

Invoice

```
import java.io.*;
import java.util.Scanner;
class Invoice{
    String number;
    String description;
    int quantity;
    double price;
    public Invoice() {
    }
    public Invoice(String num, String desc, int quant,
                  double pr) {
        number = num;
        description = desc;
        quantity = quant;
        price = pr;
    }
    void SetNumber(String num) {
        number = num;
    }
    void SetDescription(String desc) {
        description = desc;
    }
    void SetQuantity(int quant) {
        quantity = quant;
    }
    void SetPrice(double pr) {
        price = pr;
    }
    String GetNumber() {
        return number;
    }
}
```

```

String getDescription() {
    return description;
}

int getQuantity() {
    return quantity;
}

double getPrice() {
    return price;
}

public double getInvoiceAmount() {
    double invoiceAmount = 0.0;
    if (quantity < 0) {
        invoiceAmount = 0;
    } else if (price < 0.0) {
        invoiceAmount = 0.0;
    } else {
        invoiceAmount = price * quantity;
    }
    return invoiceAmount;
}

public class Main {
    public static void main(String[] args) {
        Invoice myInvoice;
        myInvoice = new Invoice("19BQ1A05L7");
        myInvoice.setPrice(5.0);
        System.out.println("Number: " + myInvoice.getNumber());
        System.out.println("Description: " + myInvoice.getDescription());
        System.out.println("Quantity: " + myInvoice.getQuantity());
        System.out.println("Price: " + myInvoice.getPrice());
        System.out.println("InvoiceAmount: " + myInvoice.getInvoiceAmount());
    }
}

```

Output:
 Number : 19BQ1A05L7
 Description : My First Invoice :)
 Quantity : 5
 Price : 5.0
 Invoice Amount : 25.0

Output:

Number : 19BQ1A05L7

Description : My First Invoice :)

Quantity : 5

Price : 5.0

Invoice Amount : 25.0

"My First Invoice :), 5, 2.0)"

3. Electric bill

```
import java.lang.String;
public class Main {
    private String consumerNo;
    private String consumerName;
    private int prevMonthReading;
    private int currMonthReading;
    private String typeConnection;
    public Main(String cNo, String cName, String typeCon,
               int prvMonthReading, int curMonthReading) {
        consumerNo = cNo;
        consumerName = cName;
        typeConnection = typeCon;
        prevMonthReading = prvMonthReading;
        currMonthReading = curMonthReading;
    }
    public void setConsumerNo(String cNo) {
        consumerNo = cNo;
    }
    public void setConsumerName(String cName) {
        consumerName = cName;
    }
    public void setTypeConnection(String typecon) {
        typeConnection = typecon;
    }
    public void setPrevMonthReading(int prvMonthReading) {
        prevMonthReading = prvMonthReading;
    }
    public void setCurrMonthReading(int curMonthReading) {
        currMonthReading = curMonthReading;
    }
    public String getConsumerNo() {
        return consumerNo;
    }
}
```

```
public String getConsumerName() {
    return consumerName;
}

public String getTypeConnection() {
    return typeConnection;
}

public int getPrevMonthReading() {
    return prevMonthReading;
}

public int getCurrMonthReading() {
    return currMonthReading;
}

public double billAmount() {
    int units = currMonthReading - prevMonthReading;
    double amount = 0;
    if (typeConnection.equals("domestic")) {
        if (units <= 100) {
            amount = units;
        } else if (units <= 200) {
            amount = 100 + (units - 100) * 2.50;
        } else if (units <= 500) {
            amount = 100 + 100 * 2.50 + (units - 200) * 4;
        } else {
            amount = 100 + 100 * 2.50 + 300 * 4 * 6;
        }
    } else {
        if (units <= 100) {
            amount = units * 2;
        } else if (units <= 200) {
            amount = 100 + (units - 100) * 2.50;
        }
    }
}
```

```

        else if (units <= 500) {
            amount = 100 * 2 + 100 * 4.50 + (units - 200) * 6;
        }
        else {
            amount = 100 * 2 + 100 * 4.50 + 300 * 4 + (units - 500) * 9;
        }
    }
    return amount;
}

public static void main(String[] args) {
    Main m = new Main("12345", "Srilatha", "commercial", 5500);
    System.out.println("Customer ID: " + m.getConsumerNo());
    System.out.println("Customer Name: " + m.getConsumerName());
    System.out.println("Type of EB Connection: " + m.getTypeConnection());
    System.out.println("Previous Month Reading: " + m.getPrevMonthReading());
    System.out.println("Current Month Reading: " + m.getCurMonthReading());
    double bill = m.billAmount();
    System.out.println("Your Bill: " + bill);
    System.out.println("In changing the type of EB account...In");
    m.setTypeConnection("Domestic");
    bill = m.billAmount();
    System.out.println("Your Bill: " + bill);
}

System.out.println("Customer ID : 12345");
System.out.println("Customer Name : Srilatha");
System.out.println("Type of EB Connection commercial");
System.out.println("Previous Month Reading 5500");
System.out.println("Current Month Reading 6000");
System.out.println("Your Bill : 2450.0");
System.out.println("Charging the type of EB account...");
System.out.println("Your Bill : 2450.0");

```

3. Savings Account:

```
class SavingsAccount {
    static double annualInterestRate = 0;
    private double savingsBalance;
    public SavingsAccount(double savingsBal) {
        savingsBalance = savingsBal;
    }
    public double calculateMonthlyInterest() {
        double monthlyInterest = (savingsBalance * annualInterestRate) / 12;
        savingsBalance += monthlyInterest;
        return savingsBalance;
    }
    public static void modifyInterestRate(double interestRate) {
        annualInterestRate = interestRate;
    }
}
```

3

```
public class Main {
    public static void main(String[] args) {
        SavingsAccount saver1 = new SavingsAccount(10000.00);
        SavingsAccount saver2 = new SavingsAccount(3000.00);
        saver1.modifyInterestRate(0.04);
        System.out.println("Balance of saver1: " + saver1.calculateMonthlyInterest());
        System.out.println("Balance of saver2: " + saver2.calculateMonthlyInterest());
        saver1.modifyInterestRate(0.05);
        System.out.println("Balance of saver1: " + saver1.calculateMonthlyInterest());
        System.out.println("Balance of saver2: " + saver2.calculateMonthlyInterest());
    }
}
```

output:

Balance of Saver 1 : 6120.0

Balance of Saver 2 : 7140.0

Balance of Saver 1.1 : 6243.6

Balance of Saver 2.1 : 7318.5

A. Array of Books

```
import java.io.*;
import java.util.Scanner;
class Book {
    private String bookname, authorname, publisher;
    private long ISBN;
    public Book() {
    }
    public Book(String bookname, long ISBN, String authorname,
                String publisher) {
        this.bookname = bookname;
        this.ISBN = ISBN;
        this.authorname = authorname;
        this.publisher = publisher;
    }
    public void setData(String bookname, long ISBN, String
                        authorname, String publisher) {
        this.bookname = bookname;
        this.ISBN = ISBN;
        this.authorname = authorname;
        this.publisher = publisher;
    }
    public void getData() {
        System.out.println("Book Name = " + bookname);
        System.out.println("ISBN No. = " + ISBN);
        System.out.println("Author Name = " + authorname);
        System.out.println("publisher = " + publisher);
    }
}
```

class Main {
 public static void main(String args[]) {

```
Scanner scan = new Scanner(System.in);
Book b = new Book();
S.O.P("Enter Book name, ISBN number, Author name,  
publisher :-");
b.setData(scan.nextInt(), scan.nextInt(), scan.next(),
          scan.next());
b.getData();
}
```

Output:

```
Enter Bookname, ISBN number, Author name, publisher,
Book Name = Java
ISBN No. = 12334
Author Name = James
publisher = hari
```

5. Binary Search

```
import java.util.Scanner;  
class Main {  
    public static void main(String[] args) {  
        int a[];  
        int n;  
        Scanner sc = new Scanner(System.in);  
        S.O.P("Enter the size of the array");  
        n = sc.nextInt();  
        a = new int[n];  
        S.O.P("Enter values into array");  
        for (int i = 0; i < n; i++)  
        {  
            S.O.P("Enter at index elem: ", i);  
            a[i] = sc.nextInt();  
        }  
        S.O.P("Enter an element to search");  
        int key = sc.nextInt();  
        int position = binarySearch(a, a.length, key);  
        if (position == -1)  
        {  
            S.O.P("key not found: ");  
        }  
        else {  
            S.O.P("key found at " + position);  
        }  
    }  
    public static int binarySearch(int a[], int n, int key)  
    {  
        int lb, ub, mid;  
        lb = 0;
```

```
ub=n-1;  
while(lb<=ub)  
{  
    mid=(lb+ub)/2;  
    if(a[mid]==key)  
    {  
        return key;  
    }  
    else {  
        if(a[mid]<key)  
        {  
            lb=mid+1;  
        }  
        else {  
            ub=mid-1;  
        }  
    }  
}  
return -1;  
}
```

Output:

6. MergeSort

```
public class MergeSort {
    public static void main(String args[]) {
        int a[] = {67, 56, 78, 92, 34};
        int n = a.length;
        System.out.println("In Before Sorting... ");
        display(a, n);
        mergepass(a, 0, n - 1);
        System.out.println("In After Sorting... ");
        display(a, n);
    }
    public static void mergepass(int a[], int lb, int ub) {
        int mid;
        if (lb != ub) {
            mid = (lb + ub) / 2;
            mergepass(a, lb, mid);
            mergepass(a, mid + 1, ub);
            mergepass(a, lb, mid, ub);
        }
    }
    public static void mergeSort(int a[], int lb,
                                 int mid, int ub) {
        int i;
        int j;
        int k;
        int temp[];
        temp = new int[20];
        i = lb;
        j = mid + 1;
        k = lb;
    }
}
```

while ($i \leq mid$) do ($j \leq ub$)

{ if ($a[i] <= a[j]$)

{ temp[k] = a[i];

k++; i++;

{ temp[k] = a[j];

k++; j++;

}

3 // MERGE

while ($i \leq mid$) {

temp[k] = a[i];

k++; i++;

3 while ($j \leq ub$) {

temp[k] = a[j];

k++; j++;

3 for ($i = lb$; $i \leq ub$; $i++$)

a[i] = temp[i];

3 public static void display (int a[], int n)

{

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

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

3.

Output: Before Sorting . . .

67 56 78 92 84

After Sorting . . .

34 56 67 78 92

```

1. RollPair
import java.util.Random;
public class RollPair {
    public static void main(String[] args) {
        Random randomNumbers = new Random();
        int[][] count = new int[6][6];
        int dice1, dice2;
        for (int roll = 1; roll <= 10000; roll++) {
            dice1 = 1 + randomNumbers.nextInt(6);
            dice2 = 1 + randomNumbers.nextInt(6);
            count[dice1 - 1][dice2 - 1]++;
        }
        System.out.println("Each pair has rolled the following no. of times");
        for (int i = 0; i < 6; i++) {
            for (int j = 0; j < 6; j++) {
                System.out.print("pair " + (i + 1) + "," + (j + 1) + "=" +
                    count[i][j] + " ");
            }
            System.out.println();
        }
    }
}

Output: Each pair has rolled the following no. of times
pair 1,1 = 252 pair 1,2 = 257 pair 1,3 = 271 pair 1,4 = 295
pair 1,5 = 259 pair 1,6 = 284 pair 2,1 = 286 pair 2,2 = 303
pair 2,3 = 296 pair 2,4 = 318 pair 2,5 = 262 pair 2,6 = 256
pair 3,1 = 252 pair 3,2 = 276 pair 3,3 = 291 pair 3,4 = 279
pair 3,5 = 287 pair 3,6 = 284 pair 4,1 = 276 pair 4,2 = 276
pair 4,3 = 269 pair 4,4 = 292 pair 4,5 = 257 pair 4,6 = 268
pair 5,1 = 281 pair 5,2 = 269 pair 5,3 = 339 pair 5,4 = 267
pair 5,5 = 283 pair 5,6 = 296 pair 6,1 = 266 pair 6,2 = 302
pair 6,3 = 270 pair 6,4 = 259 pair 6,5 = 274 pair 6,6 = 259

```

8. Valid password

```
import java.util.regex.*;
class GFG {
    public static boolean isValidpassword(String password) {
        String regex = "^(?=.*[0-9])"
                      + "(?=.*[a-zA-Z])(?=.*[!@#$%^&*()_+=-])"
                      + "(?=\\S+$).{8,20}$";
        Pattern p = pattern.compile(regex);
        if (password == null) {
            return false;
        }
        Matcher m = p.matcher(password);
        return m.matches();
    }
    public static void main(String args[])
    {
        String str1 = "SriLathaVUJTIO";
        S.O.P(isValidpassword(str1));
        String str2 = "2nd CSE - D";
        S.O.P(isValidpassword(str2));
        String str3 = "VVJT";
        S.O.P(isValidpassword(str3));
        String str4 = "19BQ1A05L7";
        S.O.P(isValidpassword(str4));
        String str5 = "Nambur";
        S.O.P(isValidpassword(str5));
    }
}
```

```
String str6 = "Gruntur";  
s.o.p(isValidpassword(str6));
```

3

3

Output:

false

false

false

false

false

false

Employee

```
import java.util.Scanner;  
class Employee {  
    String Emp-name;  
    int Emp-id;  
    String Address;  
    String Mail-id;  
    int Mobile-no;  
    void display() {  
        S.o.p(Emp-name);  
        S.o.p(Emp-id);  
        S.o.p(Mail-id);  
        S.o.p(Mobile-no);  
    }  
}
```

3 class programmer extends Employee {

```
    int BP;  
    HRA = (int) (0.90 * BP);  
    PF = (int) (0.12 * BP); /*  
    void display() {  
        S.o.p(BP);  
        S.o.p("DA" + 0.97 * BP);  
        S.o.p("HRA" + 0.10 * BP);  
        S.o.p("PF" + 0.12 * BP);  
        S.o.p("STAFF CLUD FUND" + 0.001 * BP);  
    }  
}
```

3 class Assistant_professor extends Employee {

```
    int BP;  
    void display() {  
        S.o.p(BP);  
    }
```

```
S.O.P ("DA"+0.97*BP);  
S.O.P ("HRA"+0.10*BP);  
S.O.P ("PF"+0.12*BP);  
S.O.P ("STAFF CLUD FUND"+0.001*BP);
```

3

```
3 class professor extends Employee {  
    int BP;  
    void display() {  
        S.O.P (BP);  
        S.O.P ("DA"+0.97*BP);  
        S.O.P ("HRA"+0.10*BP);  
        S.O.P ("PF"+0.12*BP);  
        S.O.P ("STAFF CLUD FUND"+0.001*BP);  
    }  
}
```

3

```
public class Main {  
    public static void main (String args[]) {  
        S.O.P ("1. programmer\n" +  
                "2. Assistant - professor\n" +  
                "3. Associate - professor\n" +  
                "4. professor");  
        Scanner input = new Scanner (System.in);  
        S.O.P ("Enter an integer : ");  
        int ch = input.nextInt();  
        switch (ch) {  
            case 1:  
                Employee e1 = new Employee();  
                e1.Emp-name = "ABC";  
                e1.Address = "Y-city";  
                e1.Mail_id = "prav@gmail.com";  
                e1.Emp_id = 567;
```

```
e1. Mobile-no = 2345678;  
p1. BP = 15000;  
p1. display();  
e1. display();  
break;
```

Case 2 :

```
Employee e2 = new Employee();  
Assistant-professor p2 = new Assistant-professor();  
e2. Emp-name = "DEF";  
e2. Address = "A-city";  
e2. Mail-id = "SRI@gmail.com";  
e2. Emp-id = 123;  
e2. Mobile-no = 987321;  
p2. BP = 30000;  
p2. display();  
e2. display();  
break;
```

Case 3 :

```
Employee e3 = new Employee();  
Associate-professor p3 = new Associate-professor();  
e3. Emp-name = "GHI";  
e3. Address = "B-city";  
e3. Mail-id = "MAJN@gmail.com";  
e3. Emp-id = 456;  
e3. Mobile-no = 98710;  
p3. BP = 30000;  
p3. display();  
e3. display();  
break;
```

case 4 :

```
Employee e4=new Employee();
professor p4=new professor();
e4.Emp-name = "KANNAN";
e4.Address = "TRICHY";
e4.Mail-id = "kanna@gmail.com";
e4.Emp-id = 789;
e4.Mobile-no = 9810;
p4.BP = 30000;
p4.display();
e4.display();
break;
```

case 5

```
/exit(1);
default :
S.O.P("enter correct choice");
}
```

}

3

Output:

enter an integer : 4

1. programme
2. Assistant-professor
3. Associate-professor
4. professor

Enter an integer : 30000

DA 29100.0

HRA 3000.0

PF 3600.0

S A I F F C L U D F U N D 30.0

KANNAN

789

kanna@gmail.com
9810

10] Rational experiment

```
import java.io.*;
import java.util.*;
class Rational {
    int numerator;
    int denominator;
    void readInput() {
        Scanner in = new Scanner(System.in);
        System.out.println("Enter the numerator");
        numerator = in.nextInt();
        System.out.println("Enter the denominator");
        denominator = in.nextInt();
    }
    int calculate() {
        int i;
        int k = (numerator < denominator) ? numerator : denominator;
        for (i = k; i > 1; i--) {
            if ((numerator % i == 0) && (denominator % i == 0))
                break;
        }
        return i;
    }
    void displayResult(int gcd) {
        if (numerator == 0)
            System.out.println("Numerator can't be zero it is invalid");
    }
}
```

```
else if(denominator == 0)
{
    System.out.println("Denominator can't be zero it is
                        invalid");
}
```

```
else
{
    numerator = numerator / gcd;
    denominator = denominator / gcd;
    S.O.P("The resultant value is :" + numerator + "/" + denominator);
}
```

```
}
```

```
class Main {
```

```
    public static void main(String args[])
    {
        Rational ra = new Rational();
        ra.readInput();
        int gcd = ra.calculate();
        ra.displayResult(gcd);
    }
}
```

Output : ①

enter the numerator 35

enter the denominator 5

The resultant value is : 7/1

O/P : ②

enter the numerator 11

enter the denominator 6

The resultant value is : 11/6.

```

②
import java.util.*;
public class ExceptionHandling {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        try {
            S.O.P("enter any two values");
            int a = SC.nextInt();
            int b = SC.nextInt();
            int c;
            c = a / b;
        } catch(ArithmeticException ae) {
            S.O.P("Arithmatic exception occurred");
        } catch(InputMismatchException ie) {
            S.O.P("Input Mismatch exception occurred");
        }
    }
}

```

Output:

45

You shouldn't divide a number by zero
 Enter any integers you entered 45

```
1 package pack;
import java.util.*;
public class Timer {
    int hours, seconds, minutes;
    int input;
    Scanner sc = new Scanner(System.in);
    Scanner sc = new Scanner(System.in);
    public void SecondsToHours() {
        S.O.P("Enter seconds to convert into
hours");
        input = sc.nextInt();
        hours = input / 360;
        minutes = input % 360 / 60;
        S.O.P(input + "seconds equal to " + hours
+ "hours");
    }
    public void hoursToMinutes() {
        S.O.P("Enter hours to convert into minutes");
        input = sc.nextInt();
        minutes = input * 60;
        S.O.P(input + "hours equal to " + minutes
+ "minutes");
    }
    public void hoursToSeconds() {
        S.O.P("Enter hours to convert into seconds");
        input = sc.nextInt();
        minutes = input * 60;
        S.O.P(input + "hours equal to " + minutes
+ "seconds");
    }
    public void of HoursToSeconds() {
        S.O.P("Enter hours to convert into seconds");
        input = sc.nextInt();
        seconds = input * 3600;
    }
}
```

S.O.P("Input + "hours equal to " + seconds + "seconds")

{
public void minutesToHours(){

S.O.P("Enter minutes to convert into hours");
input = SC.nextInt();
hours = input/60;
S.O.P(input + "minutes equals to " + hours + "hours");

{
}

{
}

Output:
=

```
④ package pack;  
import java.util.Scanner;  
public class DistanceConverter {  
    Scanner sc = new Scanner(System.in);  
    double km, meter, mile, input;  
    public void metersToKm() {  
        input = sc.nextDouble();  
        km = input * 0.001;  
        System.out.println("meters = " + km + "kilometers");  
    }  
    public void milesToKm() {  
        input = sc.nextDouble();  
        km = input * 1.609344;  
        System.out.println("miles = " + km + "kilometers");  
    }  
    public void kmToMeters() {  
        input = sc.nextDouble();  
        meter = input / 0.001;  
        System.out.println("kilometers = " + meter + "meters");  
    }  
    public void kmToMiles() {  
        input = sc.nextDouble();  
        km = input / 1.609344;  
        System.out.println("kilometers = " + km + "miles");  
    }  
}
```

```
[u] import java.util.*;  
class Square extends Thread {  
    int x;  
    public Square (int n) {  
        x = n;  
    }  
    public void run () {  
        int Sqr = x * x;  
        System.out.println ("Square of " + x + " = " + Sqr);  
    }  
}  
class Cube extends Thread {  
    int x;  
    public Cube (int n) {  
        x = n;  
    }  
    public void run () {  
        int cub = x * x * x;  
        System.out.println ("Cube of " + x + " = " + cub);  
    }  
}  
class Number extends Thread {  
    public void run () {  
        Random r = new Random();  
        for (int i = 0; i < 11; i++) {  
            int ran = r.nextInt (100);  
            System.out.println ("Random is generated : " + ran);  
            Square s = new Square (ran);  
            s.start();  
        }  
    }  
}
```

```

cube sk = new Cube (rand);
if (rand % 2 == 0) {
    sl.start();
}
else {
    sl.start();
}
try {
    thread.sleep(1000);
}
catch (Exception e) {
    s.o.p(e);
}
}
}

```

```

public class ThreeThreads {
    public static void main(String[] args) {
        Number nl = new Number();
        nl.start();
    }
}

```

Output: Random is generated : 83
 Cube of 83 = 571787
 Random is generated : 25
 cube of 25 = 15625
 Random is generated : 42
 cube of 42 = 74088
 Random is generated : 68
 cube of 68 = 314492

Random is generated : 14
 cube of 14 = 2744
 Random is generated : 24
 cube of 24 = 13824
 Random is generated : 38
 cube of 38 = 42875
 Random is generated : 71
 cube of 71 = 35794
 Random is generated : 34
 cube of 34 = 39304

```


    package pack;
    import java.util.Scanner;
    public class Currency {
        Scanner sc = new Scanner(System.in);
        double INR, USD, EURO, YEN;
        public void dollarToRupee() {
            System.out.println("Enter how many dollars");
            USD = sc.nextDouble();
            INR = USD * 73;
            System.out.println("Dollar=" + USD + "equal to INR=" + INR);
        }
        public void euroToRupee() {
            System.out.println("enter how many euros to convert");
            EURO = sc.nextDouble();
            INR = EURO * 90;
            System.out.println("EURO=" + EURO + "equals to " + INR + "INR");
        }
        public void yenToRupee() {
            System.out.println("enter how many YEN's to convert");
            YEN = sc.nextDouble();
            INR = YEN * 0.71;
            System.out.println("YEN=" + YEN + "equals to " + INR + "INR");
        }
        Random is generated: 39 public void rupeeToDollar() {
            System.out.println("Enter how many Rupees to convert");
            INR = sc.nextDouble();
            USD = INR / 73;
            System.out.println("INR=" + INR + "equal to Dollar=" + USD);
        }
    }


```

```
public void rupeeToEuro() {  
    S.o.p("Enter Rupees to convert into Euro:");  
    INR = Sc.nextDouble();  
    EURO = INR / 90;  
    S.o.p("Rupee = " + INR + " equal to EURO = " + EURO);  
}
```

```
3  
public void rupeeToYen() {  
    S.o.p("Enter Rupees to convert into Yen:");  
    INR = Sc.nextDouble();  
    YEN = (INR) / 0.71;  
    S.o.p("INR = " + INR + " equal to YEN = " + YEN);  
}
```

3
Output: