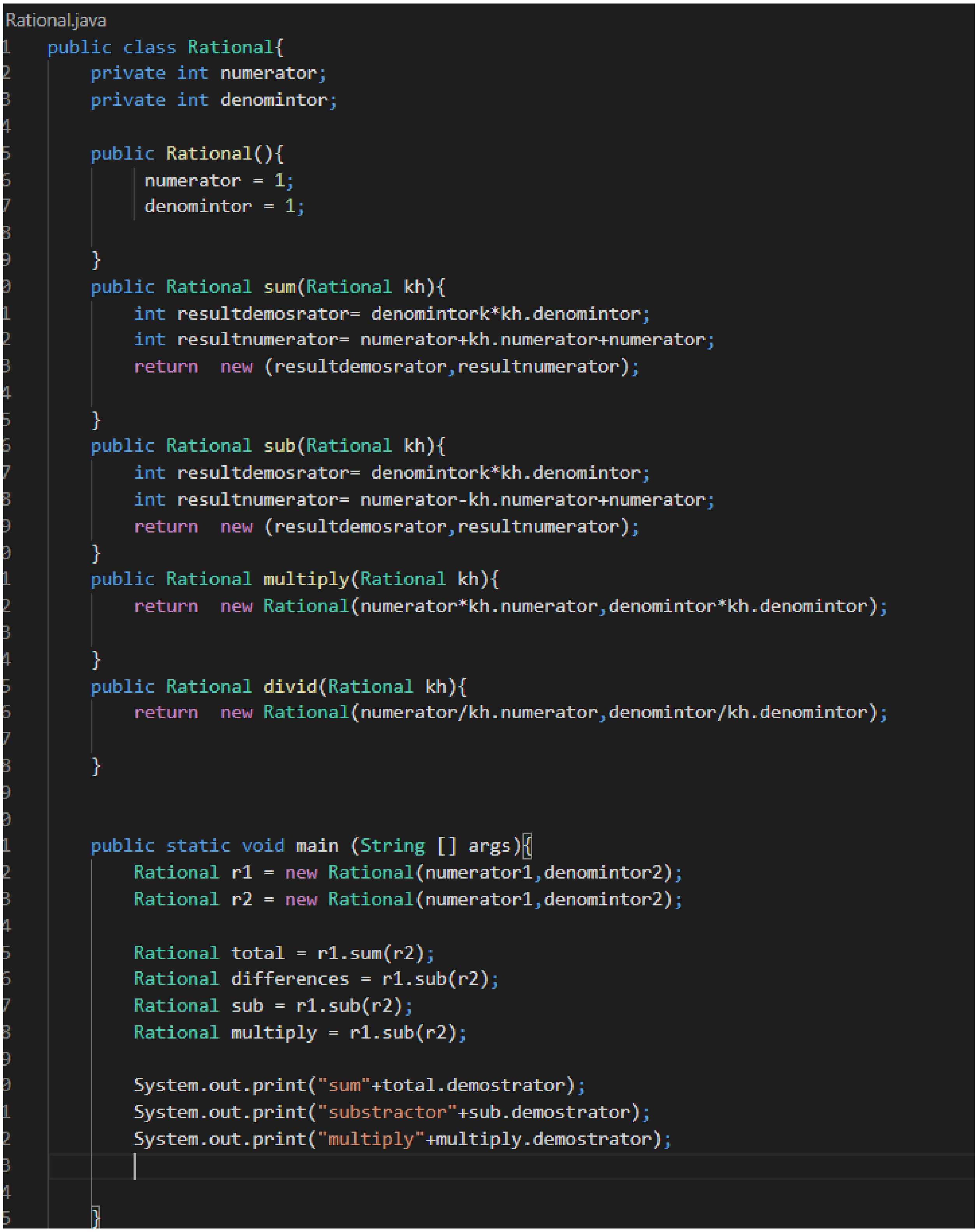


```java
public class Saving{
    private double annualInterestRate;
    private double savingBalance;

    public double getSavingBalance(){
        return this.SavingBalance;

    }
    public void CalculateMonthlyInterest(){
        SavingBalance+=SavingBalance*(annualInterestRate/12);

    }
    public void newInterest(double newRate){
        double newRate;
        annualInterestRate * newRate;
        

    }
    public static void main (String [] ards){
        Saving save1 = new Saving (2000);
        Saving save2 = new Saving (3000);

        save1.newInterest(0.04);

        save1.CalculateMonthlyInterest();
        save2.CalculateMonthlyInterest();

        System.out.println("SavE1 intrest 4%"+save1.getSavingBalance());
        System.out.println("SavE2 intrest 4%"+save1.getSavingBalance());

        save2.newInterest(0.05);

        save1.CalculateMonthlyInterest();
        save2.CalculateMonthlyInterest();

        System.out.println("SavE1 intrest 5%"+save1.getSavingBalance());
        System.out.println("SavE2 intrest 5%"+save1.getSavingBalance());



    }
}
```

13. (Rational Class) Create a class called Rational for performing arithmetic with fractions. Write a program to test your class. Use integer variables to represent the private data of the classthe numerator and the denominator. Provide a constructor that enables an object of this class to be initialized when it is declared. The constructor should contain default values in case no initializers are provided and should store the fraction in reduced form. For example, the fraction would be stored in the object as 1 in the numerator and 2 in the denominator. Provide public member functions that perform each of the following tasks: a. Adding two Rational numbers. The result should be stored in reduced form. b. Subtracting two Rational numbers. The result should be stored in reduced form. c. Multiplying two Rational numbers. The result should be stored in reduced form. d. Dividing two Rational numbers. The result should be stored in reduced form. e. Printing Rational numbers in the form a/b, where a is the

BITA/7/23/025/TZ KHADIJA HAJI ALI

numerator and b is the denominator. f. Printing Rational numbers in floating-point format


Rational.java

```java
public class Rational{
    private int numerator;
    private int denomintor;

    public Rational(){
         numerator = 1;
         denomintor = 1;

    }
    public Rational sum(Rational kh){
        int resultdemosrator= denomintork*kh.denomintor;
        int resultnumerator= numerator+kh.numerator+numerator;
        return  new (resultdemosrator,resultnumerator);

    }
    public Rational sub(Rational kh){
        int resultdemosrator= denomintork*kh.denomintor;
        int resultnumerator= numerator-kh.numerator+numerator;
        return  new (resultdemosrator,resultnumerator);
    }
    public Rational multiply(Rational kh){
        return  new Rational(numerator*kh.numerator,denomintor*kh.denomintor);

    }
    public Rational divid(Rational kh){
        return  new Rational(numerator/kh.numerator,denomintor/kh.denomintor);

    }


    public static void main (String [] args){
        Rational r1 = new Rational(numerator1,denomintor2);
        Rational r2 = new Rational(numerator1,denomintor2);

        Rational total = r1.sum(r2);
        Rational differences = r1.sub(r2);
        Rational sub = r1.sub(r2);
        Rational multiply = r1.sub(r2);

        System.out.print("sum"+total.demostrator);
        System.out.print("substractor"+sub.demostrator);
        System.out.print("multiply"+multiply.demostrator);


    }
```