

Question 1

Write a java program to implement matrix addition operation

- **Code :**

```
import java.util.Scanner;

class One {

    public static void main(String[] args) {

        Scanner s = new Scanner(System.in);

        int[][] a = new int[3][3];

        for(int i = 0;i<3;i++) {

            for(int j = 0;j<3;j++) {

                System.out.print("Enter Value of 1st Matrix[" + i + "][" + j + "] : ");

                a[i][j]= s.nextInt();

            }

        }

        System.out.print("\n");

        int[][] b = new int[3][3];

        for(int i = 0;i<3;i++) {

            for(int j = 0;j<3;j++) {

                System.out.print("Enter Value of 2nd Matrix[" + i + "][" + j + "] : ");

                b[i][j]= s.nextInt();

            }

        }

        System.out.println("\nFirst Matrix");

        for(int i=0;i<3;i++) {

            for(int j=0;j<3;j++) {

                System.out.print(a[i][j] + " ");

            } System.out.print("\n");

        }

        System.out.println("Second Matrix");

        for(int i=0;i<3;i++) {
```

```
        for(int j=0;j<3;j++) {  
            System.out.print(b[i][j] + " ");  
        } System.out.print("\n");  
    }  
  
    System.out.println("Addition Of Matrix");  
  
    for(int i=0;i<3;i++) {  
        for(int j=0;j<3;j++) {  
            System.out.print(a[i][j]+b[i][j] + " ");  
        } System.out.print("\n");  
    }  
  
    s.close();  
}  
}
```

- Output :

```
C:\Java>java One  
Enter Value of 1st Matrix[0][0] : 1  
Enter Value of 1st Matrix[0][1] : 2  
Enter Value of 1st Matrix[0][2] : 3  
Enter Value of 1st Matrix[1][0] : 4  
Enter Value of 1st Matrix[1][1] : 5  
Enter Value of 1st Matrix[1][2] : 6  
Enter Value of 1st Matrix[2][0] : 7  
Enter Value of 1st Matrix[2][1] : 8  
Enter Value of 1st Matrix[2][2] : 9  
  
Enter Value of 2nd Matrix[0][0] : 9  
Enter Value of 2nd Matrix[0][1] : 8  
Enter Value of 2nd Matrix[0][2] : 7  
Enter Value of 2nd Matrix[1][0] : 6  
Enter Value of 2nd Matrix[1][1] : 5  
Enter Value of 2nd Matrix[1][2] : 4  
Enter Value of 2nd Matrix[2][0] : 3  
Enter Value of 2nd Matrix[2][1] : 2  
Enter Value of 2nd Matrix[2][2] : 1
```

```
First Matrix
1 2 3
4 5 6
7 8 9
Second Matrix
9 8 7
6 5 4
3 2 1
Addition Of Matrix
10 10 10
10 10 10
10 10 10
```

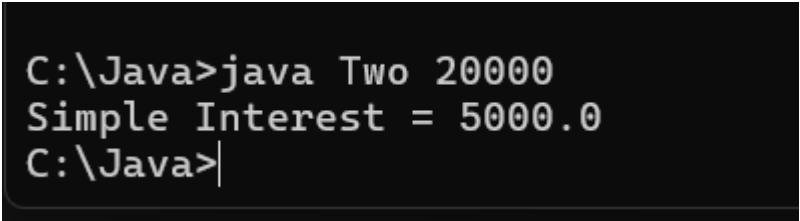
Question 2

Write a Java Program to calculate simple interest using Command -- line Arguments.

- Code :

```
class Two {  
    public static void main(String[] args) {  
        double n = Double.parseDouble(args[0]);  
        double rate = 5;  
        double year = 5;  
        double interest = rate * year * n * 0.01;  
        System.out.print("Simple Interest = " + interest);  
    }  
}
```

- Output :



```
C:\Java>java Two 20000  
Simple Interest = 5000.0  
C:\Java>
```

Question 3

Write a program which will display information of a particular product. Product class have following properties:

Data Members: [Pr no | Pr_name | Quantity | Price per unit]

Methods:

constructor () - for insertion of a product information.

search ()- which will search information of product, search will be done by product name given by user.

display ()- which will display information of a product.

- **Code :**

```
import java.util.Scanner;
```

```
class Product {  
    int pr_no;  
    String pr_name;  
    int quantity;  
    int price;  
    public Product() {  
        Scanner s = new Scanner(System.in);  
        System.out.print("Enter Product Name = ");  
        pr_name = s.nextLine();  
        System.out.print("Enter Product SrNo = ");  
        pr_no = s.nextInt();  
        System.out.print("Enter Product Quantity = ");  
        quantity = s.nextInt();  
        System.out.print("Enter Product Price = ");  
        price = s.nextInt();  
    }  
    public void Display() {
```

```
        System.out.println("Product Name = "+pr_name);
        System.out.println("Product SrNo = "+pr_no);
        System.out.println("Product Quantity = "+quantity);
        System.out.println("Product Price = "+price);
    }
    public int Search(String find,int f) {
        boolean check = find.equalsIgnoreCase(pr_name);
        if(check == true) {
            System.out.println("Product Found!");
            Display();
            f = 1;
        }
        return f;
    }
    public static void main(String []args) {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter Total Number Of the Products = ");
        int n = s.nextInt();

        Product[] pro = new Product[n];

        for(int i=0;i<n;i++) {
            int x=i+1;
            System.out.println("\nEnter Details Of Product "+x+" = ");
            pro[i]=new Product();
        }
        int option = 0;
        while(option!=3) {
            System.out.print("\n1 : Search\n2 : Display\n3 : Exit");
            System.out.print("\nEnter Choice = ");
            option = s.nextInt();
        }
    }
}
```

```
switch(option) {
    case 1:
        System.out.print("\nEnter Product Name For Search = ");
        Scanner so = new Scanner(System.in);
        String sea;
        sea = so.nextLine();
        int found = 0;
        for(int i=0;i<n;i++) {
            found = pro[i].Search(sea,found);
        }
        if(found==0) {
            System.out.println("No Product Found");
        }
        break;
    case 2:
        for(int i=0;i<n;i++) {
            int x=i+1;
            System.out.println("\nDetails Of Product "+x+" = ");
            pro[i].Display();
        }
        break;
    case 3:
        break;
    default:
        System.out.println("Enter Proper Option!!");
}

}

}
```

- Output :

```
C:\Java>java Product
Enter Total Number Of the Products = 2

Enter Details Of Product 1 =
Enter Product Name = Apple
Enter Product SrNo = 101
Enter Product Quantity = 20
Enter Product Price = 30

Enter Details Of Product 2 =
Enter Product Name = Orange
Enter Product SrNo = 102
Enter Product Quantity = 30
Enter Product Price = 20

1 : Search
2 : Display
3 : Exit
Enter Choice = 1

Enter Product Name For Search = Apple
Product Found!
Product Name = Apple
Product SrNo = 101
Product Quantity = 20
Product Price = 30
```



```
1 : Search
2 : Display
3 : Exit
Enter Choice = 2

Details Of Product 1 =
Product Name = Apple
Product SrNo = 101
Product Quantity = 20
Product Price = 30

Details Of Product 2 =
Product Name = Orange
Product SrNo = 102
Product Quantity = 30
Product Price = 20

1 : Search
2 : Display
3 : Exit
Enter Choice = 3

C:\Java>
```

Question 4

Design a class *Cricketer* having data member name and number of matches and appropriate member function to set the values. Derive two classes *Batsman* and *Bowler* from *Cricketer* class with data member total number of runs and wickets respectively. *Batsman* class is having a method to calculate average run rate and *Bowler* class is having method to calculate average wicket. Write a program to create two objects and displays information of one batsman and bowler along with average run and wicket.

- **Code :**

```
import java.util.Scanner;
```

```
class Cricketer {
    String name;
    int nMatches;
    public void getData() {
        Scanner s = new Scanner(System.in);
        System.out.print("Enter Cricketer Name = ");
        name = s.nextLine();
        System.out.print("Enter Total Number of Matches = ");
        nMatches = s.nextInt();
    }
    public void Display() {
        System.out.println("\nCricketer Name = "+name);
        System.out.println("Cricketer's Total Matches = "+nMatches);
    }
}

class Batsman extends Cricketer {
    int totalRuns;
    public Batsman() {
        Scanner s = new Scanner(System.in);
        getData();
    }
}
```

```
        System.out.print("Enter Total Runs = ");
        totalRuns=s.nextInt();
    }
    public double avgRuns() {
        double avg = (double)totalRuns/nMatches;
        return avg;
    }
}

class Bowler extends Cricketer {
    int totalWickets;
    public Bowler() {
        Scanner s = new Scanner(System.in);
        getData();
        System.out.print("Enter Total Wickets = ");
        totalWickets = s.nextInt();
    }
    public double avgWickets() {
        double avg = (double)totalWickets/nMatches;
        return avg;
    }
}

class Four {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        System.out.println("\nEnter Batsman Details :");
        Batsman b = new Batsman();
        System.out.println("\nEnter Bowler Details :");
        Bowler w = new Bowler();
        System.out.print("\nBatsman Details : ");
        b.Display();
        System.out.println("Average Runs Of Batsman = "+b.avgRuns());
    }
}
```

```
        System.out.print("\nBowler Details : ");  
        w.Display();  
        System.out.println("Average Wickets Of Bowler = "+w.avgWickets());  
    }  
}
```

- Output :

```
C:\Java>java Four  
  
Enter Batsman Details :  
Enter Cricketer Name = Player 1  
Enter Total Number of Matches = 4  
Enter Total Runs = 124  
  
Enter Bowler Details :  
Enter Cricketer Name = Player 2  
Enter Total Number of Matches = 6  
Enter Total Wickets = 6  
  
Batsman Details :  
Cricketer Name = Player 1  
Cricketer's Total Matches = 4  
Average Runs Of Batsman = 31.0  
  
Bowler Details :  
Cricketer Name = Player 2  
Cricketer's Total Matches = 6  
Average Wickets Of Bowler = 1.0
```

Question 5

Write a program that will accept a number from command line and raise a user defined exception if the number consists of odd number of digits.

- **Code :**

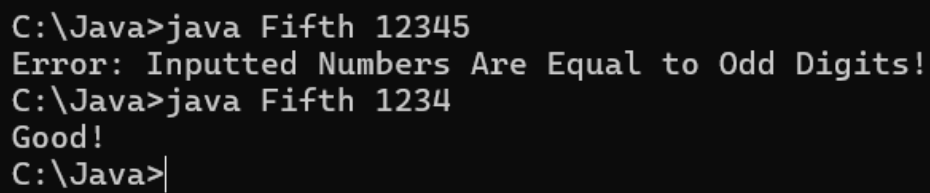
```
import java.util.Scanner;
```

```
class oddDigits extends Exception {  
    public oddDigits() {  
        super("Error: Inputted Numbers Are Equal to Odd Digits!");  
    }  
}
```

```
class Fifth {  
    public void checkNumber(int c) throws oddDigits {  
        if(c%2==1) {  
            throw new oddDigits();  
        } else {  
            System.out.print("Good!");  
        }  
    }  
  
    public static void main(String[] args) {  
        try{  
            Fifth f = new Fifth();  
            int n = Integer.parseInt(args[0]);  
            int z = n;  
            int count = 0;  
            while(z!=0) {  
                z=z/10;  
                count++;  
            }  
        }  
    }  
}
```

```
        f.checkNumber(count);  
    } catch (oddDigits e) {  
        System.out.print(e.getMessage());  
    }  
}  
}
```

- Output :



```
C:\Java>java Fifth 12345  
Error: Inputted Numbers Are Equal to Odd Digits!  
C:\Java>java Fifth 1234  
Good!  
C:\Java>
```

Question 6

Write a program to create interface area. Create three classes called rectangle, triangle, and square calculate areas respectively.

- Code :

Shapes.java

```
public interface Shapes {  
    void calcArea();  
}
```

Rectangle.java

```
import java.util.Scanner;  
  
class Rectangle implements Shapes{  
    double length , width;  
    public Rectangle() {  
        Scanner s = new Scanner(System.in);  
        System.out.println("\nRectangle Details : ");  
        System.out.print("Enter Length = ");  
        length = s.nextDouble();  
        System.out.print("Enter Width = ");  
        width = s.nextDouble();  
    }  
    public void calcArea() {  
        double Area = length * width;  
        System.out.print("Rectangle Area = "+Area);  
    }  
}
```

Triangle.java

```
import java.util.Scanner;

class Triangle implements Shapes {
    double base , height;

    public Triangle() {
        Scanner s = new Scanner(System.in);

        System.out.println("\n\nTriangle Details : ");

        System.out.print("Enter Base = ");

        base = s.nextDouble();

        System.out.print("Enter Height = ");

        height = s.nextDouble();
    }

    public void calcArea() {
        double Area = base * height * 0.5;

        System.out.print("Triangle Area = "+Area);
    }
}
```

Square.java

```
import java.util.Scanner;

class Square implements Shapes {
    double length;

    public Square() {
        Scanner s = new Scanner(System.in);

        System.out.println("\n\nSquare Details : ");

        System.out.print("Enter Length = ");

        length = s.nextDouble();
    }

    public void calcArea() {
        double Area = length * length;

        System.out.print("Square Area = "+Area);
    }
}
```



```
    }  
}
```

Main.java

```
import java.util.Scanner;  
  
class Sixth {  
    public static void main(String args[]) {  
        Rectangle r = new Rectangle();  
        r.calcArea();  
        Triangle t = new Triangle();  
        t.calcArea();  
        Square s = new Square();  
        s.calcArea();  
    }  
}
```

- Output :

```
C:\Java>java Sixth  
  
Rectangle Details :  
Enter Length = 10  
Enter Width = 20  
Rectangle Area = 200.0  
  
Triangle Details :  
Enter Base = 30  
Enter Height = 40  
Triangle Area = 600.0  
  
Square Details :  
Enter Length = 40  
Square Area = 1600.0  
C:\Java>
```

Question 7

Write a program that accepts string data. Extract either All Vowels or All Non-Vowels from given Data According to Options Selection. Also Provide an Option to Display Output in Uppercase or Lowercase.

- Code :

```
import java.util.Scanner;
```

```
class Seventh {
```

```
    public static void main(String args[]) {
```

```
        Scanner s = new Scanner(System.in);
```

```
        String orgStr;
```

```
        System.out.print("Enter String = ");
```

```
        orgStr = s.nextLine();
```

```
        String vowelStr , novowelStr;
```

```
        novowelStr = orgStr.replaceAll("[aeiouAEIOU]", "");
```

```
        vowelStr = orgStr.replaceAll("[^aeiouAEIOU]", "");
```

```
        int ch=0;
```

```
        while(ch!=3) {
```

```
            System.out.print("\n\n1 : Extract Vowels\n2 : Extract Non-Vowels\n3 : Exit");
```

```
            System.out.print("\nEnter Your Choice = ");
```

```
            ch = s.nextInt();
```

```
            System.out.print("\n");
```

```
            switch(ch) {
```

```
                case 1:
```

```
                    System.out.print("1 : UpperCase | 2 : LowerCase | Enter
```

```
Value = ");
```

```
                    int op = s.nextInt();
```

```
                    if (op==1) {
```

```
                        System.out.print("Extracted Vowels =
```

```
"+vowelStr.toUpperCase());
```

```
                    } else if (op==2) {
```

```

                                System.out.print("Extracted Vowels =
"+vowelStr.toLowerCase());

                                } else {
                                    System.out.print("Invaild Option");
                                }
                                break;
                                case 2:
                                    System.out.println("1 : UpperCase | 2 : LowerCase | Enter
Value = ");

                                    op = s.nextInt();
                                    if (op==1) {
                                        System.out.print("Extracted Non-Vowels =
"+novowelStr.toUpperCase());

                                        } else if (op==2) {
                                            System.out.print("Extracted Non-Vowels =
"+novowelStr.toLowerCase());

                                        } else {
                                            System.out.print("Invaild Option");
                                        }
                                    break;
                                case 3:
                                    break;
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
```

- Output :

```
C:\Java>java Seventh
Enter String = Hello World

1 : Extract Vowels
2 : Extract Non-Vowels
3 : Exit
Enter Your Choice = 1

1 : UpperCase | 2 : LowerCase | Enter Value = 2
Extracted Vowels = eoo

1 : Extract Vowels
2 : Extract Non-Vowels
3 : Exit
Enter Your Choice = 2

1 : UpperCase | 2 : LowerCase | Enter Value =
1
Extracted Non-Vowels = HLL WRLD

1 : Extract Vowels
2 : Extract Non-Vowels
3 : Exit
Enter Your Choice = 3
```

Question 8

Create two class one is College and another is Course, College class has data member like name, address and contact no, Course class inherit the properties of College class and it has data member like course name, duration, and intake and do the following operation.

- . Create constructor for base and derived class.
- . Insert data through derived class parameterized constructor.
- . Illustrate the use of this reference and Super.
- . Display all the information through parent class reference variable.

- **Code :**

```
import java.util.Scanner;
```

```
class College {  
    String name;  
    String add;  
    int phno;  
    public College(String name,String add,int phno) {  
        this.name = name;  
        this.add = add;  
        this.phno = phno;  
    }  
    public void display() {  
        System.out.println("College Name = "+name);  
        System.out.println("College Address = "+add);  
        System.out.println("College Contact No = "+phno);  
    }  
}  
  
class Course extends College{  
    String cname;
```

```
int duration;

String intake;

public Course(String name,String add,int phno,String course,int dur,String intakes) {

    super(name,add,phno);

    cname = course;

    duration = dur;

    intake = intakes;

}

@Override

public void display() {

    super.display();

    System.out.println("Course Name = "+cname);

    System.out.println("Course Duration = "+duration);

    System.out.println("Course Intake = "+intake);

}

}

class Eighth {

    public static void main(String args[]) {

        Scanner s = new Scanner(System.in);

        String a,b,c,d;

        int x,y;

        System.out.print("Enter College Name = ");

        a = s.nextLine();

        System.out.print("Enter College Address = ");

        b = s.nextLine();

        System.out.print("Enter College Phone No = ");

        x = s.nextInt();

        s.nextLine();

        System.out.print("Enter Course Name = ");

        c = s.nextLine();

        System.out.print("Enter Course Duration = ");
```

```
y = s.nextInt();  
s.nextLine();  
System.out.print("Enter Course Intake = ");  
d = s.nextLine();  
  
Course obj = new Course(a,b,x,c,y,d);  
College coll = obj;  
  
System.out.println("\nAll Details = ");  
coll.display();  
}  
}
```

- Output

```
C:\Java>java Eighth  
Enter College Name = Udhna College  
Enter College Address = Udhna Surat  
Enter College Phone No = 987654321  
Enter Course Name = BCA  
Enter Course Duration = 3  
Enter Course Intake = 2024-25  
  
All Details =  
College Name = Udhna College  
College Address = Udhna Surat  
College Contact No = 987654321  
Course Name = BCA  
Course Duration = 3  
Course Intake = 2024-25
```

Question 9

Write a program a menu driven program which will do following operations. Declare a String object named str containing String "We are Full Stack Java Developer".

Display length of the String.

Display the first character of each word of the String.

Count total number of white spaces in the given string.

Display the first word in the String.

- **Code :**

```
import java.util.Scanner;
```

```
class Nine {
```

```
    String str;
```

```
    public Nine() {
```

```
        Scanner s = new Scanner(System.in);
```

```
        System.out.print("Enter String = ");
```

```
        str = s.nextLine();
```

```
    }
```

```
    public void firstLetter() {
```

```
        boolean nword = true;
```

```
        for(int i=0;i<str.length();i++) {
```

```
            char ccstr = str.charAt(i);
```

```
            if(nword && ccstr!=' ') {
```

```
                System.out.print(ccstr+" ");
```

```
                nword = false;
```

```
            }
```

```
            if(ccstr == ' ') {
```



```
        nword = true;
    }
} System.out.print("\n");
}

public void countSpace() {
    int count = 0;
    for(int i=0;i<str.length();i++) {
        char ccstr = str.charAt(i);
        if(ccstr == ' ') {
            count++;
        }
    }

    System.out.println("Total WhiteSpaces in String = "+count);
}

public void firstWord() {
    System.out.print("First Word = ");
    for(int i=0;i<str.length();i++) {
        char ccstr = str.charAt(i);
        if(ccstr!=' ') {
            System.out.print(ccstr);
        }
        if(ccstr==' ') {
            break;
        }
    } System.out.print("\n");
}

public static void main(String []args) {
    Scanner s = new Scanner(System.in);
    Nine n = new Nine();
    int c = 0;
    while(c!=5) {
```

```
System.out.print("\n1 : Length\n2 : First Letter\n3 : Count Spaces\n4 : First
Word\n5 : Exit");

System.out.print("\nEnter your Choice = ");
c = s.nextInt();
switch(c) {
    case 1:
        System.out.println("String length = "+n.str.length());
        break;
    case 2:
        System.out.print("First Letter Of Each Word = ");
        n.firstLetter();
        break;
    case 3:
        n.countSpace();
        break;
    case 4:
        n.firstWord();
        break;
    case 5:
        break;
    default:
        System.out.println("Error : Enter Proper Value!");
}
}
}
}
```

- Output :

```
C:\Java>java Nine
Enter String = We Are Full Stack Java Developer
```

```
1 : Length
2 : First Letter
3 : Count Spaces
4 : First Word
5 : Exit
Enter your Choice = 1
String length = 32
```

```
1 : Length
2 : First Letter
3 : Count Spaces
4 : First Word
5 : Exit
Enter your Choice = 2
First Letter Of Each Word = W A F S J D
```

```
1 : Length
2 : First Letter
3 : Count Spaces
4 : First Word
5 : Exit
Enter your Choice = 3
Total WhiteSpaces in String = 5
```

```
1 : Length
2 : First Letter
3 : Count Spaces
4 : First Word
5 : Exit
Enter your Choice = 4
First Word = We
```

Question 10

Write a Java Program which will read a string and rewrite it in the alphabetical Order eg. The word "STRING" should be written a "GINRST".

- Code :

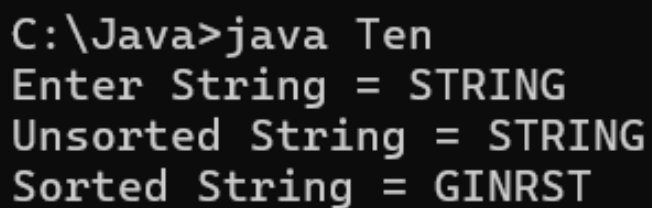
```
import java.util.Arrays;
import java.util.Scanner;

class Ten {
    public static void main(String[] args) {
        Scanner s = new Scanner(System.in);
        String str;
        System.out.print("Enter String = ");
        str = s.nextLine();

        char[] a = str.toCharArray();
        Arrays.sort(a);

        String sortedstr = new String(a);
        System.out.println("Unsorted String = "+str);
        System.out.println("Sorted String = "+sortedstr);
    }
}
```

- Output :



```
C:\Java>java Ten
Enter String = STRING
Unsorted String = STRING
Sorted String = GINRST
```

Question 11

Create package stores. Under it create a class called stock with member variable (item_no, item_name, stock_available, and cost). Under the default package create a class called sales with field name (qty_sold) and it is the child class of stores class. Write a program to print the current stock of each item and perform addition.

- Code :

Stock.java

```
package stores;
```

```
public class Stock {  
    int item_no;  
    String item_name;  
    int stock_available;  
    int cost;  
  
    public Stock(int item_no,String item_name,int stock_available,int cost) {  
        this.item_no = item_no;  
        this.item_name = item_name;  
        this.stock_available = stock_available;  
        this.cost = cost;  
    }  
    public void updateStock(int upStock) {  
        stock_available = upStock;  
    }  
    public void addStock(int number) {  
        stock_available = stock_available + number;  
        System.out.println("Stock Added to "+item_name);  
        System.out.println("Curret Stock = "+stock_available);  
    }  
}
```

```
    public int getNo() {  
        return item_no;  
    }  
    public String getName() {  
        return item_name;  
    }  
    public int getStock() {  
        return stock_available;  
    }  
    public int getCost() {  
        return cost;  
    }  
}
```

Sales.java

```
import stores.Stock;  
import java.util.Scanner;  
  
class Sales extends Stock {  
    int qty_sold;  
    public Sales(int a,String b,int c,int d) {  
        qty_sold = 0;  
        super(a,b,c,d);  
    }  
    public void sale(int number) {  
        if(getStock()-number<=0) {  
            System.out.println("\nError : We Don't Have "+getName() +" Enough Stock  
To Sell!");  
            System.out.println("\nCurrent Stocks Of Products & Qty Sold : ");  
            curretStock();  
        } else {
```

```
        updateStock(getStock()-number);

        System.out.println(number+" "+getName()+" Sold!");

        qty_sold = qty_sold + number;

    }

}

public void curretStock() {

    System.out.println(getName() + " | " + getStock() + " | " + qty_sold);

}

public static void main(String []args) {

    Scanner s = new Scanner(System.in);

    System.out.print("Enter Total Product Numbers = ");

    int n = s.nextInt(); s.nextLine();

    Sales[] st = new Sales[n];

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

        System.out.println("\nEnter Details Of Product "+(i+1));

        System.out.print("Enter Item No = ");

        int a = s.nextInt(); s.nextLine();

        System.out.print("Enter Item Name = ");

        String b = s.nextLine();

        System.out.print("Enter Current Item Stock = ");

        int c = s.nextInt(); s.nextLine();

        System.out.print("Enter Item Cost = ");

        int d = s.nextInt(); s.nextLine();

        st[i] = new Sales(a,b,c,d);

    }

    int ch = 0;

    while(ch!=4) {

        System.out.print("\n1 : Check Stock\n2 : Add Stock\n3 : Sale Item\n4 : Exit");

        System.out.print("\nEnter Choice = ");
```

```
ch = s.nextInt(); s.nextLine();
switch(ch) {
    case 1:
        System.out.println("\nCurrent Stocks Of Products & Qty Sold
: ");

        for(int i=0;i<n;i++) {
            st[i].currentStock();
        }
        break;
    case 2:
        for(int i=0;i<n;i++) {
            System.out.print("\nWant to Add Stock In
st[i].getName()+ " ? y/n : ");

            char co = s.nextLine().charAt(0);
            if(co=='y') {
                System.out.print("Enter Number To Add = ");
                int x = s.nextInt(); s.nextLine();
                st[i].addStock(x);
            }
        }
        break;
    case 3:
        for(int i=0;i<n;i++) {
            System.out.print("\nWant to Sale
st[i].getName()+ " ? y/n : ");

            char co = s.nextLine().charAt(0);
            if(co=='y') {
                System.out.print("Enter Number To Sale =

");

                int x = s.nextInt(); s.nextLine();
                st[i].sale(x);
            }
        }
    }
```



```
        }  
        break;  
    case 4:  
        break;  
    default:  
        System.out.println("Please Enter Proper Option!");  
    }  
}  
}
```

- Output :

```
C:\Java>java --enable-preview Sales  
Enter Total Product Numbers = 2
```

```
Enter Details Of Product 1  
Enter Item No = 101  
Enter Item Name = Orange  
Enter Current Item Stock = 10  
Enter Item Cost = 35
```

```
Enter Details Of Product 2  
Enter Item No = 102  
Enter Item Name = Apple  
Enter Current Item Stock = 20  
Enter Item Cost = 55
```

```
1 : Check Stock  
2 : Add Stock  
3 : Sale Item  
4 : Exit  
Enter Choice = 1
```

```
Current Stocks Of Products & Qty Sold :  
Orange | 10 | 0  
Apple | 20 | 0
```

```
1 : Check Stock
2 : Add Stock
3 : Sale Item
4 : Exit
Enter Choice = 3

Want to Sale Orange ? y/n : y
Enter Number To Sale = 20

Error : We Don't Have Orange Enough Stock To Sell!

Current Stocks Of Products & Qty Sold :
Orange | 10 | 0

Want to Sale Apple ? y/n : y
Enter Number To Sale = 5
5 Apple Sold!
```

```
1 : Check Stock
2 : Add Stock
3 : Sale Item
4 : Exit
Enter Choice = 2

Want to Add Stock In Orange ? y/n : y
Enter Number To Add = 10
Stock Added! to Orange
Curret Stock = 20

Want to Add Stock In Apple ? y/n : n

1 : Check Stock
2 : Add Stock
3 : Sale Item
4 : Exit
Enter Choice = 1

Current Stocks Of Products & Qty Sold :
Orange | 20 | 0
Apple | 15 | 5
```

Question 12

Write a program that accept Book information like Title, Author, Publication and Price for the N book from the user and display books in descending order with interval of 1 second using thread.

- **Code :**

```
import java.util.Scanner;
```

```
import java.util.Arrays;
```

```
class Book extends Thread{
```

```
    String title,author,publication;
```

```
    int price;
```

```
    public Book(String title,String author,String publication,int price) {
```

```
        this.title = title;
```

```
        this.author = author;
```

```
        this.publication = publication;
```

```
        this.price = price;
```

```
    }
```

```
    @Override
```

```
    public void run() {
```

```
        try {
```

```
            System.out.println(this);
```

```
            Thread.sleep(1000);
```

```
        } catch (InterruptedException e) {
```

```
            e.printStackTrace();
```

```
        }
```

```
    }
```

```
    @Override
```

```
    public String toString() {
```

```
        return "Book{" +
```

```
            "title=" + title + '\n' +
```

```
        ", author='" + author + '\" +  
        ", publication='" + publication + '\" +  
        ", price=" + price +  
        '};  
    }  
  
    public static void main(String args[]) {  
        Scanner s = new Scanner(System.in);  
        System.out.print("Enter Total Books Number = ");  
        int n = s.nextInt(); s.nextLine();  
        Book[] books = new Book[n];  
        for(int i=0;i<n;i++) {  
            System.out.println("\nEnter Details of Book "+(i+1));  
            System.out.print("Enter Book Title = ");  
            String a = s.nextLine();  
            System.out.print("Enter Book Author Name = ");  
            String b = s.nextLine();  
            System.out.print("Enter Book Publish Date = ");  
            String c = s.nextLine();  
            System.out.print("Enter Book Price = ");  
            int d = s.nextInt(); s.nextLine();  
            books[i] = new Book(a,b,c,d);  
        }  
        System.out.print("\n");  
        Arrays.sort(books, (b1, b2) -> b2.title.compareTo(b1.title));  
  
        for (Book book : books) {  
            book.start();  
            try {  
                book.join();  
            } catch (InterruptedException e) {  
                e.printStackTrace();  
            }  
        }  
    }  
}
```

```
    }  
    }  
  
    }  
}
```

- Output :

```
C:\Java>java Book  
Enter Total Books Number = 2  
  
Enter Details of Book 1  
Enter Book Title = A book  
Enter Book Author Name = Author 1  
Enter Book Publish Date = 4th June  
Enter Book Price = 100  
  
Enter Details of Book 2  
Enter Book Title = B Book  
Enter Book Author Name = Author 2  
Enter Book Publish Date = 7th June  
Enter Book Price = 200  
  
Book{title='B Book', author='Author 2', publication='7th June', price=200}  
Book{title='A book', author='Author 1', publication='4th June', price=100}
```

Question 13

Create a Menu driven program to implement Singly Linked list and perform operation like create, traverse, insert and delete node.

- **Code :**

```
import java.util.Scanner;
```

```
class Node {
```

```
    int data;
```

```
    Node next;
```

```
    Node(int data) {
```

```
        this.data = data;
```

```
        this.next = null;
```

```
    }
```

```
}
```

```
class SinglyLinkedList {
```

```
    private Node head;
```

```
    public void create(int data) {
```

```
        head = new Node(data);
```

```
        System.out.println("Linked list created with head node " + data);
```

```
    }
```

```
    public void traverse() {
```

```
        if (head == null) {
```

```
            System.out.println("Linked list is empty.");
```

```
            return;
```

```
        }
```

```
        Node temp = head;
```

```
while (temp != null) {  
    System.out.print(temp.data + " ");  
    temp = temp.next;  
}  
System.out.println();  
}  
  
public void insert(int data) {  
    Node newNode = new Node(data);  
    if (head == null) {  
        head = newNode;  
        System.out.println("Inserted " + data + " as head node.");  
    } else {  
        Node temp = head;  
        while (temp.next != null) {  
            temp = temp.next;  
        }  
        temp.next = newNode;  
        System.out.println("Inserted " + data + " at the end of the list.");  
    }  
}  
  
public void delete(int data) {  
    if (head == null) {  
        System.out.println("Linked list is empty.");  
        return;  
    }  
    if (head.data == data) {  
        head = head.next;  
        System.out.println("Deleted head node with data " + data);  
        return;  
    }  
}
```

```
    }  
    Node temp = head;  
    while (temp.next != null && temp.next.data != data) {  
        temp = temp.next;  
    }  
    if (temp.next == null) {  
        System.out.println("Node with data " + data + " not found.");  
    } else {  
        temp.next = temp.next.next;  
        System.out.println("Deleted node with data " + data);  
    }  
}  
}
```

```
class Last {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        SinglyLinkedList list = new SinglyLinkedList();  
        int choice, data;  
  
        while (true) {  
            System.out.println("\nMenu:");  
            System.out.println("1. Create Linked List");  
            System.out.println("2. Traverse Linked List");  
            System.out.println("3. Insert Node");  
            System.out.println("4. Delete Node");  
            System.out.println("5. Exit");  
            System.out.print("Enter your choice: ");  
            choice = scanner.nextInt();  
  
            switch (choice) {
```



```
        case 1:
            System.out.print("Enter data for head node: ");
            data = scanner.nextInt();
            list.create(data);
            break;
        case 2:
            list.traverse();
            break;
        case 3:
            System.out.print("Enter data to insert: ");
            data = scanner.nextInt();
            list.insert(data);
            break;
        case 4:
            System.out.print("Enter data to delete: ");
            data = scanner.nextInt();
            list.delete(data);
            break;
        case 5:
            System.out.println("Exiting...");
            return;
        default:
            System.out.println("Invalid choice. Please try again.");
    }
}
```

- Output :

```
C:\Java>java Last
```

```
Menu:
```

1. Create Linked List
2. Traverse Linked List
3. Insert Node
4. Delete Node
5. Exit

```
Enter your choice: 1
```

```
Enter data for head node: 10
```

```
Linked list created with head node 10
```

```
Menu:
```

1. Create Linked List
2. Traverse Linked List
3. Insert Node
4. Delete Node
5. Exit

```
Enter your choice: 3
```

```
Enter data to insert: 20
```

```
Inserted 20 at the end of the list.
```

```
Menu:
1. Create Linked List
2. Traverse Linked List
3. Insert Node
4. Delete Node
5. Exit
Enter your choice: 3
Enter data to insert: 30
Inserted 30 at the end of the list.
```

```
Menu:
1. Create Linked List
2. Traverse Linked List
3. Insert Node
4. Delete Node
5. Exit
Enter your choice: 2
10 20 30
```

```
Menu:
1. Create Linked List
2. Traverse Linked List
3. Insert Node
4. Delete Node
5. Exit
Enter your choice: 4
Enter data to delete: 10
Deleted head node with data 10
```

```
Menu:
1. Create Linked List
2. Traverse Linked List
3. Insert Node
4. Delete Node
5. Exit
Enter your choice: 2
20 30
```

