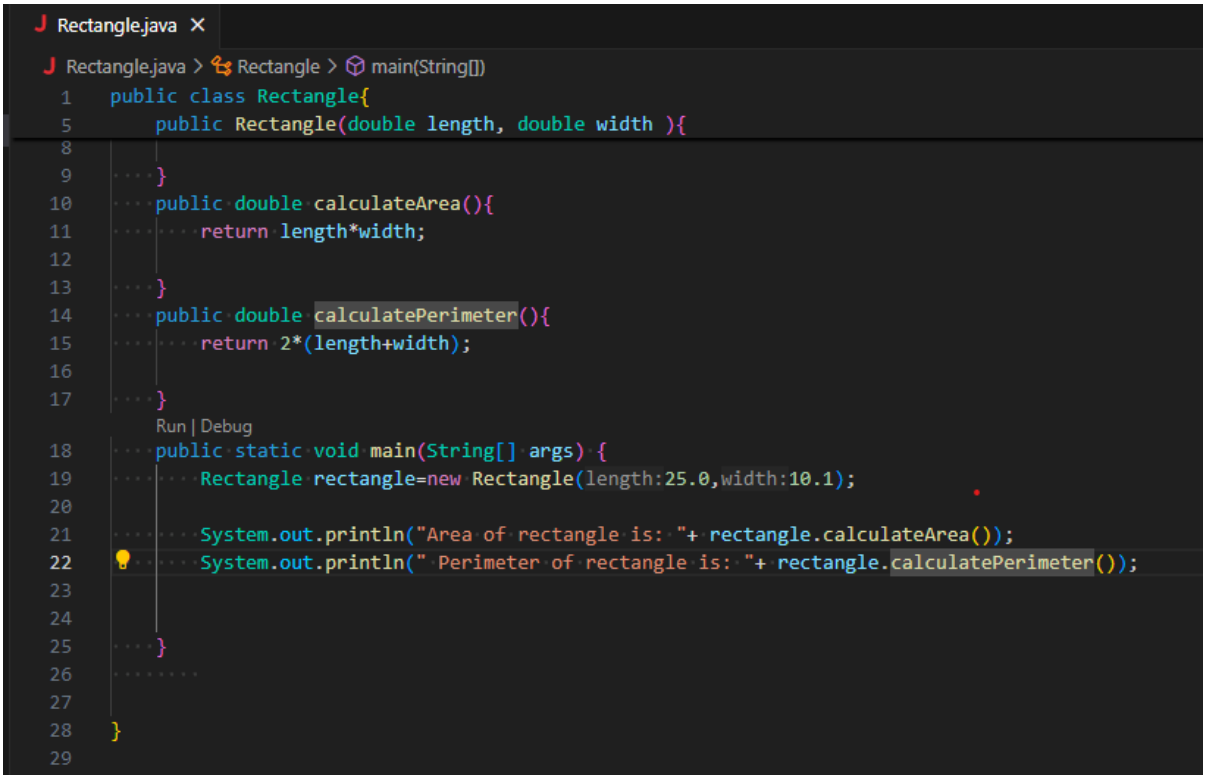
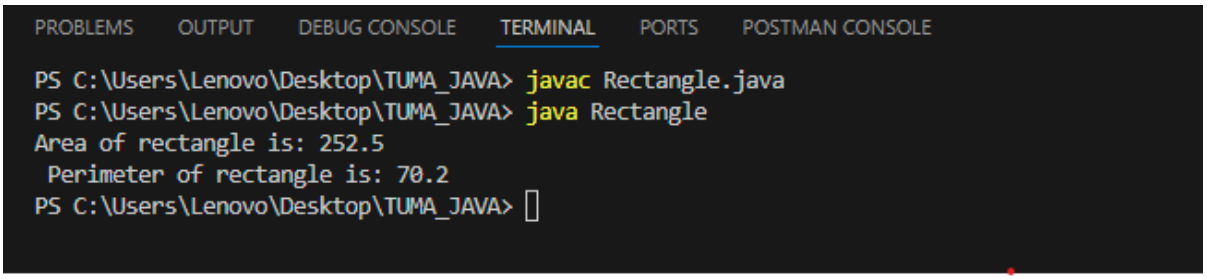
FATUMA KHAMIS ISSA BITA/7/23/062/TZ

1. (Rectangle Class) Create a class Rectangle with attributes length and width. Provide member functions that calculate the perimeter and the area of the rectangle.

CODE



OUTPUT



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. 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.

CODE

Invoice.java > Invoice

```java
public class Invoice {
    private String part_number;
    private String part_description;
    private int quantity;
    private int price_item;

    public Invoice (String price_per_item,String part_description,int quantity,int price_item ) {
        this.part_number =part_number;
        this.part_description = part_description;
        setQuantity(quantity);
        setPrice(price_item);


    }

    void setQuantity(int quantity){
        this.quantity =quantity;
        if (quantity<0)
            quantity =0;


    }
    void setPrice(int price_item){
        this.price_item =price_item;
        if (price_item<0)
            price_item =0;
    }
    public int getQuantity(){
        return quantity;
    }
```

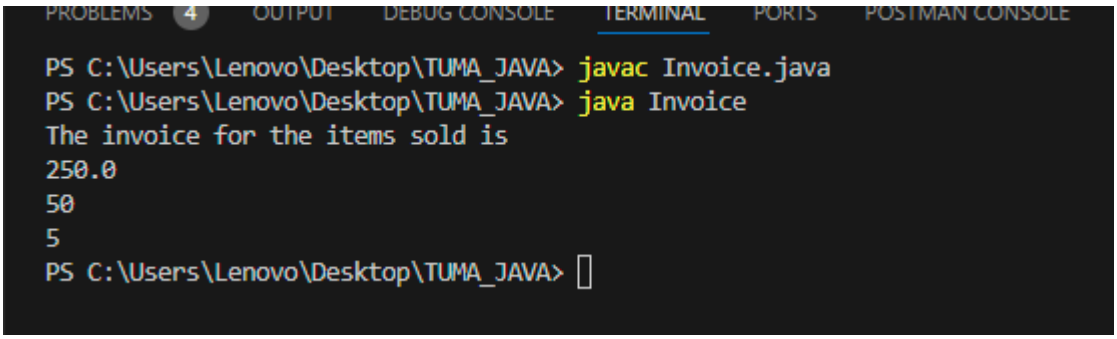
```java
    }
    public int getQuantity(){
        return quantity;
    }
    public int getprice(){
        return price_item;
    }
    double getInvoiceAmount(){
        return quantity*price_item;
    }

    Run | Debug
    public static void main(String []args){
        Invoice invoice = new Invoice(price_per_item:"z123",part_description:"Basket",quantity:5,pr…50);
        System.out.println(x:"The invoice for the items sold is ");
        System.out.println(invoice.getInvoiceAmount());
        System.out.println(invoice.getprice());
        System.out.println(invoice.getQuantity());


    }

}
```

OUTPUT



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.

CODE

FATUMA KHAMIS ISSA BITA/7/23/062/TZ

```java
public class Account {
    private int balance;
    public Account (int initial_balance){
        if(initial_balance >= 0){
            this.balance =initial_balance;
        } else{
            this.balance =0;
            System.out.println("Amount is invalid ");
        }
    }
    public void credit(int amount){
        this.balance += amount;
    }
    public void debit (int amount){
        if (amount <= balance) {
            this.balance-= amount;

        }else{
            System.out.println("The debit aamount exceed the account balance ");
        }
    }
    public int getBalance(){
        return this.balance;
    }
```

```java
    Run | Debug
    public static void main(String[]args){
        Account account1 =new Account(initial_balance:5000);
        Account account2 =new Account(initial_balance:0);

        account1.credit(amount:200);
        account1.debit(amount:300);
        account2.credit(amount:80);
        account1.debit(amount:55);

        System.out.println("Account1 balance is: " +account1.getBalance() );
        System.out.println("Account2 balance is: " + account2.getBalance());

        account2.debit(amount:100);

    }
}
```

OUTPUT

```
PS C:\Users\Lenovo\Desktop\TUMA_JAVA> javac Account.java
PS C:\Users\Lenovo\Desktop\TUMA_JAVA> java Account
Account1 balance is: 4845
Account2 balance is: 80
The debit aamount exceed the account balance
PS C:\Users\Lenovo\Desktop\TUMA_JAVA>
```

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. 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.

CODE

Employee.java > Employee > main(String[])

```java
public class Employee{
    private String firstname;
    private String lastname;
    private int monthly_salary;

    public Employee(String firstname,String lastname,int monthly_salary ){
        this.firstname =firstname;
        this.lastname = lastname;
        this.monthly_salary=Math.max(a:0,monthly_salary);

    }
    public void giveRaise(double raisePercentage){
        double raiseAmount = this.monthly_salary*raisePercentage/100;
        this.monthly_salary += raiseAmount;
    }
    public int getYearlySalary(){
        return this.monthly_salary*12;
    }
    public void displayEmployee(){
        System.out.println("Employee named " +this.firstname+this.lastname);
        System.out.println("His yearly salary is " +this.getYearlySalary());
    }
    Run | Debug
    public static void main(String[] args) {
        Employee employee1 = new Employee(firstname:"Fatma", lastname:"Issa", monthly_salary:50000);
        Employee employee2 = new Employee(firstname:"Zaituni", lastname:"Shamata", monthly_salary:60000);

        System.out.println("Annual salary of " +employee1.firstname+" "+
        employee1.lastname+" is "+employee1.getYearlySalary() );
        System.out.println("Annual salary of " +employee2.firstname+" "+
        employee2.lastname+" is "+employee2.getYearlySalary() );
    }
```

OUTPUT

```
PS C:\Users\Lenovo\Desktop\TUMA_JAVA> javac Employee.java
PS C:\Users\Lenovo\Desktop\TUMA_JAVA> java Employee
Annual salary  of Fatma Issa is 600000
Annual salary of Zaituni Shamata is 720000
PS C:\Users\Lenovo\Desktop\TUMA_JAVA>
```

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 member function displayDate that displays the month, day and year separated by forward slashes (/). Write a test program that demonstrates class

   Date's capabilities.

CODE

Date.java > Date

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

    public Date(int month,int day,int year){
        this.month =(month>=1 && month<=12) ? month:1;
        this.day=day;
        this.year = year;
    }
    public void displayDate(){
        System.out.println(this.month +"/"+this.day+"/"+this.year);
    }
    Run | Debug
    public static void main(String[] args) {
        Date date1 =new Date(month:4, day:2, year:2003);
        Date date2 = new Date(month:4, day:30, year:2003);
        System.out.println(x:"Date1 is: ");
        date1.displayDate();
        System.out.println(x:"Date2 is: ");
        date2.displayDate();
    }

}
```

OUTPUT

```
PS C:\Users\Lenovo\Desktop\TUMA_JAVA> javac Date.java
PS C:\Users\Lenovo\Desktop\TUMA_JAVA> java Date
Date1 is:
4/2/2003
Date2 is:
4/30/2003
PS C:\Users\Lenovo\Desktop\TUMA_JAVA>
```