This document showcases list of tasks and solutions that I developed while learning Java Programming.

Work Done By: Jeyakumar N K

---

1. Electricity Bill Generation

Task Description:

To develop a Java application to generate Electricity bill. Create a class with the following members: Consumer no., consumer name, previous month reading, current month reading, type of EB connection (i.e domestic or commercial). Compute the bill amount using the following tariff.

If the type of the EB connection is domestic, calculate the amount to bepaid as follows:

- ➢ First 100 units - Rs. 1 per unit
- ➢ 101-200 units - Rs. 2.50 per unit
- ➢ 201 -500 units - Rs. 4 per unit
- ➢ > 501 units - Rs. 6 per unit

If the type of the EB connection is commercial, calculate the amount tobe paid as follows:

- ➢ First 100 units - Rs. 2 per unit
- ➢ 101-200 units - Rs. 4.50 per unit
- ➢ 201 -500 units - Rs. 6 per unit
- ➢ > 501 units - Rs. 7 per unit

Solution Developed:

```
import java.util.*;
class Ebill
 {
        public static void main (String args[])
         {
        Customerdata ob = new Customerdata();
        ob.getdata();
        ob.calc();
        ob.display();

        }
}

 class Customerdata
 {
        Scanner in = new Scanner(System.in);
        Scanner ins = new Scanner(System.in);
        String cname,type;
        int bn;
```

```
    double current,previous,tbill,units;
    void getdata()
     {
        System.out.print ("\n\t Enter consumer number ");
        bn = in.nextInt();
        System.out.print ("\n\t Enter Type of connection (D for Domestic or
        C for Commercial) "); type = ins.nextLine();
        System.out.print ("\n\t Enter consumer name ");
        cname = ins.nextLine();
        System.out.print ("\n\t Enter previous month reading ");
        previous= in.nextDouble();
        System.out.print ("\n\t Enter current month reading ");
        current= in.nextDouble();
     }

    void calc()
     {
        units=current-previous;
        if(type.equals("D"))
         {
            if (units<=100)
                tbill=1 * units;
            else if (units>100 && units<=200)
                tbill=2.50*units;
            else if(units>200 && units<=500)
                tbill= 4*units;
            else
                tbill= 6*units;
         }
        else
         {

            if (units<=100)
                tbill= 2 * units;
            else if(units>100 && units<=200)
                tbill=4.50*units;
            else if(units>200 && units<=500)
                tbill= 6*units;
            else }
                 tbill= 7*units;


}

    void display()
     {
        System.out.println("\n\t Consumer number = "+bn);
        System.out.println ("\n\t Consumer name = "+cname);
        if(type.equals("D"))
            System.out.println ("\n\t type of connection = DOMESTIC ");
        else
            System.out.println("\n\t type of connection = COMMERCIAL ");
            System.out.println ("\n\t Current Month Reading = "+current);
            System.out.println("n\t Previous Month Reading = "+previous);
            System.out.println ("\n\t Total units = "+units);
```

```
                        System.out.println ("\n\t Total bill = RS "+tbill);
        }
}
```

Output:

```
D:\ >javac Ebill.java
D:\ >java Ebill

  Enter consumer number 2132
  Enter Type of connection (D for Domestic or C for Commercial)
  D Enter consumer name pager
  Enter previous month reading 3000
  Enter current month reading 4000
  Consumer number = 2132
  Consumer name = Jeyakumar
  type of connection = DOMESTIC Current
  Month Reading = 4000.0
  Previous Month Reading = 3000.0
  Total units = 1000.0
  Total bill = RS 6000.0
```

2. Currency Converter, Distance Converter and Time Converter Implementation Using Packages

Task Description:

To develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa) , time converter (hours to minutes, seconds and vice versa) using packages.

Program:

currency.java

```
package currencyconversion;
import java.util.*;
public class currency
{
        double inr,usd;
        double euro,yen;
        Scanner in=new Scanner(System.in);
        public void dollartorupee()
        {
                System.out.println("Enter dollars to convert into Rupees:");
                usd=in.nextInt();
                inr=usd*67;
                System.out.println("Dollar ="+usd+"equalto INR="+inr);
        }


 public void rupeetodollar()
        {
        System.out.println("Enter Rupee to convert into
        Dollars:"); inr=in.nextInt();
        usd=inr/67;
        System.out.println("Rupee ="+inr+"equal to
        Dollars="+usd); }
        public void eurotorupee()
        {
                System.out.println("Enter euro to convert into
                Rupees:"); euro=in.nextInt();
                inr=euro*79.50;
                System.out.println("Euro ="+euro +"equal to
        INR="+inr); }
        public void rupeetoeuro()
        {
                System.out.println("Enter Rupees to convert into
                Euro:"); inr=in.nextInt();
                euro=(inr/79.50);
                System.out.println("Rupee ="+inr +"equal to
        Euro="+euro); }


        public void yentorupee()
        {
                System.out.println("Enter yen to convert into
                Rupees:"); yen=in.nextInt();
                inr=yen*0.61;
                System.out.println("YEN="+yen +"equal to
        INR="+inr); }
        public void rupeetoyen()
         {
                System.out.println("Enter Rupees to convert into
                Yen:"); inr=in.nextInt();
                yen=(inr/0.61);
                System.out.println("INR="+inr +"equal to YEN"+yen);
         }
 }
```

distance.java

```
package distanceconversion;
import java.util.*;
public class distance
 {
        double km,m,miles;
        Scanner sc = new Scanner(System.in);
        public void kmtom()
         {
                System.out.print("Enter in km ");
                km=sc.nextDouble();


                m=(km*1000);
                System.out.println(km+"km" +"equal
         to"+m+"metres"); }
        public void mtokm()
         {
                System.out.print("Enter in meter ");
                m=sc.nextDouble();
                km=(m/1000);
                System.out.println(m+"m"
         +"equalto"+km+"kilometres"); }
        public void milestokm()
         {
                System.out.print("Enter in miles");
                miles=sc.nextDouble();
                km=(miles*1.60934);
                System.out.println(miles+"miles"
         +"equalto"+km+"kilometres"); }
        public void kmtomiles()
         {
                System.out.print("Enter in km");
                km=sc.nextDouble();
                miles=(km*0.621371);
                System.out.println(km+"km"
         +"equalto"+miles+"miles"); }
 }
```

timer.java

```
package timeconversion;
import java.util.*;
public class timer
 {
        int hours,seconds,minutes;
        int input;
        Scanner sc = new Scanner(System.in);
        public void secondstohours()
         {
                System.out.print("Enter the number of seconds: ");
                input = sc.nextInt();

                hours= input/ 3600;
                minutes = (input % 3600) / 60;
                seconds = (input % 3600) % 60;
                System.out.println("Hours:"+ hours);
                System.out.println("Minutes: " + minutes);
                System.out.println("Seconds:"+ seconds);
         }
        public void minutestohours()
         {
                System.out.print("Enter the number of minutes: ");


                minutes=sc.nextInt();
                hours=minutes/60;
                minutes=minutes%60;
                System.out.println("Hours:"+ hours);
                System.out.println("Minutes:"+ minutes);
         }
        public void hourstominutes()
         {
                System.out.println("enter the no of hours");
                hours=sc.nextInt();
                minutes=(hours*60);
                System.out.println("Minutes:"+ minutes);
         }
        public void hourstoseconds()
         {
                System.out.println("enter the no of hours");
                hours=sc.nextInt();
                seconds=(hours*3600);
                System.out.println("Minutes:"+ seconds);
         }
 }
```

converter.java

```
import java.util.*;
import java.io.*;
import currencyconversion.*;
import distanceconversion.*;
import timeconversion.*;
class converter
 {
        public static void main(String args[])
         {
        Scanner s=new Scanner(System.in);
        int choice,ch;
        currency c=new currency();
        distance d=new distance();
        timer t=new timer();
```

```
do
    {
            System.out.println("1.dollar to rupee ");
            System.out.println("2.rupee to dollar ");
            System.out.println("3.Euro to rupee ");
            System.out.println("4..rupee to Euro ");
            System.out.println("5.Yen to rupee ");
            System.out.println("6.Rupee to Yen ");
            System.out.println("7.Meter to kilometer ");
            System.out.println("8.kilometer to meter ");
            System.out.println("9.Miles to kilometer ");
            System.out.println("10.kilometer to miles");
            System.out.println("11.Hours to Minutes");
            System.out.println("12.Hours to Seconds");
            System.out.println("13.Seconds to Hours");
            System.out.println("14.Minutes to Hours");
            System.out.println("Enter your choice");
            choice=s.nextInt();

            switch(choice) {

            case 1: c.dollartorupee(); break;

            case 2: c.rupeetodollar(); break;

            case 3: c.eurotorupee(); break;

            case 4: c.rupeetoeuro(); break;

            case 5: c.yentorupee(); break;

            case 6: c.rupeetoyen(); break;

            case 7: d.mtokm(); break;

            case 8: d.kmtom(); break;

            case 9: d.milestokm(); break;

            case 10: d.kmtomiles(); break;

            case 11: t.hourstominutes(); break;

            case 12: t.hourstoseconds(); break;

            case 13: t.secondstohours(); break;

            case 14: t.minutestohours(); break;

     }

    System.out.println("Enter 0 to quit and 1 to continue

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

```
        }while(ch==1);
    }
}
```

Output:

```
E:\PROGRAMS>javac.converter.java
E:\PROGRAMS>java converter
 1.dollar to rupee
 2.rupee to dollar
 3.Euro to rupee
 4.rupee to Euro
 5.Yen to rupee
 6.Rupee to Yen
 7.Meter to kilometer
 8.kilometer to meter
 9.Miles to kilometer
 10.kilometer to miles
 11.Hours to Minutes
 12.Hours to Seconds
 13.Seconds to Hours
 14.Minutes to Hours
 Enter your choice: 1
 Enter dollars to convert into Rupees:
 500 Dollar =500.0 equal to INR=33500.0
 Enter 0 to quit and 1 to continue: 1
 1.dollar to rupee
 2.rupee to dollar
 3.Euro to rupee
 4.rupee to Euro
```

```
5.Yen to rupee

6.Rupee to Yen

7.Meter to kilometer

8.kilometer to meter

9.Miles to kilometer

10.kilometer to miles

11.Hours to Minutes

12.Hours to Seconds

13.Seconds to Hours

14.Minutes to Hours

Enter your choice 8

Enter in km 2

2.0 km equal to 2000.0 meters

Enter 0 to quit and 1 to continue: 0
```

3. Pay Slip Generation Using Inheritance

Task Description:

To develop a java application to generate pay slip for different category of employees using the concept of **inheritance**.

Program:

Salary.java

```
import java.util.*;
class Employee
{
        int empid;
        long mobile;
        String name, address, mailid;
        Scanner get = new Scanner(System.in);
        void getdata()
        {
                System.out.println("Enter Name of the Employee");
                name = get.nextLine();
                System.out.println("Enter Mail id");
                mailid = get.nextLine();
                System.out.println("Enter Address of the Employee:");
                address = get.nextLine();
                System.out.println("Enter employee id ");
                empid = get.nextInt();
                System.out.println("Enter Mobile Number");
                mobile = get.nextLong();
        }
        void display()
        {
                System.out.println("Employee Name: "+name);
                System.out.println("Employee id: "+empid);
                System.out.println("Mail id: "+mailid);
                System.out.println("Address: "+address);
                System.out.println("Mobile Number: "+mobile);
        }
}

class Programmer extends Employee
{
        double salary,bp,da,hra,pf,club,net,gross;
        void getprogrammer()
        {
                System.out.println("Enter basic pay");
                bp = get.nextDouble();
        }


        void calculateprog()
        {
                da=(0.97*bp);
                hra=(0.10*bp);
                pf=(0.12*bp);
                club=(0.1*bp);
                gross=(bp+da+hra);
                net=(gross-pf-club);
                System.out.println("********************************
                * ***********"); System.out.println("PAY SLIP FOR
                PROGRAMMER");
                System.out.println("********************************
                * ***********"); System.out.println("Basic Pay: Rs. "+bp);
                System.out.println("DA: Rs. "+da);
                System.out.println("HRA: Rs. "+hra);
                System.out.println("PF: Rs. "+pf);
                System.out.println("CLUB: Rs. "+club);
                System.out.println("GROSS PAY: Rs. "+gross);
                System.out.println("NET PAY: Rs. "+net);
        }
    }
class Asstprofessor extends Employee
{
        double salary,bp,da,hra,pf,club,net,gross;
        void getasst()
        {
                System.out.println("Enter basic pay");
                bp = get.nextDouble();
        }
        void calculateasst()
        {
        da=(0.97*bp); hra=(0.10*bp); pf=(0.12*bp); club=(0.1*bp);
        gross=(bp+da+hra); net=(gross-pf-club);
                System.out.println("***********************************");
                System.out.println("PAY SLIP FOR ASSISTANT
                PROFESSOR");
                System.out.println("***********************************");
                System.out.println("Basic Pay: Rs. "+bp);
                System.out.println("DA: Rs. "+da);
                System.out.println("HRA: Rs. "+hra);
                System.out.println("PF: Rs. "+pf);
                System.out.println("CLUB: Rs. "+club);
                System.out.println("GROSS PAY: Rs. "+gross);
                System.out.println("NET PAY: Rs. "+net);
        }
    }
class Associateprofessor extends Employee
{
        double salary,bp,da,hra,pf,club,net,gross;
        void getassociate()
        {
                System.out.println("Enter basic pay");
                bp = get.nextDouble();
        }
        void calculateassociate()
        {
                da=(0.97*bp);
```

```
                hra=(0.10*bp);
                pf=(0.12*bp);
                club=(0.1*bp);
                gross=(bp+da+hra);
                net=(gross-pf-club);

                System.out.println("***********************************");
                System.out.println("PAY SLIP FOR ASSOCIATE
                PROFESSOR");
                System.out.println("***********************************");
                System.out.println("Basic Pay: Rs. "+bp);
                System.out.println("DA: Rs. "+da);
                System.out.println("HRA: Rs. "+hra);
                System.out.println("PF: Rs. "+pf);
                System.out.println("CLUB: Rs. "+club);
                System.out.println("GROSS PAY: Rs. "+gross);
                System.out.println("NET PAY: Rs. "+net);
         }
 }

class Professor extends Employee
{
        double salary,bp,da,hra,pf,club,net,gross;
        void getprofessor()
         {
                System.out.println("Enter basic pay");
                bp = get.nextDouble();
         }
        void calculateprofessor()
         {
                da=(0.97*bp);
                hra=(0.10*bp);
                pf=(0.12*bp);
                club=(0.1*bp);
                gross=(bp+da+hra);
                net=(gross-pf-club);
                System.out.println("************************");
                System.out.println("PAY SLIP FOR PROFESSOR");
                System.out.println("************************");
                System.out.println("Basic Pay: Rs. "+bp);
                System.out.println("DA: Rs. "+da);
                System.out.println("HRA: Rs. "+hra);
                System.out.println("PF: Rs. "+pf);
                System.out.println("CLUB: Rs. "+club);
                System.out.println("GROSS PAY: Rs. "+gross);
                System.out.println("NET PAY: Rs. "+net);
         }
 }
class Salary
{
        public static void main(String args[])
         {
                int choice,cont;
                do
                {
                System.out.println("PAYROLL");
                        System.out.println(" 1.PROGRAMMER \t 2.ASSISTANT
                  PROFESSOR \t 3.ASSOCIATE PROFESSOR \t 4.PROFESSOR ");
                Scanner c = new Scanner(System.in);
                System.out.print("Enter Your Choice:");
                choice=c.nextInt();
                switch(choice)
                {
                        case 1:
                        {
                                Programmer p=new Programmer();
                                p.getdata();
                                p.getprogrammer();
                                p.display();
                                p.calculateprog();
                                break;
                              }

                              case 2:
                              {

                                      Asstprofessor asst=new
                                      Asstprofessor(); asst.getdata();
                                      asst.getasst();
                                      asst.display();
                                      asst.calculateasst();
                                                          break;
                        }
                        case 3:
                        {

                                                          Associateprofessor asso=new
                                                          Associateprofessor();
                                                          asso.getdata();
                                                          asso.getassociate();
                                                          asso.display();
                                                          asso.calculateassociate();
                        }
                                                          break;
                        case 4:
                        {


                                                          Professor prof=new Professor();
                                                          prof.getdata();
                                                          prof.getprofessor();
                                                          prof.display();
                                                          prof.calculateprofessor();
                        }
                                                          break;

                        }



                       System.out.print("Please enter 0 to quit and 1 to continue: ");
                        cont=c.nextInt();
                }while(cont==1);
        }
```

```
}
```

Output:

```
1. PROGRAMMER
2. ASSISTANT PROFESSOR
3. ASSOCIATE PROFESSOR
4. PROFESSOR
Enter Your Choice: 2
Enter Name of the Employee: Jeyakumar N K
Enter Mail id: 2025vcetitb32@gmail.com
Enter Address of the Employee: 3117,TNHB Colony, Madurai-
09. Enter employee id: 2132
Enter Mobile Number: 9488465168
Enter basic pay: 20000
Employee Name: Jeyakumar N K
Employee id: 2132
Mail id: 2025vcetitb32@gmail.com
Address: 3117,TNHB Colony, Madurai-09.
Mobile Number: 9876543210

***********************************
PAY SLIP FOR ASSISTANT PROFESSOR
***********************************

   Basic Pay:Rs. 20000.0
   DA:Rs. 19400.0
   HRA:Rs. 2000.0
   PF:Rs. 2400.0
   CLUB:Rs. 2000.0
   GROSS PAY:Rs. 41400.0
   NET PAY:Rs. 37000.0
   Please enter 0 to quit and 1 to continue: 0
```

4. Stack ADT Implementation Using Inheritance (Interface)

Task Description:

To design a Java application to implement array implementation of stack using the concept of **Interface** and **Exception handling**.

Program:

Teststack.java

```
import java.io.*;
interface Stackoperation
{
          public void push(int i);
          public void pop();
}

class Astack implements Stackoperation
{
         int stack[]=new int[5];
         int top=-1;
         int i;
         public void push(int item)
          {
                  if(top>=4)
                   {
                           System.out.println("Overflow");
                   }
                  else
                   {
                  top=top+1; stack[top]=item;
                           System.out.print("Element pushed: "+stack[top]);
                   }
          }

         public void pop()
          {
                  if(top<0)
                           System.out.println("Underflow");
                  else
                   {
                           System.out.print("Element popped: "+stack[top]);
                          top=top-1;
                   }
          }


         public void display()
          {
                  if(top<0)
                           System.out.println("No Element in stack");
                  else
```

```
                    {
                            for(i=0;i<=top;i++)
                            System.out.println("Element:"+stack[i]);
                    }
            }
    }


    class Teststack
    {
            public static void main(String args[]) throws
             IOException {
                    int ch,c;
                    int i;
                    Astack s=new Astack();
                    DataInputStream in=new DataInputStream(System.in);
                    do
                    {
                      try
                      {
                            System.out.println("ARRAY STACK");
                            System.out.println("1.Push 2.Pop 3.Display 4.Exit");
                            System.out.print("Enter your Choice:");
                            ch=Integer.parseInt(in.readLine());
                            switch(ch)
                            {
                            case 1:
                                                            the value to push:");
                                                            i=Integer.parseInt(in.read
                                                            Line());
                                                            s.push(i);
                                                            break;
                          case 2:

                                                            s.pop();
                                                            break;
                          case 3:

                                                            System.out.println("The
                                                            elements are: ");
                                                            s.display();
                                                            break;
                          default:
                          System.out.print("Enter


                                  break;
                          }
                  }

                catch(IOException e)
                {
                          System.out.println("IO Error");
                }
                System.out.println("Please enter 0 to quit and 1 to continue ");
                c=Integer.parseInt(in.readLine());
        }while(c==1);
}
```

```
}
```

Output:

```
D:\ >javac Teststack.java D:\>

java Teststack
 ARRAY STACK
 1.Push
 2.Pop
 3.Display
 4.Exit
 Enter your Choice: 1
 Enter the value to push: 10 Element
 pushed: 10
 Please enter 0 to quit and 1 to continue: 1

 ARRAY STACK
 1.Push
 2.Pop
 3.Display
 4.Exit
 Enter your Choice: 1
 Enter the value to push: 20
 Element pushed: 20
 Please enter 0 to quit and 1 to continue: 1

  ARRAY STACK
  1.Push
  2.Pop
  3.Display
  4.Exit
  Enter your Choice: 1
  Enter the value to push: 30
  Element pushed: 30
  Please enter 0 to quit and 1 to continue: 1

  ARRAY STACK
  1.Push
  2.Pop
  3.Display
  4.Exit Enter your Choice: 3
  The elements are:
  Element: 10
  Element: 20
  Element: 30
  Please enter 0 to quit and 1 to continue: 1

  ARRAY STACK
  1.Push
  2.Pop
  3.Display
  4.Exit Enter your Choice: 2
```

```
Element popped: 30
Please enter 0 to quit and 1 to continue: 1

ARRAY STACK
1.Push
2.Pop
3.Display
4.Exit Enter your Choice: 3
The elements are:
Element:10
Element:20
Please enter 0 to quit and 1 to continue: 0
```

5. String Operations Using Array List

Task Description:

To write a java program to perform string operations using ArrayList for the following functions:

a. Append add at end
b. Insert - add at particular index
c. Search
d. List all string starts with given letter

Program:

<u>Arraylistexample.java</u>

```
import java.util.*;
import java.io.*;
public class Arraylistexample
{
public static void main (String args[]) throws IOException
{
   ArrayList <String> obj = new ArrayList <String>();
   DataInputStream in = new DataInputStream(System.in);
   int c,ch;
   int i,j;
   String str,str1;
   do
   {
      System.out.println("STRING MANIPULATION");
      System.out.println("*************************");
      System.out.println("1.Append at end\t2.Insert at particular index\t3.Search\t");
      System.out.println("4.List string that starting with letter\t");
      System.out.println("5.Size\t6.Remove\t7.Sort\t8.Display\t");
      System.out.println("Enter the choice: ");
      c=Integer.parseInt(in.readLine());
      switch(c)
      {
         case 1:
         {
            System.out.println("Enter the String: ");
            str=in.readLine();
            obj.add(str);
            break;
         }
         case 2:
         {
            System.out.println("Enter the string: ");
            str = in.readLine();
            System.out.println("Specify the index position to insert: ");
            i=Integer.parseInt(in.readLine());
            obj.add(i-1,str);
            System.out.println("The array list has following elements:"+obj);
            break;
            }
            case 3:
            {
                System.out.println("Enter the string to search: ");
                str=in.readLine();
                j=obj.indexOf(str);
                if(j==-1)
                    System.out.println("Element not found");
                else
                    System.out.println("Index of: "+str+"is"+j);
                break;
            }
            case 4:
            {
                System.out.println("Enter the character to List string that starts with sp
character:");
                str=in.readLine();
                for(i=0;i<(obj.size()-1);i++)
                {
                    str1 = obj.get(i);
                    if(str1.startsWith(str))
                    {
                        System.out.println(str1);
                    }
                }
                break;
            }
            case 5:
            {
                System.out.println("Size of the list"+obj.size());
                break;
            }
            case 6:
            {
                System.out.println("Enter the element to remove: ");
                str=in.readLine();
                if(obj.remove(str))
                {
                    System.out.println("Element removed"+str);
                }
                else
                {
                    System.out.println("Element not present");
                }
                break;
            }
            case 7:
            {
                Collections.sort(obj);
                System.out.println("The array list has following elements:"+obj);
                break;
            }
            case 8:
            {
                System.out.println("The array list has following elements: "+obj);
                break;
              }
           }
          System.out.println("Please Enter 0 to break and 1 to continue");
          ch=Integer.parseInt(in.readLine());
       }while(ch==1)
              }
    }
```

Output:

```
D:Java Programs javac Arraylistexample.java
D:\Java Programs java Arraylistexample

STRING MANIPULATION
*******************
1. Append at end 2.Insert at particular index 3.Search 4. List
string that starting with letter
5.Size 6. Remove 7.Sort 8.Display
Enter the choice: 1
Enter the string: FIRST
Enter 0 to break and I to continue: 1

STRING MANIPULATION
************************
1. Append at end 2.Insert at particular index 3.Search 4. List
string that starting with letter
5.Size 6. Remove 7.Sort 8.Display
Enter the choice: 1
Enter the string: LAST
Enter 0 to break and 1 to continue: 1

STRING MANIPULATION
**********************
1. Append at end 2.Insert at particular index 3.Search 4. List
string that starting with letter
5.Size 6. Remove 7.Sort 8.Display
Enter the choice: 8
The array list has following elements:[FIRST, LAST]
Enter 0 to break and 1 to continue: 1

STRING MANIPULATION
************************
1. Append at end 2.Insert at particular index 3.Search 4. List
string that starting with letter
5.Size 6. Remove 7.Sort 8.Display
Enter the choice: 2
Enter the string: SECOND
Specify the index/position to insert: 1
The array list has following elements:[SECOND, FIRST, LAST]
Enter 0 to break and 1 to continue: 1

STRING MANIPULATION
************************
1. Append at end 2.Insert at particular index 3.Search 4. List
```

```
string that starting with letter
5.Size 6. Remove 7.Sort 8.Display
Enter the choice: 3
Index of: LAST is 2
Enter 0 to break and 1 to continue: 1

STRING MANIPULATION
************************
1. Append at end 2.Insert at particular index 3.Search 4. List
string that starting with letter
5.Size 6. Remove 7.Sort 8.Display
Enter the choice: 5
Size of the list 3
Enter 0 to break and 1 to continue: 1


STRING MANIPULATION
************************
1. Append at end 2.Insert at particular index 3.Search 4. List
string that starting with letter
5.Size 6. Remove 7.Sort 8.Display
Enter the choice: 7
The array list has following elements:[FIRST, LAST, SECOND]
Enter 0 to break and 1 to continue: 0
```

6. Abstract Class Implementation

Task Description:

To write a Java program to calculate the area of rectangle, circle and triangle using the concept of abstract class.

Program:

Shapeclass.java

```
import java.util.*;
abstract class shape
{
 int a,b;
 abstract public void printarea();
 }

class rectangle extends shape
{
 public int area_rect;
 public void printarea()
 {
 Scanner s = new Scanner(System.in);
 System.out.println("Enter the length and breadth of rectangle");   a =
s.nextInt();
 b = s.nextInt();
 area_rect-a*b;
 System.out.println("Length of rectangle: "+a+"breadth of rectangle: "+b);
System.out.println("The area of rectangle is:"+area_rect);
 }
 }

class triangle extends shape
{
 double area tri;
 public void printarea()
 {

 Scanner s-new Scanner(System.in);
 System.out.println("Enter the base and height of triangle:");  a =
s.nextInt();
 b = s.nextInt();
 System.out.println("Base of triangle: "+a+"height of triangle: "+b);  area_tri =
(0.5*a*b);
 System.out.println("The area of triangle is:"+area_tri);  }
 }

class circle extends shape
{
 double area_circle;
 public void printarea()
 {
```

```
Scanner s-new Scanner(System.in); System.out.println("Enter the radius of  circle:");
a = s.nextInt();
area_circle (3.14*a*a);
System.out.println("Radius of circle:"+a);
System.out.println("The area of circle is:"+area_circle);  }
}

public class Shapeclass
{
public static void main(String[] args)
{
rectangle r = new rectangle();
r.printarea();
triangle t = new triangle();
t.printarea();
circle rl = new circle();
r1.printarea();
}
}
```

Output:

```
D: Java Programs javac Shapeclass.java
D: Java Programs java Shapeclass

Enter the length and breadth of rectangle:
2
3
Length of rectangle: 2 breadth of rectangle: 3
The area of rectangle is:6

Enter the base and height of triangle:
5
6
Base of triangle: 5 height of triangle: 6
The area of triangle is: 15.0

Enter the radius of circle
4
Radius of circle: 4
The area of circle is:50.24
```

7. User Defined Exception Handling Implementation

Task Description:

To write a Java program to implement user defined exception handling.

Program 1:

userdefined.java

```
import java.util.*;
class NegativeAmtException extends Exception
 {
  String msg;
  NegativeAmtException(String msg)
  {
  this.msg = msg;
  }
  public String toString()
  {
  return msg;
  }
 }
public class userdefined
{
  public static void main(String[] args)
  {
  Scanner s = new Scanner(System.in);
  System.out.print("Enter Amount:");
  int a = s.nextInt();
  try
  {
if(a<0)
  {
  throw new NegativeAmtException("Invalid Amount");  }
  System.out.println("Amount Deposited");
  }
  catch(NegativeAmtException e)
  {
  System.out.println(e);
  }
  }
 }
```

Output:

```
Enter Amount: 5000
Amount Deposited
Enter Amount: -2000
 Invalid Amount
```

Program 2:

example.java

```
class MyException extends Exception
{
 String strl;
 MyException(String str2)
 {
 strl = str2;
 }
 public String toString()
 {
 return ("MyException Occurred: "+strl);
 }
 }
class example
{
 public static void main(String args[])
 {
 try
 {
 System.out.println("Starting of try block");   throw new
MyException("This is My error Message");  }
 catch(MyException exp)
 {
 System.out.println("Catch Block");
System.out.println(exp);
 }
 }
 }
```

Output:

```
Starting of try block
Catch Block
MyException Occurred: This is My error Message
```

8. File Handling

Task Description:

To write a java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file in bytes.

Program:

filedemo.java

```
import java.io.*;
import java.util.*;
class filedemo
{
 public static void main(String args[])
 {
 String filename;
 Scanner s = new Scanner(System.in);
 System.out.println(“Enter the file name “);
 filename = s.nextLine();
 File f1 = new File(filename);
 System.out.println(“*********************”);
 System.out.println("FILE INFORMATION");
 System.out.println(“*********************”);
 System.out.println("NAME OF THE FILE "+fl.getName());
System.out.println("PATH OF THE FILE "+fl.getPath());
System.out.println("PARENT"+fl.getParent());
 if(fl.exists())
 System.out.println("THE FILE EXISTS");
 else
 System.out.println("THE FILE DOES NOT EXISTS");
if(fl.canRead())
 System.out.println("THE FILE CAN BE READ ");  else
 System.out.println("THE FILE CANNOT BE READ ");   if
(fl.canWrite())
 System.out.println("WRITE OPERATION IS PERMITTED");  else
 System.out.println("WRITE OPERATION IS NOT PERMITTED");
if(fl.isDirectory())
 System.out.println("IT IS A DIRECTORY");
 else
 System.out.println("NOT A DIRECTORY");
 if(fl.isFile())
 System.out.println("IT IS A FILE");
 else
 System.out.println("NOT A FILE");
 System.out.println("File last modified fl.lastModified());
System.out.println("LENGTH OF THE FILE "+fl.length());
System.out.println("FILE DELETED "+fl.delete());
   }
 }
```

Output:

```
E:\PROGRAM>Java filedemo

  Enter the file name
  teststack.java
  ********************
  FILE INFORMATION
  ********************
  NAME OF THE FILE teststack.java
  PATH OF THE FILE teststack.java
  PARENT null
  THE FILE EXISTS
  THE FILE CAN BE READ
  WRITE OPERATION IS PERMITTED
  NOT A DIRECTORY
  IT IS A FILE
  File last modified 152138
  LENGTH OF THE FILE 150
  FILE DELETED true
```

9. Multithreading Implementation

Task Description:

To write a Java program to implements a multi-threaded application.

Program :

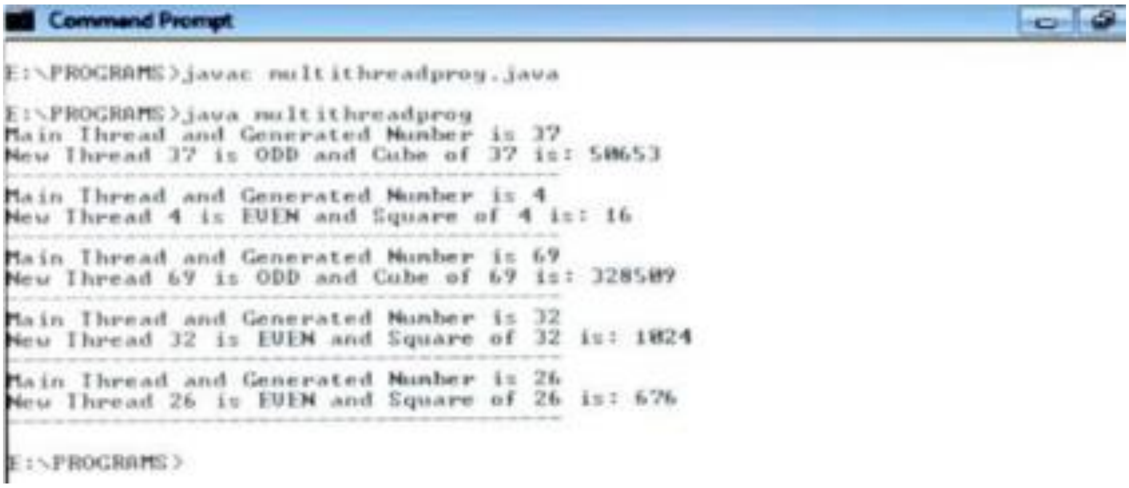
multithreadprog.java

```
import java.util. *;
class even implements Runnable
{
public int x;
public even(int x)
{ this.x = x;
}
public void
run() {
System.out.printIn("New Thread "+ x +" is EVEN and Square of " + x +"is:" +x * x);  }
}
class odd implements Runnable
{
public int x;
public odd(int x)
{ this.x = x; }
public void run()
{
System.out.printin("New Thread "+ x +" is ODD and Cube of "+ x+"is:"+X*X*X);  }

}
class A extends Thread
{
public void run()
{ int num = 0;
Random r = new Random();
try {
for (int i = 0: i < 5: i++)
{
num = r.nextint( 100);
System.out.printin("Main Thread and Generat is" + num) ;
if (num % 2 — 0)
{
Thread tl = new Thread(new even(num));
tl.start(); }
else
{
Thread t2 = new Thread(new odd(num));
t2. start();
}
Thread. sleep( 1000);
System.out.printin("—-—--------------------");  }
catch (Exception ex)
{
System.oul. printin(ex.getMessage());
}
}

public class multithreadprog
{
public static void main(String[] args)
{
 Aa=new A();
a.start();
}
}
```

Output:



```
E:\PROGRAMS>javac multithreadprog.java

E:\PROGRAMS>java multithreadprog
Main Thread and Generated Number is 37
New Thread 37 is ODD and Cube of 37 is: 50653
--------------------------------------------
Main Thread and Generated Number is 4
New Thread 4 is EVEN and Square of 4 is: 16
--------------------------------------------
Main Thread and Generated Number is 69
New Thread 69 is ODD and Cube of 69 is: 328509
--------------------------------------------
Main Thread and Generated Number is 32
New Thread 32 is EVEN and Square of 32 is: 1024
--------------------------------------------
Main Thread and Generated Number is 26
New Thread 26 is EVEN and Square of 26 is: 676
--------------------------------------------

E:\PROGRAMS>
```

10. Generic Function Implementation

Task Description:

To write a Java program to find the maximum value from the given type of elements using a generic function.

Program:

genericdemo.java

```
class MyClass<T extends Comparable<T>>
{
T[]vals;
MyClass(T[]o)
{
vals = o;
 }
public T min()
{
T v = vals[0];
for(int i=1; i<vals.length; i++)
 if(vals[i].compareTo(v) < 0)
 v= vals[i]; return v;
 }
public T max()
{
T v = vals[0];
for(int i=1; i<vals.length;i++)
if(vals[i].compareTo(v) > 0)
 v =vals[i];
return v;
 }
 }
class genericdemo
{
public static void main(String args[])
{
 int i;
 Integer inums[]={10,2,5,4,6,1};
 Character chs[]= {'v','p','s','a','n','h'} ;
 Double d[]={20.2,45.4,71.6,88.3,54.6,10.4};
 MyClass<Integer>iob = new MyClass<Integer>(inums);
 MyClass<Character> cob = new MyClass<Character>(chs);
 MyClass<Double>dob = new MyClass<Double>(d);
 System.out.println("Max value in inums: " + iob.max());
 System.out.println("Min value in inums: " + iob.min());
 System.out.println("Max value in chs: " + cob.max());
 System.out.println("Min value in chs: " + cob.min());
 System.out.println("Max value in chs: " + dob.max());
 System.out.println("Min value in chs: " + dob.min());
 }
 }
```

Output:

```
D:\>Java Prgs>javac genericdemo.java

D:\>Java Prgs>java genericdemo

   Max value in inums: 10

   Min value in inums: 1

   Max value in chs: v

   Min value in chs: a

   Max value in chs: 88.3

   Min value in chs: 10.4
```