Index

S.no	Particulars	Page no
1	**Write a 'java' Program to Demonstrate the concept of class Box with Constructors**	
2	**2.Write a java program to calculate salary of n employees using concept of classes with constructor and method?**	
3	**3.write a program to calculate students grade using class methods?**	
4	**Write a 'java' Program to Demonstrate the concept of Multilevel Inheritance**	
5	**Write a program to implement multilevel inheritance?**	
6	**Write a program to demonstrate abstract class and dynamic polymorphism?**	
7	**Write a program to implement packages?**	
8	**Write a program to demonstrate various arithmetic calculations using packages?**	
9	**Write a program to implement string handling methods?**	
10	**Write a program to implement Exceptional handling?**	
11	**Write a program to implement Multithreading?**	
12	**Write a program to demonstrate mutual exclusion using thread synchronization?**	
13	**Write a program to demonstrate Linked list class?**	
14	**Write a program to demonstrate Hash Set class?**	
15	**Write a program to demonstrate Iterater class?**	
16	**Write a program to demonstrate Enumeration interface?**	
17	**Write a program to demonstrate Comparator Interface?**	
18	**Write a program to implement string Tokenizer?**	
19	**Write a program to accept data and display output in key, value pair?**	
20	**Write a program to create a registration form with different Controls?**	
21	**Write a program to create a registration form with different menus?**	

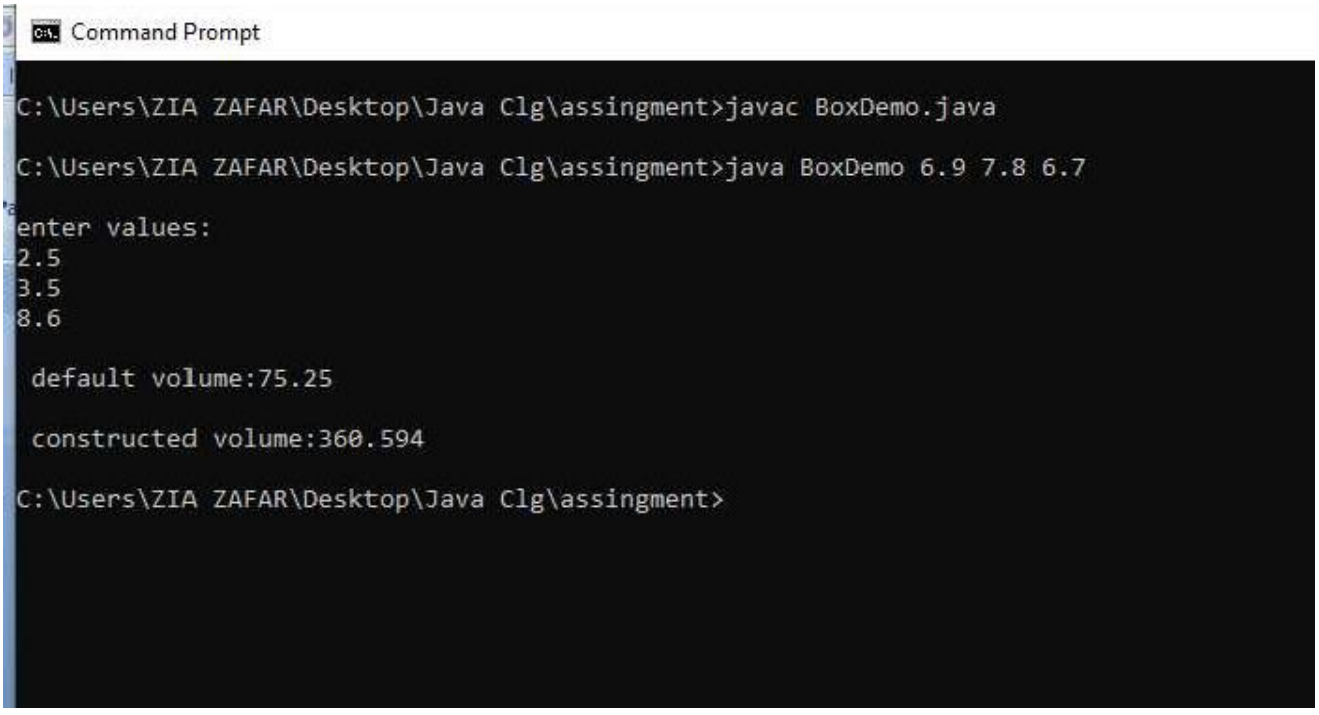
S.no	Particulars	Page no
22	**Write a program to create a registration form for demonstrating event handling?**	
23	**Write a program to copy data from one file to another file?**	
24	**write a program to read a file and display output on console?**	
25	**Write a program to illustrate Serialization?**	
26	**Write a program to retrieve web page using URL?**	
27	**Write a program to implement java network programming?(client and server program)**	
28	**Write a program to implement border Layout?**	
29	**Write a program to implement flow layout?**	
30	**Write a program to Demonstrate Key Listener?**	

1. Write a 'java' Program to Demonstrate the concept of class Box with Constructors

```
import java.io.*;
 class Box
{
double width; double
height; double depth;
Box()
{
BufferedReader br=new BufferedReader(new
InputStreamReader(System.in)); System.out.println("\nenter values:");
try{
width=Double.parseDouble(br.readLine());
height=Double.parseDouble(br.readLine());
depth=Double.parseDouble(br.readLine());
}
catch(IOException ioe)
{
}
}
Box(double w,double h,double d)
{
width=w;
height=h; depth=d;
}
double volume()
{
return (width*height*depth);
}
```

```
}

class BoxDemo

{

public static void main(String args[])

{

Box b=new Box();

Box b1=new
Box(Double.parseDouble(args[0]),Double.parseDouble(args[1]),Double.parseDouble(

args[2])); double vol; vol=b.volume();

System.out.println("\n default volume:"+vol);

vol=b1.volume();

System.out.println("\n constructed volume:"+vol);

} }
```

Output□

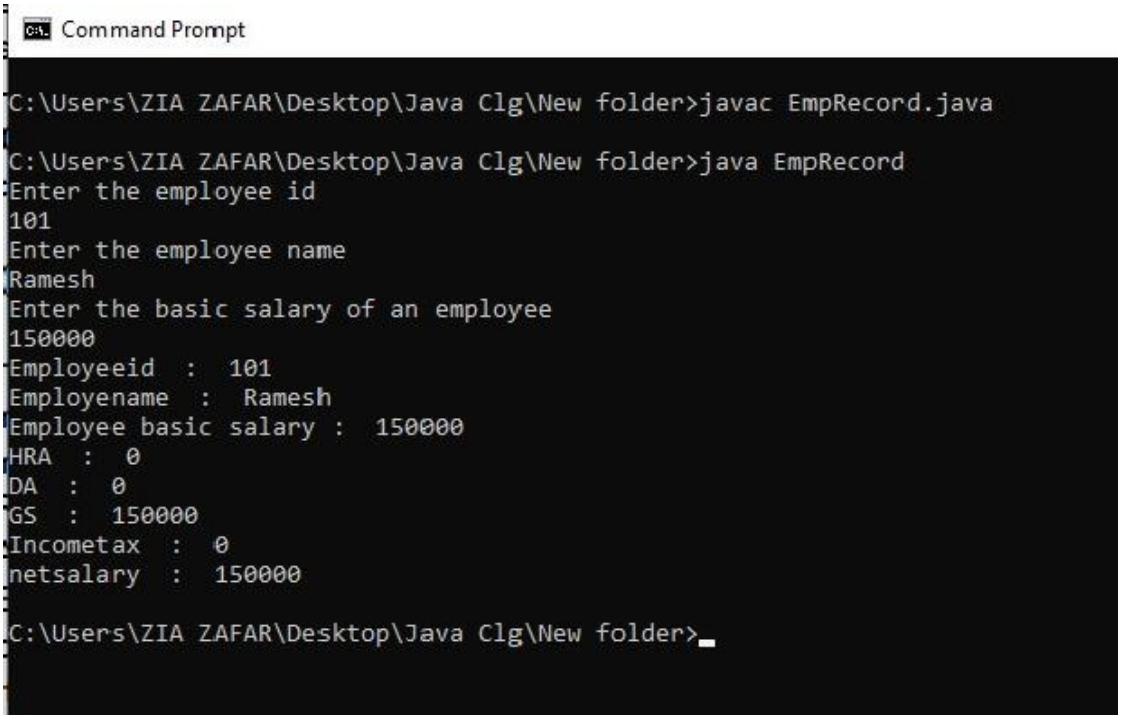


2. Write a java program to calculate salary of n employees using concept of classes with constructor and method?

```
import java.util.*;
class Employee
{
    private String employeid;
    private String empname;
    private int basicsalary,HRA,DA,GS,incometax,netsalary;
    public void Employee() //Contrcutor
    {
      Scanner sc= new Scanner(System.in);
      System.out.println("Enter the employee id"); //taking all the inputs from the user
      employeid=sc.next();
      System.out.println("Enter the employee name");
      empname=sc.next();
      System.out.println("Enter the basic salary of an employee");
      basicsalary=sc.nextInt();
      calculate();
    }
    public void calculate() //calculating all the parameters
    {
      HRA=(10/100)*basicsalary;
      DA=(73/100)*basicsalary;
      GS=basicsalary+DA+HRA;
      incometax=(30/100)*GS;
      netsalary=GS-incometax;
    }
    public void display() //displaying the calculating parameters
    {
```

```
      System.out.println("Employeeid : "+employeid);
System.out.println("Employename : "+empname);
System.out.println("Employee basic salary : "+basicsalary);
System.out.println("HRA  :   "+HRA);
System.out.println("DA :  "+DA);
System.out.println("GS : "+GS);
System.out.println("Incometax : "+incometax);
System.out.println("netsalary : "+netsalary);
    }
}
class EmpRecord
{
    public static void main(String args[])
    {
       Employee emp=new Employee();
       emp.Employee();
      emp.display();
    }
}
```

Output:

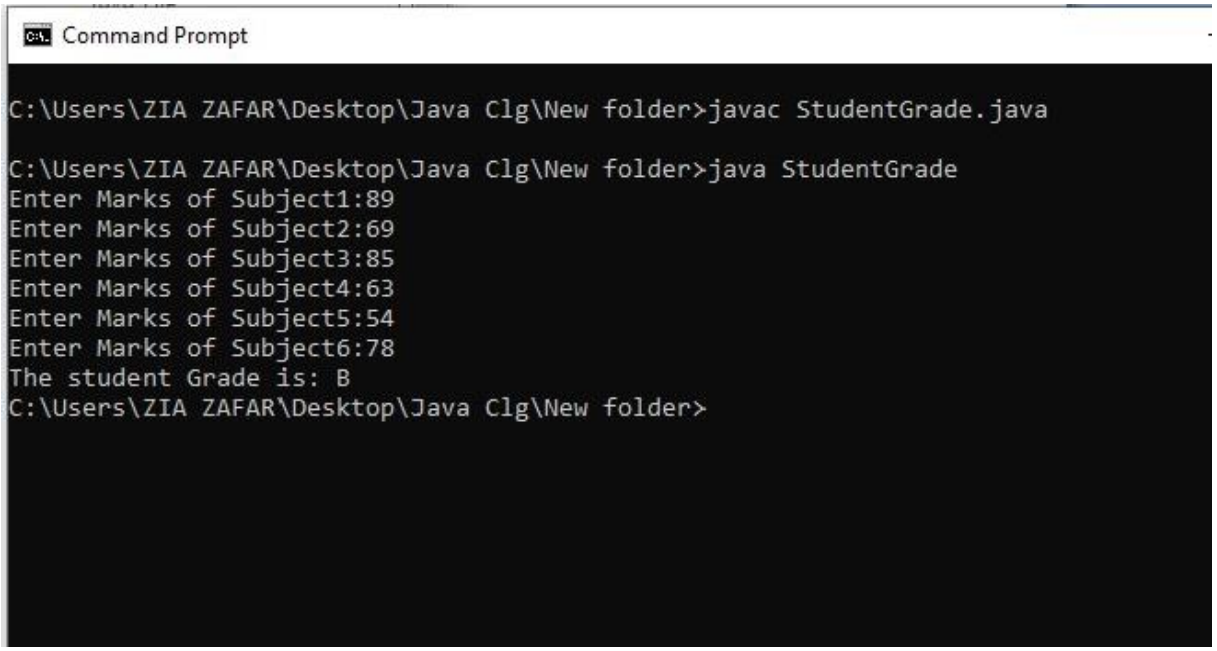


3. write a program to calculate students grade using class methods?

```
import java.util.Scanner;
class StudentGrade

{
int marks[] = new int[6];
int i;
float total=0, avg;
public void student()
{
Scanner scanner = new Scanner(System.in);
for(i=0; i<6; i++)
{
System.out.print("Enter Marks of Subject"+(i+1)+":");
marks[i] = scanner.nextInt();
total = total + marks[i];
}
```

```
scanner.close();
}
//Calculating average here
public void stdGrade()
{
avg = total/6;
System.out.print("The student Grade is: ");
if(avg>=80)
{
System.out.print("A");
}
else if(avg>=60 && avg<80)
{
System.out.print("B");
}
else if(avg>=40 && avg<60)
{
System.out.print("C");
}
else
{
System.out.print("D");
}
}
public static void main(String args[])
{
StudentGrade sg =new StudentGrade();
sg.student();
sg.stdGrade();
}}
```

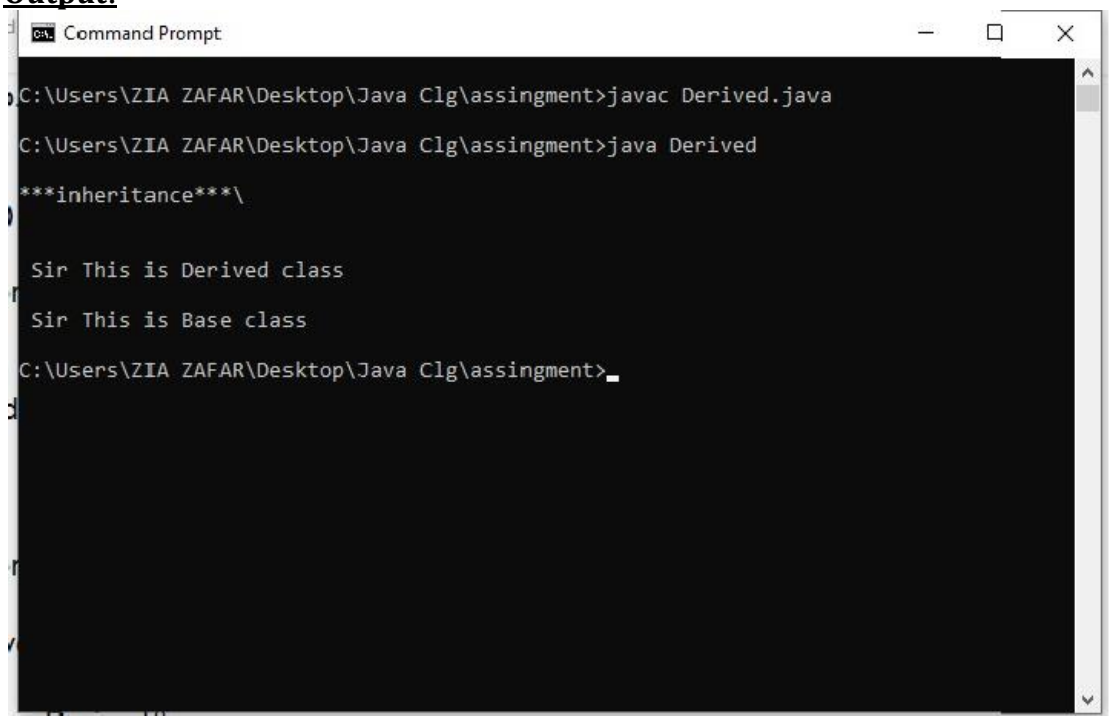
Output:


Command Prompt
C:\Users\ZIA ZAFAR\Desktop\Java Clg\New folder>javac StudentGrade.java
C:\Users\ZIA ZAFAR\Desktop\Java Clg\New folder>java StudentGrade
Enter Marks of Subject1:89
Enter Marks of Subject2:69
Enter Marks of Subject3:85
Enter Marks of Subject4:63
Enter Marks of Subject5:54
Enter Marks of Subject6:78
The student Grade is: B
C:\Users\ZIA ZAFAR\Desktop\Java Clg\New folder>

4. write a program to implement single inheritance?

```
import java.io.*;
class Base
{
void display()
{
System.out.println("\n Sir This is Base class");
}
}
class Derived extends Base
{
void show()
{
System.out.println("\n Sir This is Derived class");
}
public static void main(String args[])
{
Derived d=new Derived();
System.out.println("\n***inheritance***\\\n");
d.show();
d.display();
}
}
```

Output:

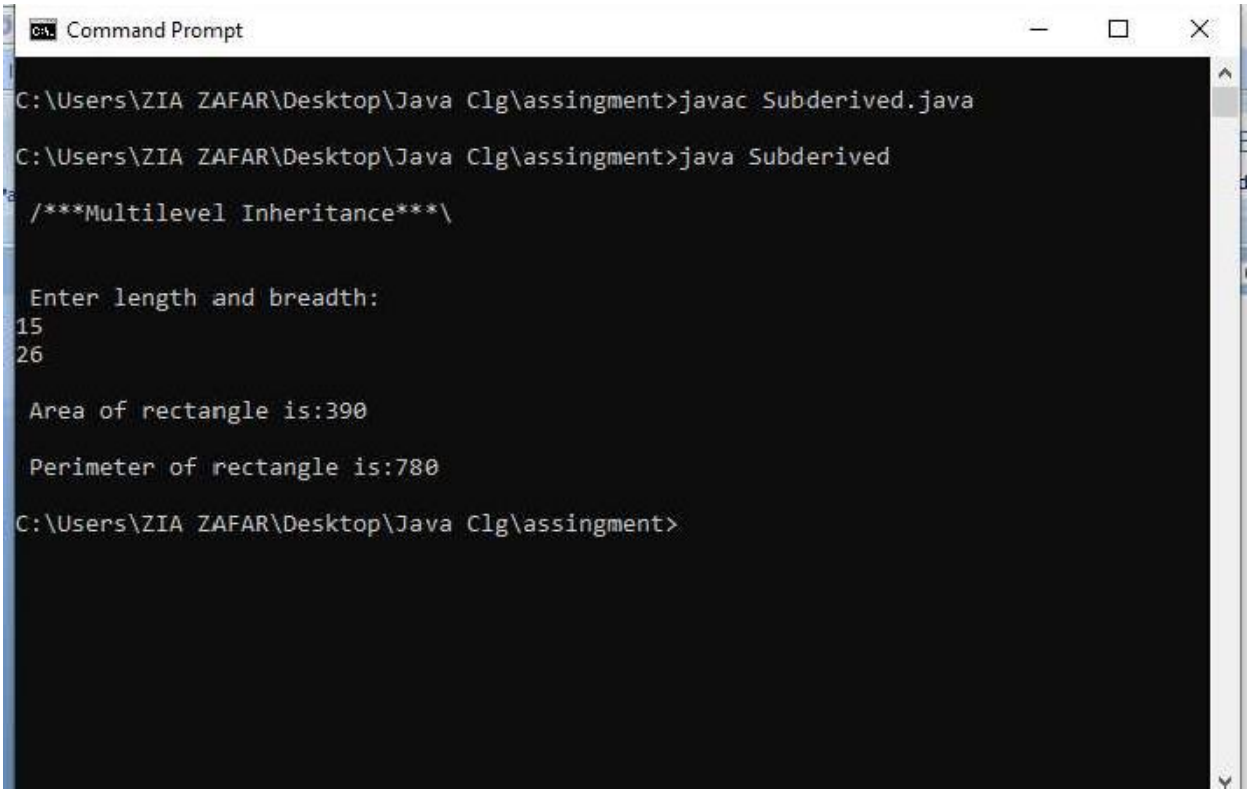

Command Prompt
C:\Users\ZIA ZAFAR\Desktop\Java Clg\assingment>javac Derived.java
C:\Users\ZIA ZAFAR\Desktop\Java Clg\assingment>java Derived
inheritance\
Sir This is Derived class
Sir This is Base class
C:\Users\ZIA ZAFAR\Desktop\Java Clg\assingment>

5. Write a program to implement multilevel inheritance?

```
import java.io.*;
import java.lang.*;
class Base
{
int l,b;
void area()
{
System.out.println("\n Area of rectangle is:"+(l*b));
}
}
class Derived extends Base
{
void perimeter()
{
area();
System.out.println("\n Perimeter of rectangle is:"+2*(l*b));
}
}
class Subderived extends Derived
{
void Rectangle() throws IOException
{
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
System.out.println("\n Enter length and breadth:");
l=Integer.parseInt(br.readLine());
b=Integer.parseInt(br.readLine());
perimeter();
}
public static void main(String args[]) throws Exception
{
```

```
System.out.println("\n /***Multilevel Inheritance***\\\n");
Subderived sd=new Subderived();
sd.Rectangle();
}
}
```

Output:

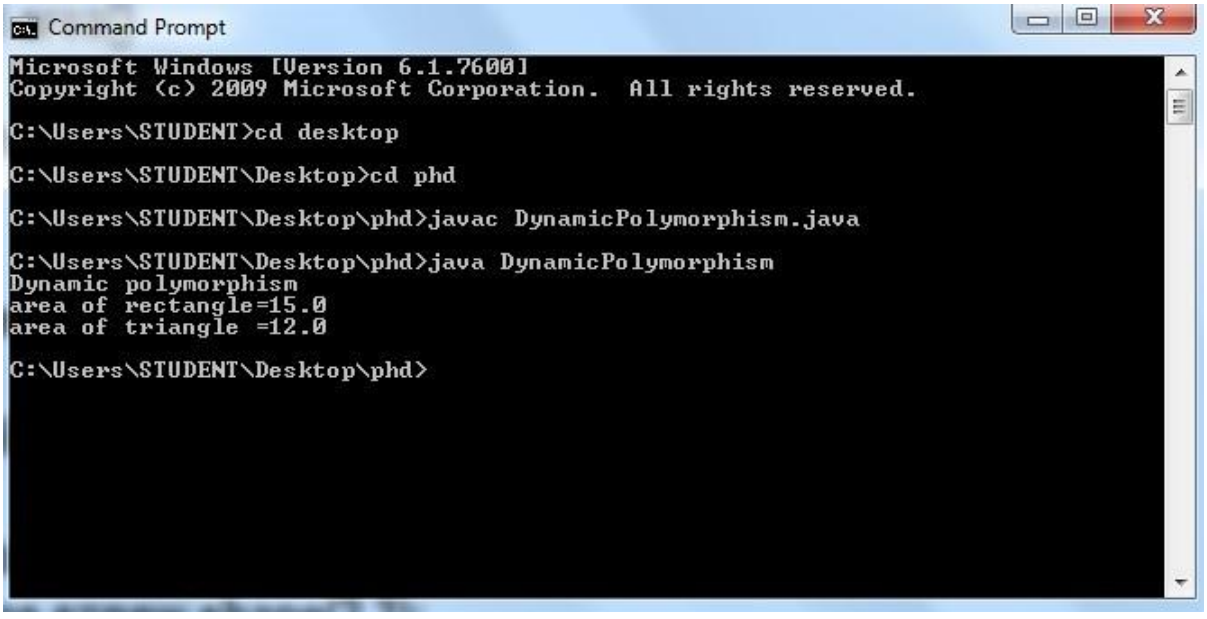


6. Write a program to demonstrat abstract class and dynamic polymorphism?

```
import java.io.*;
class shape
{
double a,b;
shape(double a,double b)
{
this.a=a;
this.b=b;
}
}
class Rectangle extends shape
{
Rectangle(double a,double b)
{
super(a,b);
}
void area()
{
System.out.println("area of rectangle="+(a*b));
}
}
class Triangle extends shape
{
Triangle(double a,double b)
{
super(a,b);
}
void area()
```

```
{
System.out.println("area of triangle ="+((a*b/2)));
}
}
class DynamicPolymorphism
{
public static void main(String args[])
{
System.out.println("Dynamic polymorphism");
shape s=new shape(2,3);
Rectangle r=new Rectangle(3,5);
Triangle t=new Triangle(4,6);
shape ref;
ref=r;
ref.area();
ref=t;
ref.area();
}
}
```

Output



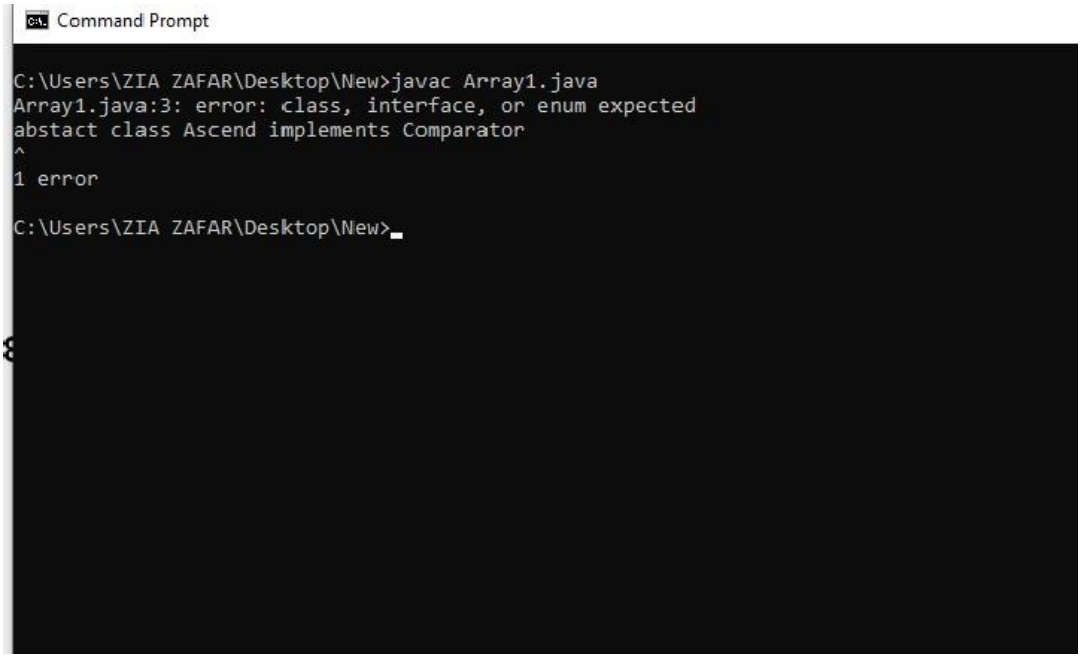
7. Write a program to implement packages?

```
package pack;
import java.util.*;
public class DemoPackage
{
public void msg()
{
System.out.println("Now You Are in package Enter Two number magic...");
Scanner sc=new Scanner(System.in);
int n1=sc.nextInt();
int n2=sc.nextInt();
int sum=n1+n2;
System.out.println("Sum Of Two Number is ="+sum );
}
}
```

// Access Package

```
import pack.DemoPackage;
class AccessPackage
{
  public static void main(String args[])
{
   DemoPackage obj = new DemoPackage();
   obj.msg();
  }
}
```

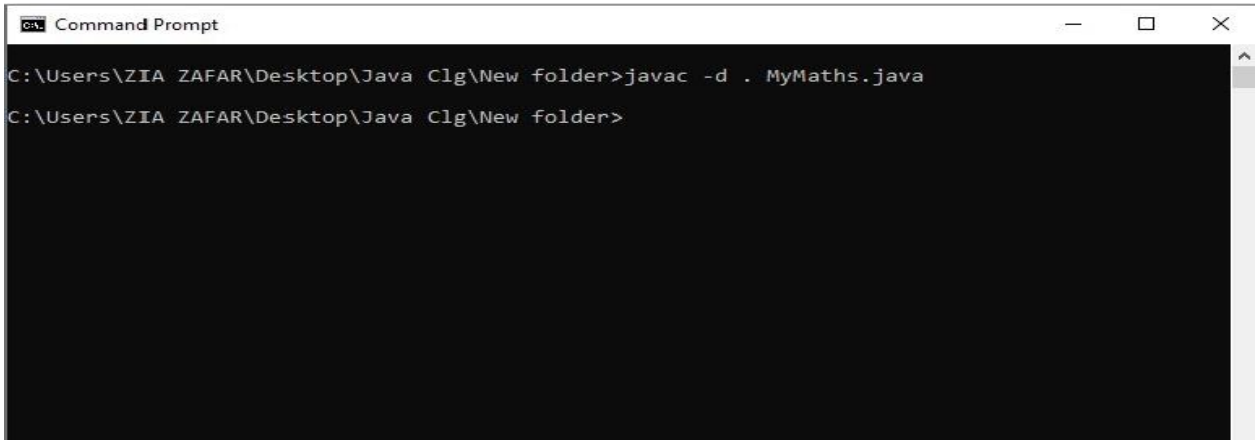
Output:


Command Prompt
C:\Users\ZIA ZAFAR\Desktop\New>javac Array1.java
Array1.java:3: error: class, interface, or enum expected
abstact class Ascend implements Comparator
^
1 error
C:\Users\ZIA ZAFAR\Desktop\New>

8. Write a program to demonstrate various arithmetic calculations using packages?

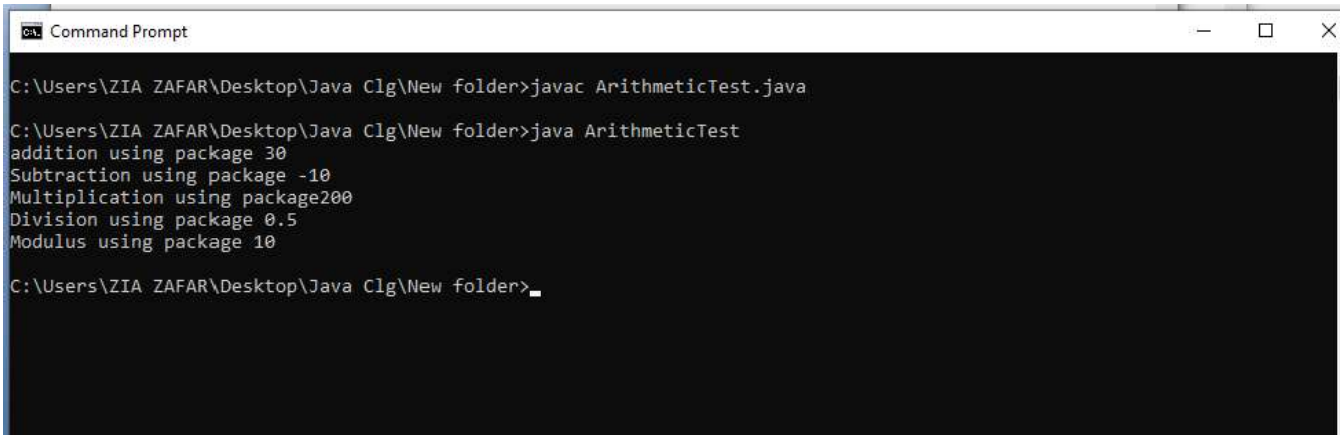
```
package demopack;
public class MyMaths
{
public int add(int x,int y)
{
return x+y;
}
public int sub(int x,int y)
{
return x-y;
}
public int mul(int x,int y)
{
return x*y;
}
public double div(int x,int y)
{
return (double)x/y;
}
public int mod(int x,int y)
{
return x%y;
}
}
```

Package Creations




```
import demopack.MyMaths;
import java.util.*;
public class ArithmeticTest
{
public static void main(String args[])
{
 MyMaths m = new MyMaths();
System.out.println("addition using package "+ m.add(10,20));
System.out.println("Subtraction using package "+ m.sub(10,20));
System.out.println("Multiplication using package"+ m.mul(10,20));
System.out.println("Division using package "+ m.div(10,20));
System.out.println("Modulus using package "+ m.mod(10,20));
}
}
```

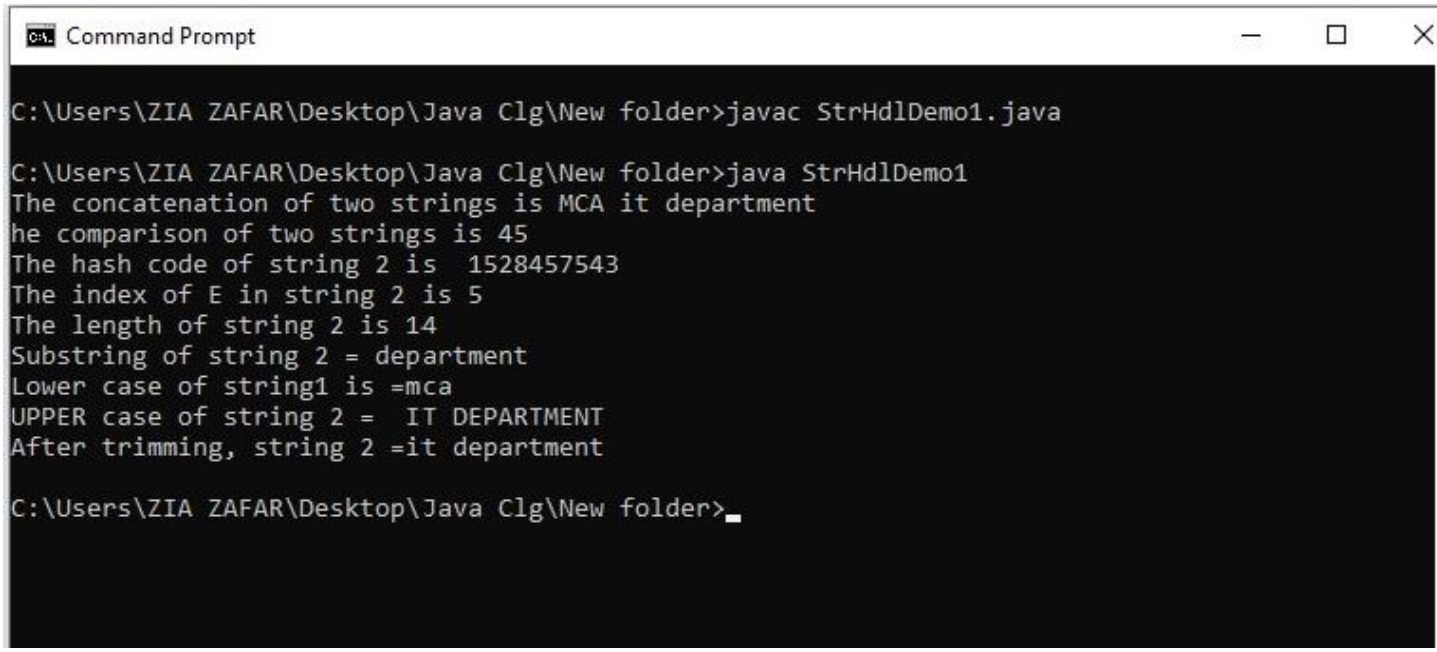
Output:


Command Prompt
C:\Users\ZIA ZAFAR\Desktop\Java Clg\New folder>javac ArithmeticTest.java
C:\Users\ZIA ZAFAR\Desktop\Java Clg\New folder>java ArithmeticTest
addition using package 30
Subtraction using package -10
Multiplication using package200
Division using package 0.5
Modulus using package 10
C:\Users\ZIA ZAFAR\Desktop\Java Clg\New folder>

9. Write a program to implement string handling methods?

```
class StrHdlDemo1
{
public static void main(String args[])
{
String s1="MCA";
String s2=" it department";
String st=s1.concat(s2);
int cmp=s1.compareTo(s2);
long hs=s2.hashCode();
int s=s2.indexOf('e',3);
int ln=s2.length();
String sb=s2.substring(4);
String lwr=s1.toLowerCase();

String upr=s2.toUpperCase();
String trm=s2.trim();
System.out.println("The concatenation of two strings is "+st);
System.out.println("he comparison of two strings is "+cmp);
System.out.println("The hash code of string 2 is "+ hs);
System.out.println("The index of E in string 2 is "+s);
System.out.println("The length of string 2 is "+ln);
System.out.println("Substring of string 2 = "+sb);
System.out.println("Lower case of string1 is ="+lwr);
System.out.println("UPPER case of string 2 = "+upr);
System.out.println("After trimming, string 2 ="+trm);
}
}
```

Output□

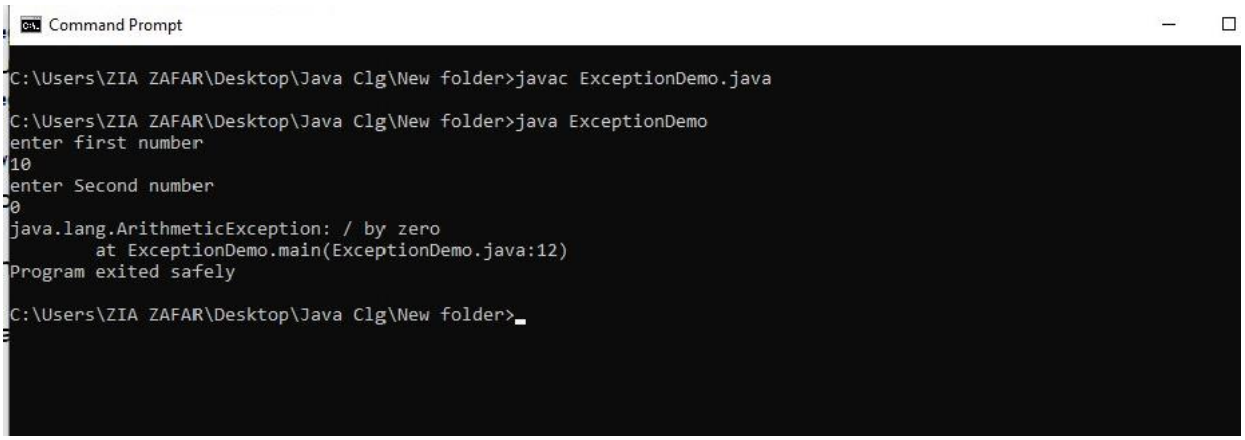


10. Write a program to implement Exceptional handling?

import java.io.*;

```
class ExceptionDemo
{
 public static void main(String args[]) throws IOException
 {
  BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
   System.out.println("enter first number");
  int a = Integer.parseInt(br.readLine());

  System.out.println("enter Second number");
  int b = Integer.parseInt(br.readLine());
  try
  { int c =a/b;
    System.out.println("the output = "+c);
    }
catch(ArithmeticException ae)
{
  ae.printStackTrace();
}
finally
{
 System.out.println("Program exited safely ");
 }
}
}
```

Output:

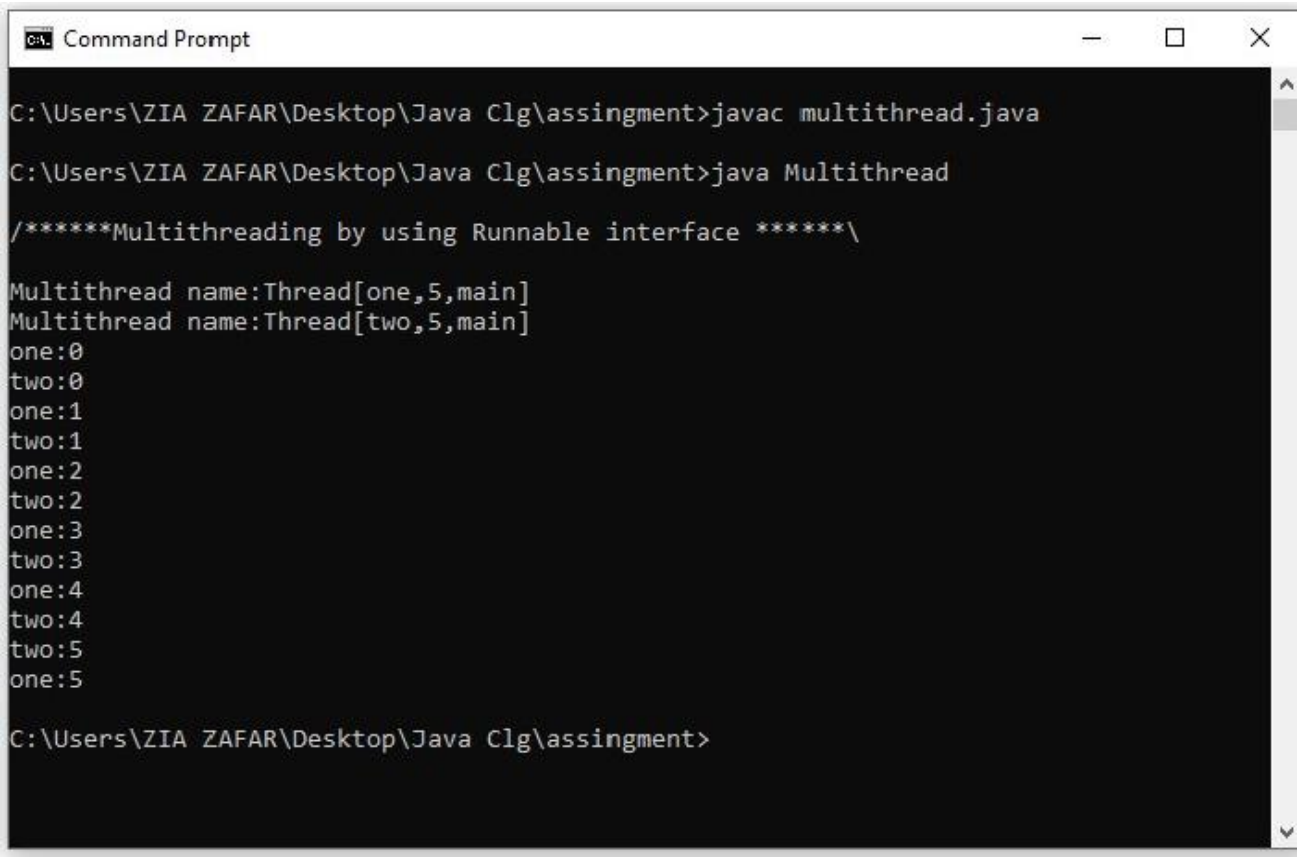


11. Write a program to implement Multithreading?

```
import java .util.*;
class Multithread implements Runnable
{
String name;Thread t;
Multithread(String s1)
{
name=s1;t=new Thread(this,name);
System.out.println("Multithread name:"+t);

t.start();
}
public void run()
{
try
{
for(int i=0;i<=5;i++)
{
System.out.println(name+":"+i);
Thread.sleep(500);
}
}
catch(InterruptedException e)
{
System.out.println("Interrupted");
}
}
public static void main(String[]args)
{
```

```
System.out.println("\n/******Multithreading by using Runnable interface ******\\\n");
new Multithread("one");
new Multithread("two");
}
}
```

Output

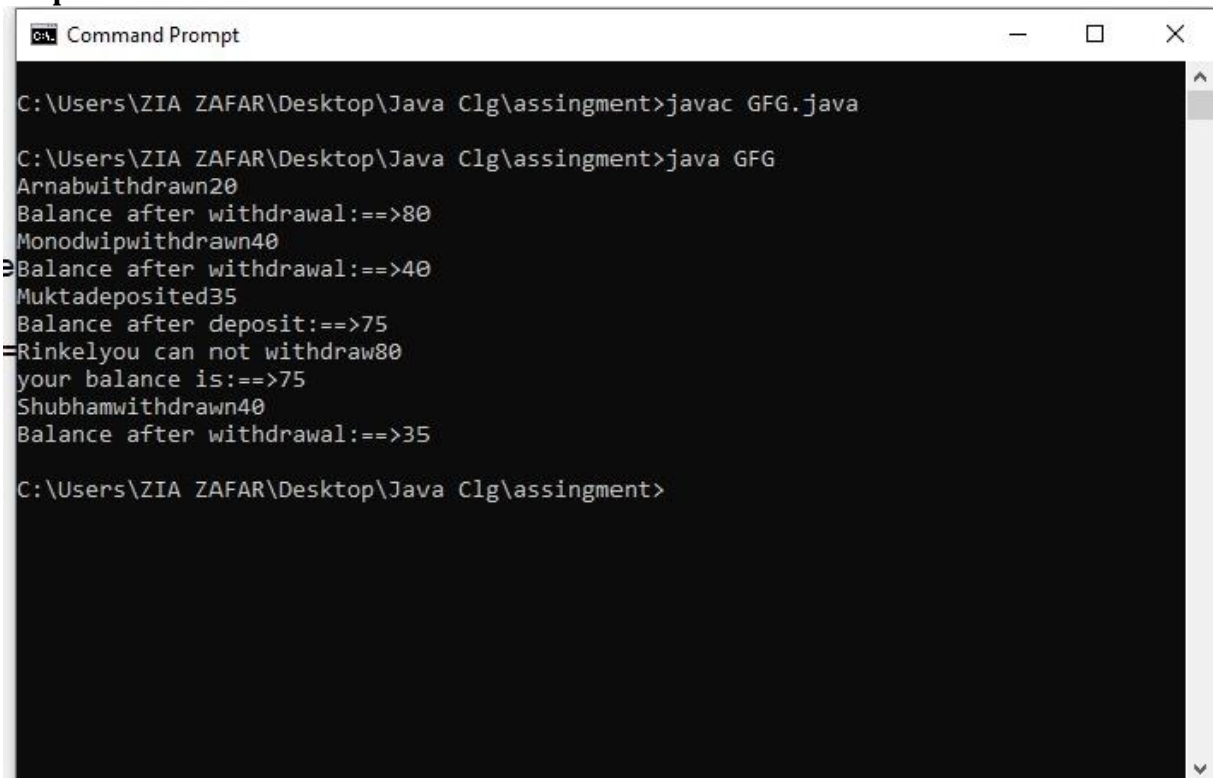


12. Write a program to demonstrate mutual exclusion using thread synchronization?

```
import java.io.*;
class Bank
{
int total=100;
void withdrawn(String name,int withdrawal)
{
if(total>=withdrawal){
System.out.println(name+ "withdrawn"+ withdrawal);
total=total-withdrawal;
System.out.println("Balance after withdrawal:==>" +total);
try{

Thread.sleep(1000);
}
catch(InterruptedException e)
{
e.printStackTrace();
}
}
else
{
System.out.println(name+ "you can not withdraw" + withdrawal);
System.out.println("your balance is:==>"  +total);
try
{
Thread.sleep(1000);
}
catch(InterruptedException e)
{e.printStackTrace();
```

```
}}}
void deposit(String name,int deposit)
{
System.out.println(name+ "deposited" +deposit);
total=total+deposit;
System.out.println("Balance after deposit:==>"+total);
try
{
Thread.sleep(1000);
}
catch(InterruptedException e)
{
e.printStackTrace();
}}}
class GFG
{
public static void main(String[] args)throws IOException
{
Bank obj=new Bank();
obj.withdrawn("Arnab",20);
obj.withdrawn("Monodwip",40);
obj.deposit("Mukta",35);
obj.withdrawn("Rinkel",80);
obj.withdrawn("Shubham",40);
}}
```

Output

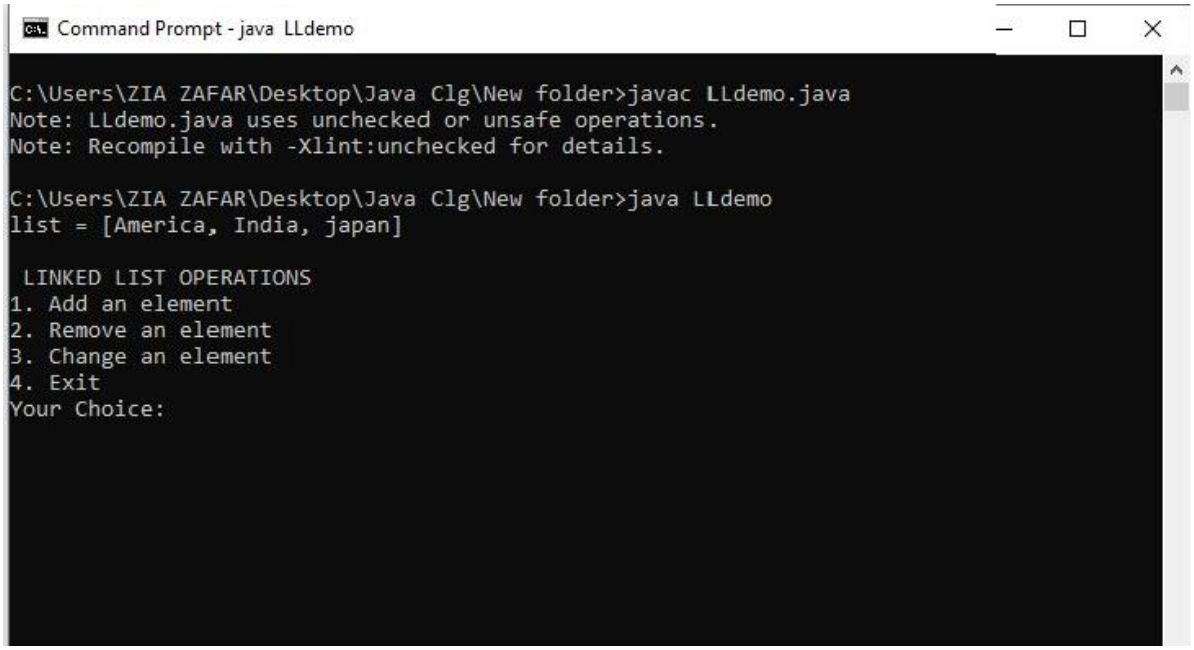

Command Prompt
C:\Users\ZIA ZAFAR\Desktop\Java Clg\assingment>javac GFG.java
C:\Users\ZIA ZAFAR\Desktop\Java Clg\assingment>java GFG
Arnabwithdrawn20
Balance after withdrawal:==>80
Monodwipwithdrawn40
Balance after withdrawal:==>40
Muktadeposited35
Balance after deposit:==>75
Rinkelyou can not withdraw80
your balance is:==>75
Shubhamwithdrawn40
Balance after withdrawal:==>35
C:\Users\ZIA ZAFAR\Desktop\Java Clg\assingment>

13. Write a program to demonstrate Linked list class?

```
import java.io.*;
import java.util.*;
class LLdemo
{
 public static void main(String args[]) throws IOException
 {
  LinkedList ll = new LinkedList();
        ll.add("America");
        ll.add("India");
        ll.add("japan");
        System.out.println("list = "+ll);
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

        String element;
        int position,choice=0;
        while(choice<4)
        {
                System.out.println("\n LINKED LIST OPERATIONS");
                System.out.println("1. Add an element");
                System.out.println("2. Remove an element");
                System.out.println("3. Change an element");
                System.out.println("4. Exit");
        System.out.println("Your Choice: ");
        choice = Integer.parseInt(br.readLine());
        switch(choice)
        {
                case 1: System.out.println("Enter element : ");
                        element = br.readLine();
                        System.out.println("At what position");
```

```
                position = Integer.parseInt(br.readLine());
                ll.add(position-1,element);
                break;
            case 2: System.out.println("enter position");
                position = Integer.parseInt(br.readLine());
                ll.remove(position-1);
                break;
            case 3: System.out.println("enter position");
                position = Integer.parseInt(br.readLine());
                System.out.println("enter new element");
                element=br.readLine();

                ll.set(position-1,element);
                break;
            default: return;
            }
            System.out.println("list= "+ll);
            Iterator it = ll.iterator();
            while(it.hasNext());
            System.out.println("it.next()"+" ");
        }
}
}
```

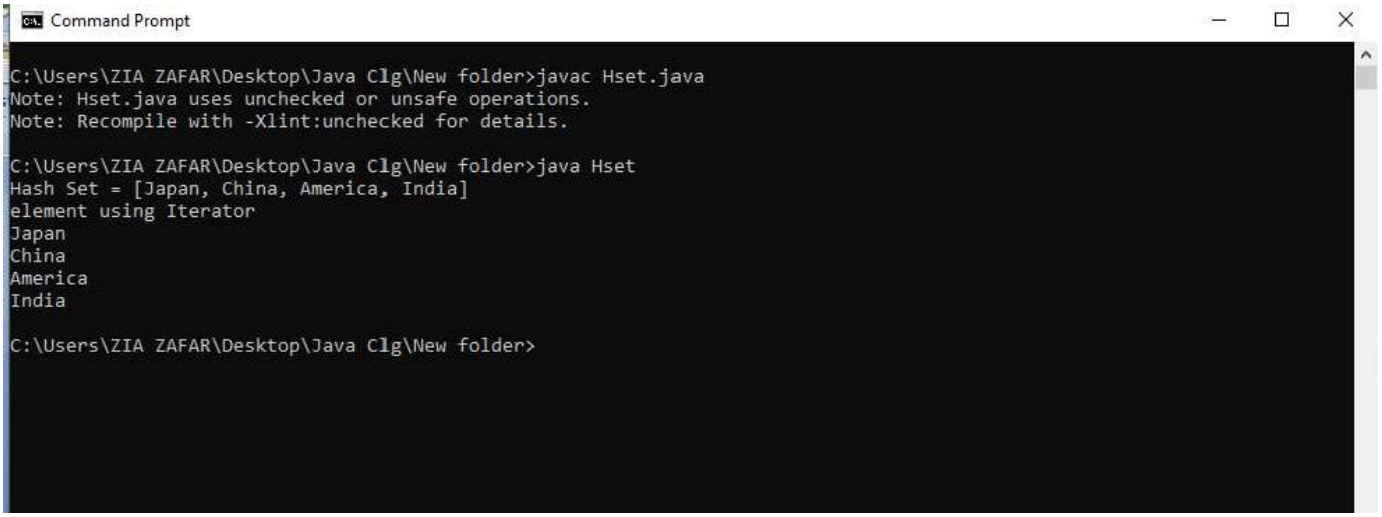
Output:


Command Prompt - java LLdemo
C:\Users\ZIA ZAFAR\Desktop\Java Clg\New folder>javac LLdemo.java
Note: LLdemo.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
C:\Users\ZIA ZAFAR\Desktop\Java Clg\New folder>java LLdemo
list = [America, India, japan]
LINKED LIST OPERATIONS
1. Add an element
2. Remove an element
3. Change an element
4. Exit
Your Choice:

14. Write a program to demonstrate Hash Set class?

```
import java.util.*;
class Hset
{
 public static void main(String args[])

 {
  HashSet<String> hs = new HashSet();
  hs.add("India");
  hs.add("America");
  hs.add("Japan");
  hs.add("China");
  hs.add("America");
 System.out.println("Hash Set = "+hs);
 Iterator<String> it = hs.iterator();
 System.out.println("element using Iterator");
 while(it.hasNext())
  {
     //String s = (String)it.next;
    System.out.println(it.next());
   }
 }
}
```

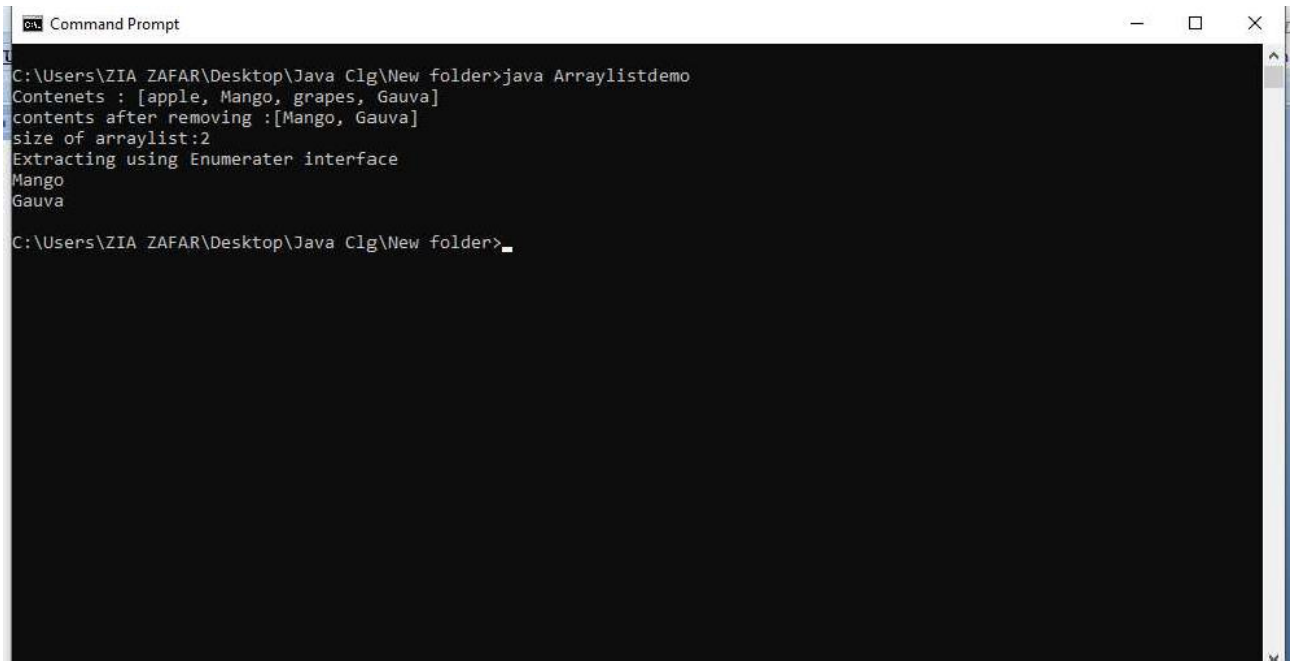
Output:



15. Write a program to demonstrate Iterator class?

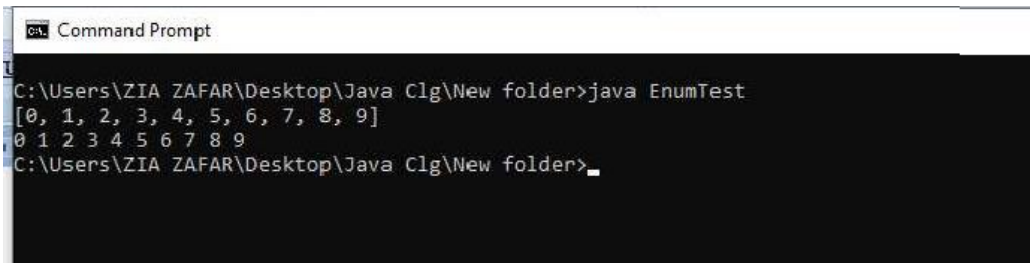
```
import java.util.*;
class Arraylistdemo
{
 public static void main(String args[])
 {
  ArrayList arl = new ArrayList();
     arl.add("apple");
     arl.add("Mango");
     arl.add("grapes");
     arl.add("Gauva");
     System.out.println("Contenets : "+arl);

     arl.remove("apple");
     arl.remove(1);
     System.out.println("contents after removing :"+arl);
     System.out.println("size of arraylist:"+arl.size());
     System.out.println("Extracting using Enumerater interface");
     //Enumeration e = arl.enumeration();
     //while(e.hasMoreElements())
     Iterator tr=arl.iterator();
     while(tr.hasNext())
     {
     System.out.println(tr.next());
     }

  }
}
```

Output:



16. Write a program to demonstrate Enumeration interface?

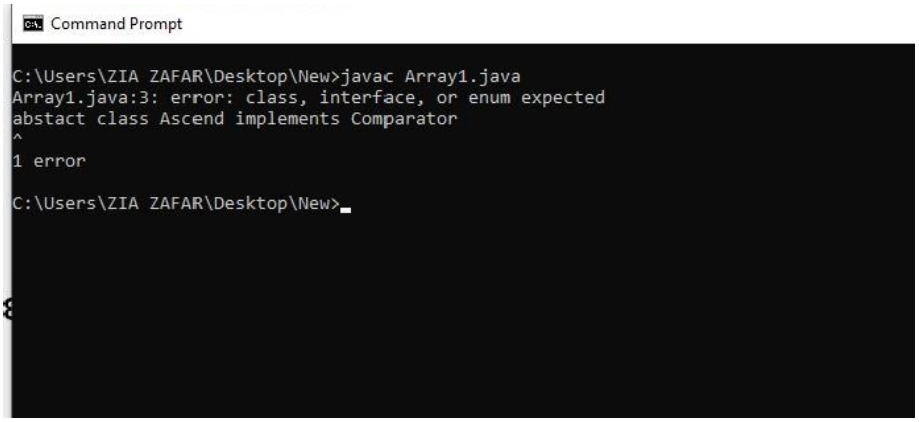
```
import java.util.Enumeration;
import java.util.Vector;
public class EnumTest
{
    public static void main(String[] args)
  {
        Vector v = new Vector();
       for (int i = 0; i < 10; i++)
       v.addElement(i);
          System.out.println(v);

 Enumeration e = v.elements();
  while (e.hasMoreElements())
    {
      int i = (Integer)e.nextElement();
       System.out.print(i + " ");
    }
  }
}
```



17. Write a program to demonstrate Comparator Interface?

```
import java.io.*;
import java.util.*;
abstact class Ascend implements Comparator
{
	public int compare(Integer i1,Integer i2)
	{	return i1.compareTo(i2); }
}
class Decend implements Comparator
{
	public int compare(Integer i1,Integer i2)
	{	return i2.compareTo(i1); }
}
class Array1
{
	public static void main(String args[]) throws IOException
	{
	 BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
	 System.out.print("how many elements");
	 int size = Integer.parseInt(br.readLine());
	 Integer arr[] = new Integer[size];
	 for(int i=0;i<size;i++)
	 {
		System.out.println("Enter int:");
		arr[i]=Integer.parseInt(br.readLine());

	}
Arrays.sort(arr,new Ascend());
System.out.println("\n sorted in Asending order:");
display(arr);
 }
static void display(Integer arr[])
{
 for(Integer i: arr)
	System.out.print(i+"\t");
 }
}
```

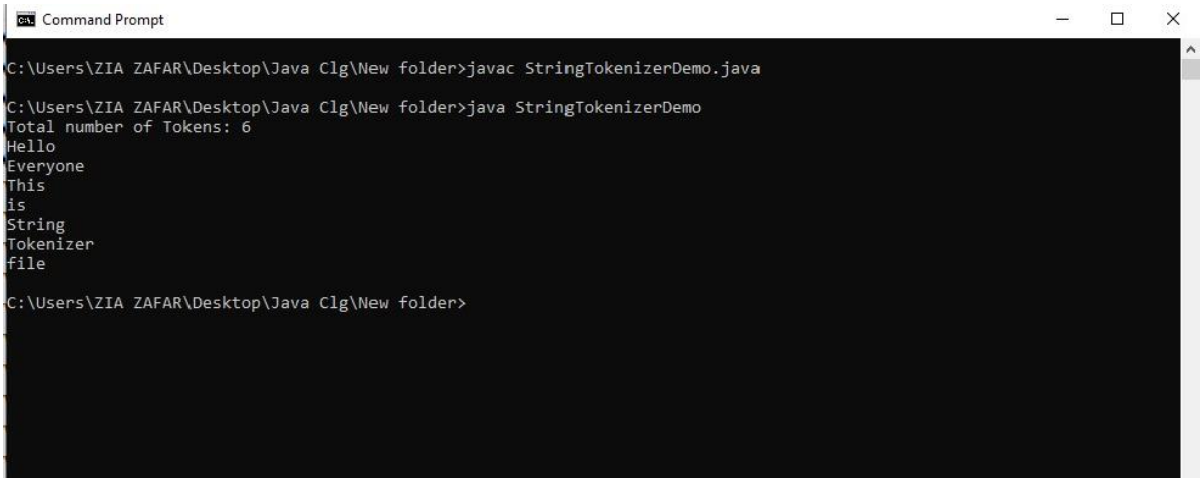
Ouput


Command Prompt
C:\Users\ZIA ZAFAR\Desktop\New>javac Array1.java
Array1.java:3: error: class, interface, or enum expected
abstact class Ascend implements Comparator
^
1 error
C:\Users\ZIA ZAFAR\Desktop\New>

18. Write a program to implement sring tokenizer?

```
import java.util.StringTokenizer;
class StringTokenizerDemo
{
public static void main(String args[])
{
StringTokenizer st = new StringTokenizer("Hello Everyone Have a nice day"," ");
System.out.println("Total number of Tokens: "+st.countTokens());
StringTokenizer sd = new StringTokenizer("Hello Everyone This is String Tokenizer file"," ");
while (sd.hasMoreTokens())
{
System.out.println(sd.nextElement());
}
}
}
```

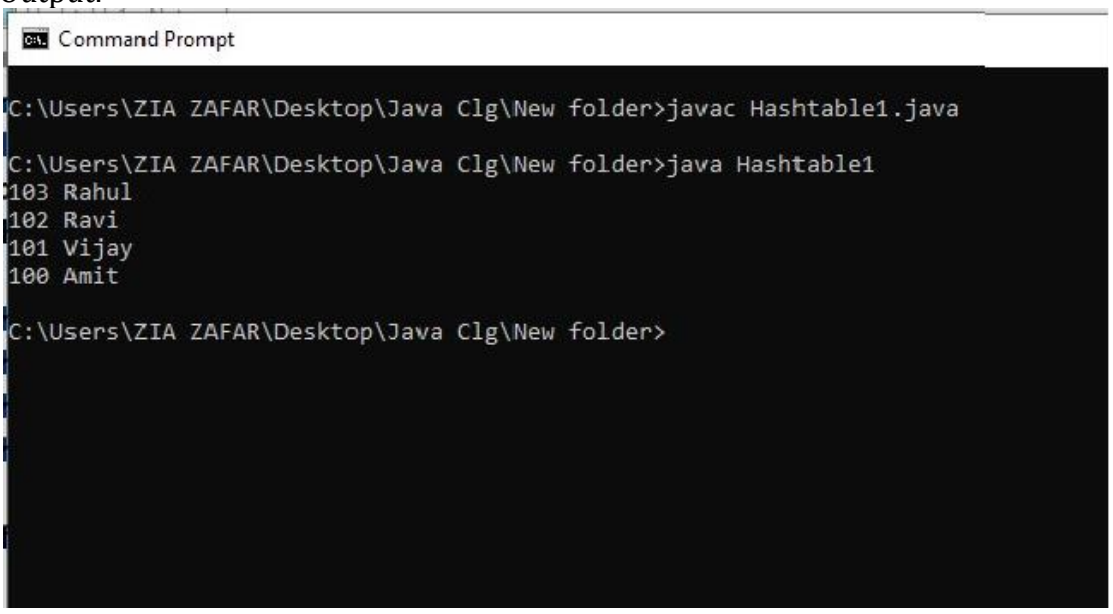
OutPut:



19. Write a program to accept data and display output in key, value pair?

```
import java.util.*;
class Hashtable1{
 public static void main(String args[]){
  Hashtable<Integer,String> hm=new Hashtable<Integer,String>();

  hm.put(100,"Amit");
  hm.put(102,"Ravi");
  hm.put(101,"Vijay");
  hm.put(103,"Rahul");

  for(Map.Entry m:hm.entrySet()){
   System.out.println(m.getKey()+" "+m.getValue());
  }
 }
}
```

Output:

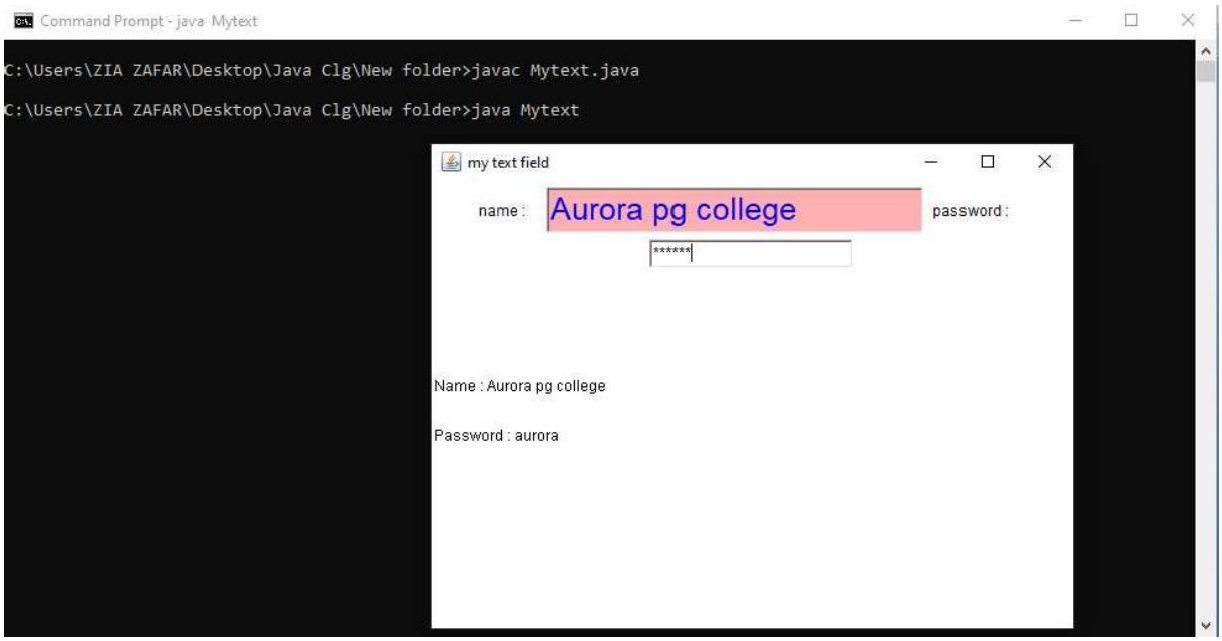


20. Write a program to create a registration form with different Controls?

```
import java.awt.*;
import java.awt.event.*;
class Mytext extends Frame implements ActionListener
{
 TextField name,pass;
 Mytext()
{
 setLayout(new FlowLayout());
 Label n = new Label("name :",Label.LEFT);
 name = new TextField(20);
 Label p = new Label("password : ",Label.LEFT);
 pass = new TextField(20);

pass.setEchoChar('*');

name.setBackground(Color.pink);
name.setForeground(Color.blue);
Font f = new Font("Arial",Font.PLAIN,25);
name.setFont(f);

this.add(n);
this.add(name);
this.add(p);
this.add(pass);
name.addActionListener(this);
pass.addActionListener(this);

this.addWindowListener(new WindowAdapter()
{
 public void windowClosing(WindowEvent we)
 { System.exit(0);}
 });
}//constructor
public void actionPerformed(ActionEvent ae)
{
  repaint();}
public void paint(Graphics g)
{
  g.drawString("Name : "+name.getText(),10,200);
  g.drawString("Password : "+pass.getText(),10,240);

}
public static void main(String args[])
{
 Mytext mt = new Mytext();
 mt.setTitle("my text field");
```

```
 mt.setSize(400,400);
 mt.setVisible(true);
}
}
```

Output:



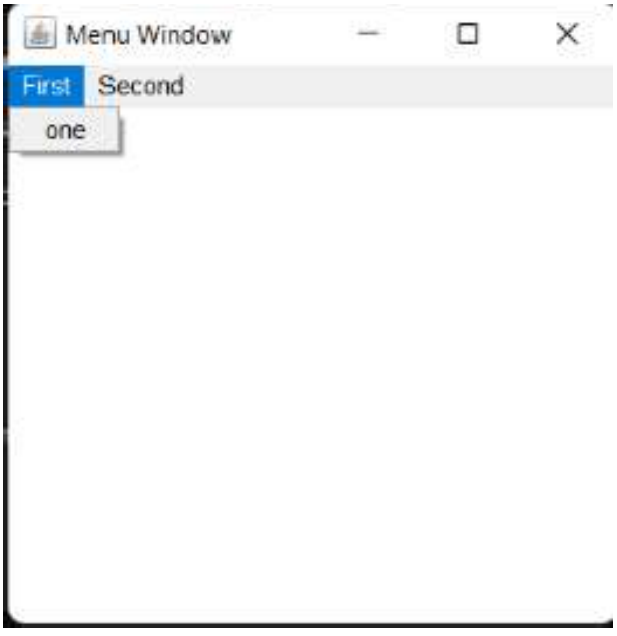
21. Write a program to create a registration form with different menus?

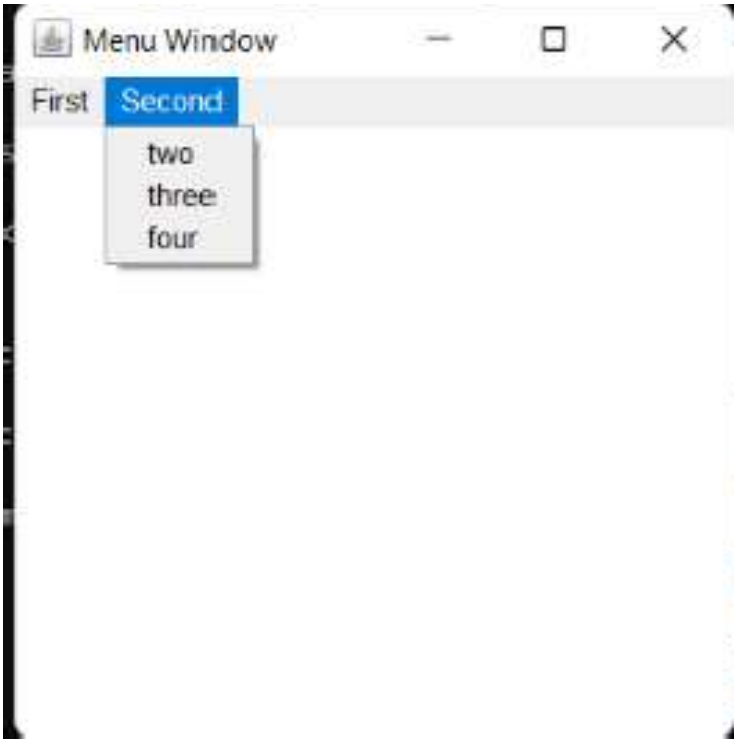
```
import java.awt.*;
import java.awt.event.*;
class Menutest extends Frame implements ActionListener
{
MenuBar mb=new MenuBar();

Menu mnu1=new Menu("First");
Menu mnu2=new Menu("Second");

MenuItem mi1=new MenuItem("one");
MenuItem mi2=new MenuItem("two");
MenuItem mi3=new MenuItem("three");
MenuItem mi4=new MenuItem("four");

public Menutest()

{
setTitle("Menu Window");
setSize(300,300);
setLocation(100,100);
mnu1.add(mi1);
mnu2.add(mi2);
mnu2.add(mi3);
mnu2.add(mi4);
mb.add(mnu1);
mb.add(mnu2);
mi1.addActionListener(this);
mi2.addActionListener(this);
mi3.addActionListener(this);
mi4.addActionListener(this);
setMenuBar(mb);
setVisible(true);
}
public void actionPerformed(ActionEvent e)
{
if (e.getSource()==mi1)
```

```
System.out.println("one selected");
if (e.getSource()==mi2)
System.out.println("two selected");
if (e.getSource()==mi3)
System.out.println("three selected");
if (e.getSource()==mi4)
System.out.println("four selected");
}
```

Output:

```
                allow classes to depend on preview features of this release
To specify an argument for a long option, you can use --<name>=<value> or
--<name> <value>.


D:\MAHESH>javac menutest.java

D:\MAHESH>java menutest.java
```



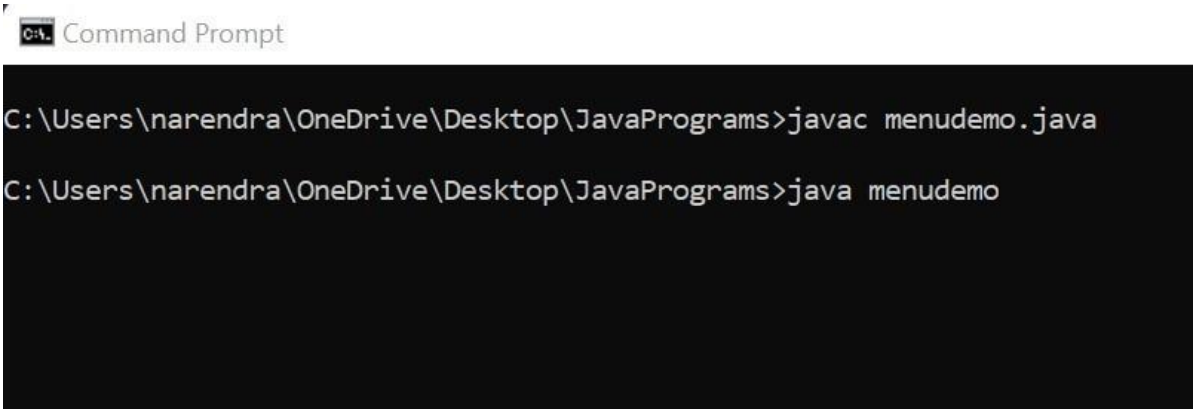

Menu Window
First
Second
two
three
four

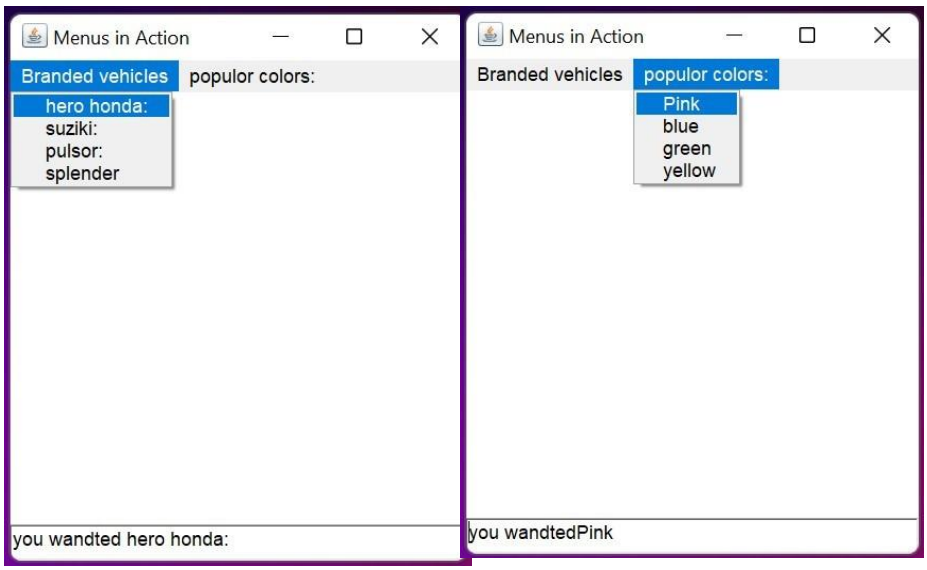
22. Write a program to create a registration form for demonstrating event handling?

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
class menudemo extends Frame implements ActionListener
{
Menu vehicles,colors;
TextField tf;
public menudemo()
{
MenuBar mb = new MenuBar();
setMenuBar(mb);
vehicles = new Menu("Branded vehicles");
colors = new Menu("populor colors:");

vehicles.add(new MenuItem(" hero honda:"));
vehicles.add(new MenuItem(" suziki:"));
vehicles.add(new MenuItem(" pulsor:"));
vehicles.add(new MenuItem(" splender"));
colors.add(new MenuItem("Pink"));
colors.add(new MenuItem("blue"));
colors.add(new MenuItem("green"));
colors.add(new MenuItem("yellow"));
mb.add(vehicles);
mb.add(colors);
vehicles.addActionListener(this);
colors.addActionListener(this);
tf = new TextField(15);
add(tf,"South");
setTitle("Menus in Action");
```

```
setSize(300,350);
setVisible(true);
}
public void actionPerformed(ActionEvent e)
{
String str = e.getActionCommand();
tf.setText("you wandted" + str);
}
public static void main(String a[])
{
new menudemo();
}
}
```

Output☐

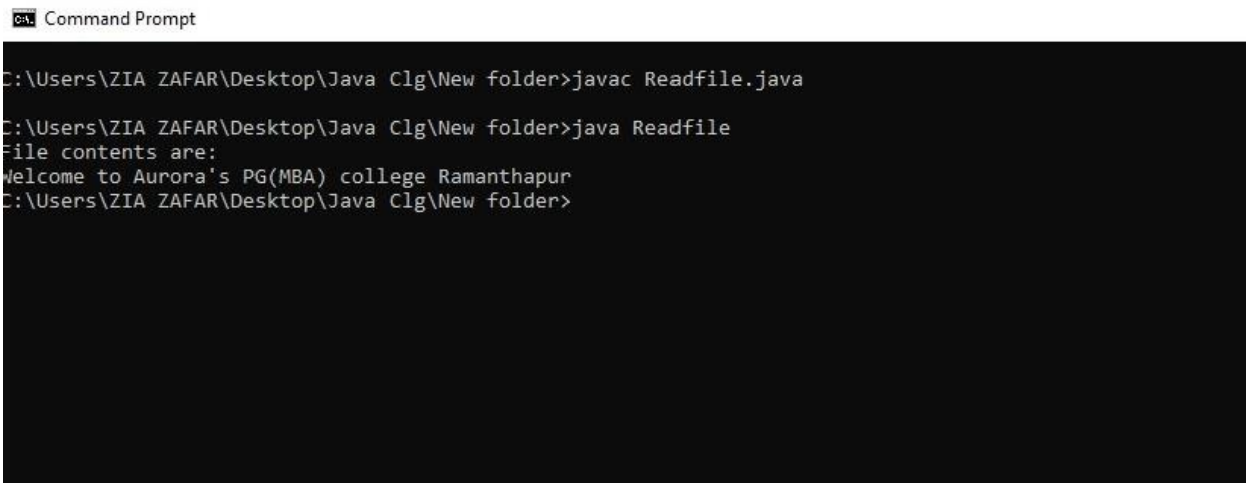




23. Write a program to copy data from one file to another file?

```
import java.io.*;
class Copyfile
{
 public static void main(String args[]) throws IOException
 {
 int ch;
  FileInputStream fin = new FileInputStream(args[0]);
  FileOutputStream fos = new FileOutputStream(args[1]);
  while((ch=fin.read())!=-1)

   fos.write(ch);

  fin.close();
  fos.close();
  System.out.println("1 file is copied");
 }
}
```

Output:

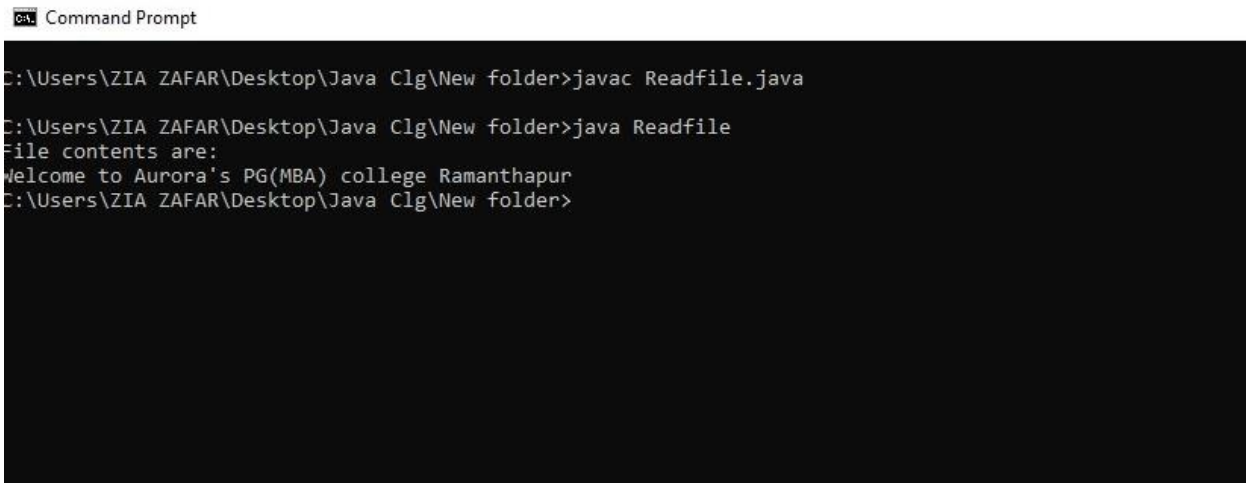


24. Write a program to read a file and display output on console?

//reading data from a file using FileInputStream

```
import java.io.*;
class Readfile
{
 public static void main(String args[]) throws IOException
{
 FileInputStream fin = new FileInputStream("myfile1.txt");
 System.out.println("File contents are:");
 int ch;
 while((ch=fin.read())!=-1)
 System.out.print((char)ch);
 fin.close();
}
}
```

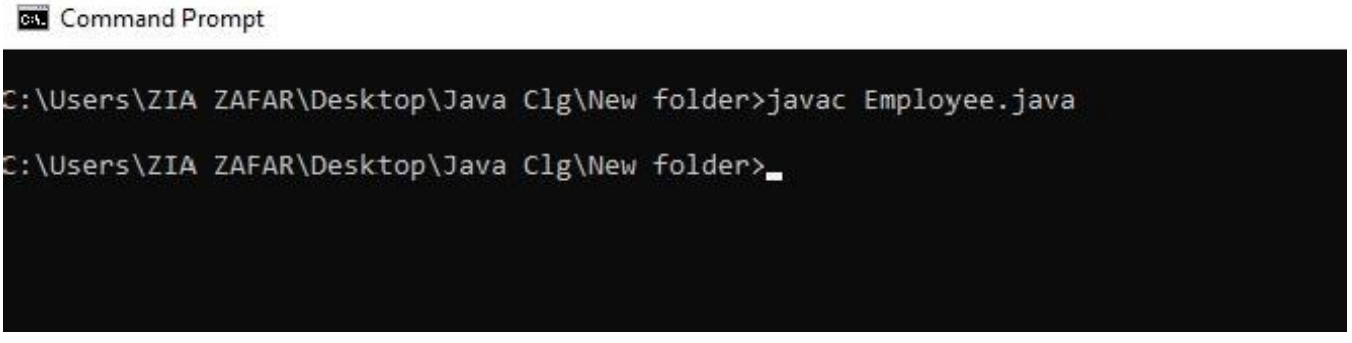
Output:



25. Write a program to illustrate Serialization?

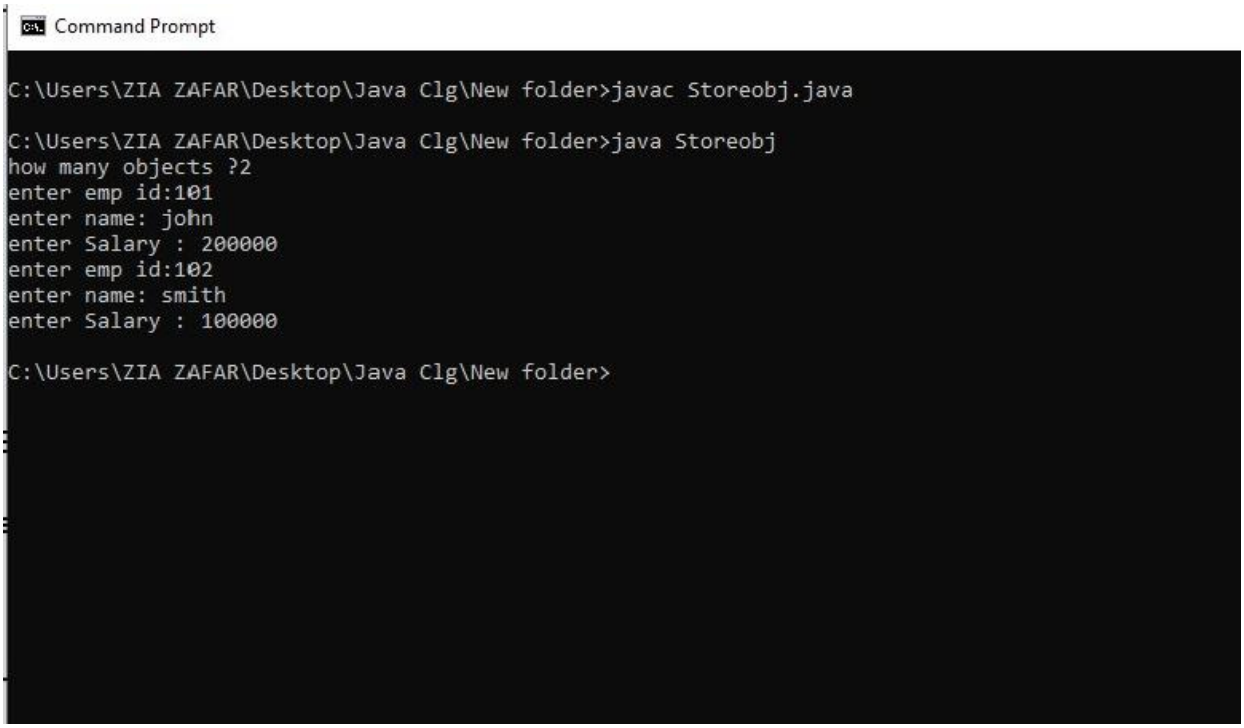
```
import java.io.*;
import java.util.Date;
class Employee implements Serializable
{
    private int id; private
    String name; private
    float sal; private Date
    doj;
Employee(int i,String n,float s,Date d)
    {
            id=i;
            name=n;
            sal=s;
            doj=d;
    }
    void display()
    {
     System.out.println(id+"\t"+name+"\t"+sal+"\t"+doj);
    }
    static Employee getdata() throws IOException
    {
            BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
            System.out.print("enter emp id:");
            int id = Integer.parseInt(br.readLine());
            System.out.print("enter name: "); String
            name = br.readLine();
            System.out.print("enter Salary : ");
            float sal = Float.parseFloat(br.readLine());
            Date d = new Date();
            Employee e = new Employee(id,name,sal,d);
 return e;
}
}
```

Compiling the Employee class



```
import java.io.*;
import java.util.*;
class Storeobj

{
    public static void main(String args[]) throws IOException
    {
            BufferedReader br = new BufferedReader(new
InputStreamReader(System.in));
            FileOutputStream fos = new FileOutputStream ("Objfile");
            ObjectOutputStream oos= new ObjectOutputStream (fos);
            System.out.print("how many objects ?");
            int n=Integer.parseInt(br.readLine());
            for(int i=0;i<n;i++)
            {
                    Employee e1 = Employee.getdata();
                    oos.writeObject(e1);
            }
    oos.close();
    }
}
```

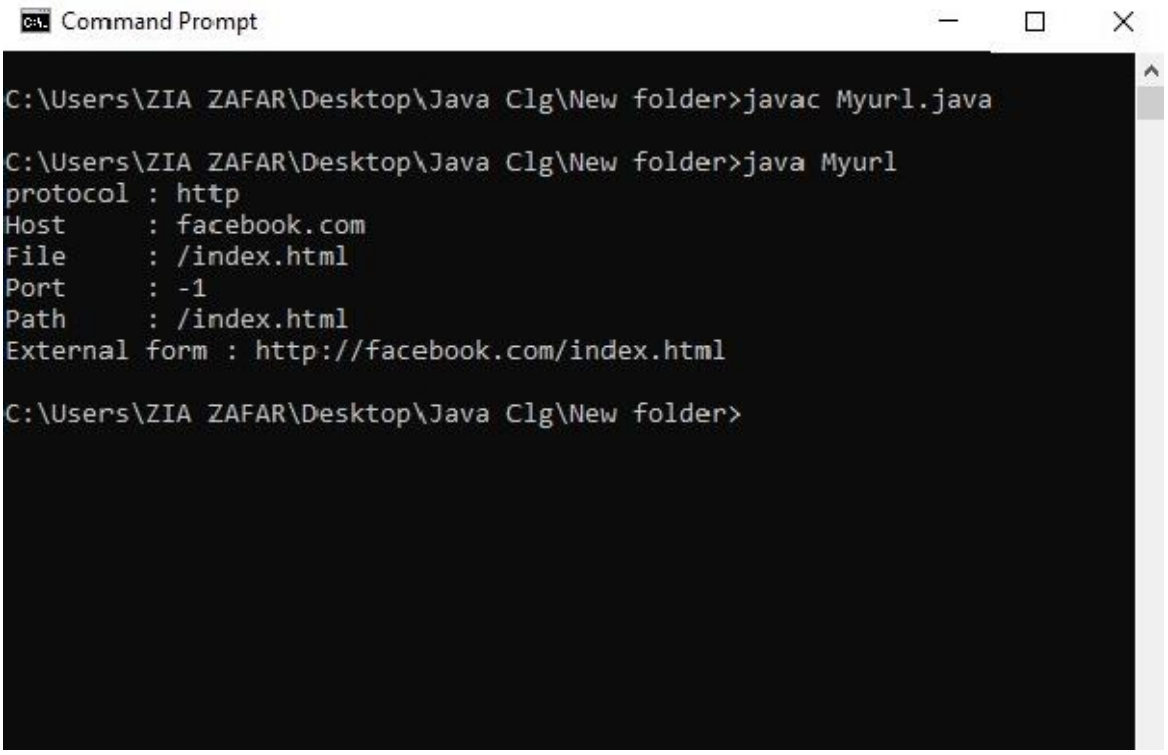
Serializing an object into the file objfile



26. Write a program to retrieve web page using URL?

```
import java.io.*;
import java.net.*;
class Myurl
{
 public static void main(String args[]) throws IOException
 {
  URL obj = new URL("http://facebook.com/index.html");
  System.out.println("protocol : "+obj.getProtocol());
  System.out.println("Host     : "+obj.getHost());
  System.out.println("File     : "+obj.getFile());
  System.out.println("Port     : "+obj.getPort());
  System.out.println("Path     : "+obj.getPath());
  System.out.println("External form : "+obj.toExternalForm());
}
}
```

Output:



27. Write a progrma to implement java network programming?(client and server program)

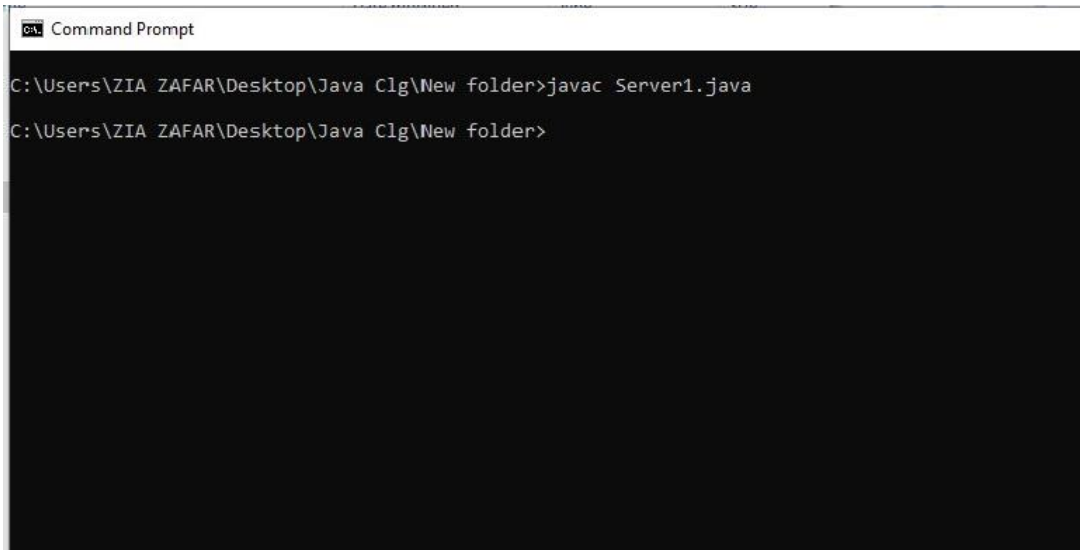
//server

```
import java.io.*;

import java.net.*;


class Server1

{

 public static void main(String args[]) throws Exception

 {

 ServerSocket ss = new ServerSocket(777);

 Socket s =ss.accept();

 System.out.println("Connection established");

 OutputStream obj = s.getOutputStream();

 PrintStream ps = new PrintStream(obj);

 String str = "Hello client";

 ps.println(str);

 ps.println("Bye");

 ps.close();

 ss.close();

 }

}
```

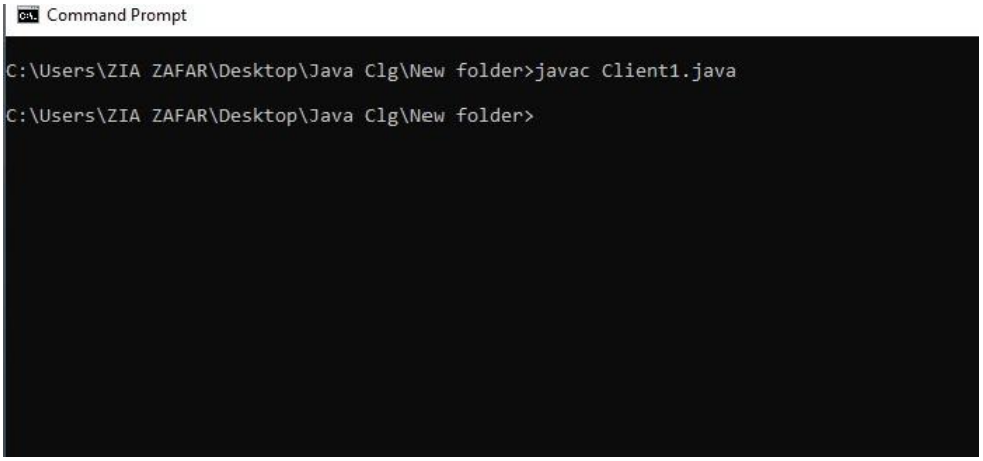
//creating server for sending some string to the client

```
import java.io.*;
import java.net.*;

class Client1
{
 public static void main(String args[]) throws Exception
 {
 //create client socket with some port
 Socket s = new Socket("localhost",777);
 //to read data from server
 InputStream obj=s.getInputStream();
 //to read from socket
 BufferedReader br = new BufferedReader(new InputStreamReader(obj));
 String str;
 while((str=br.readLine())!=null)
 System.out.println("from Server: "+str);
 br.close();
 s.close();
 }
}
```

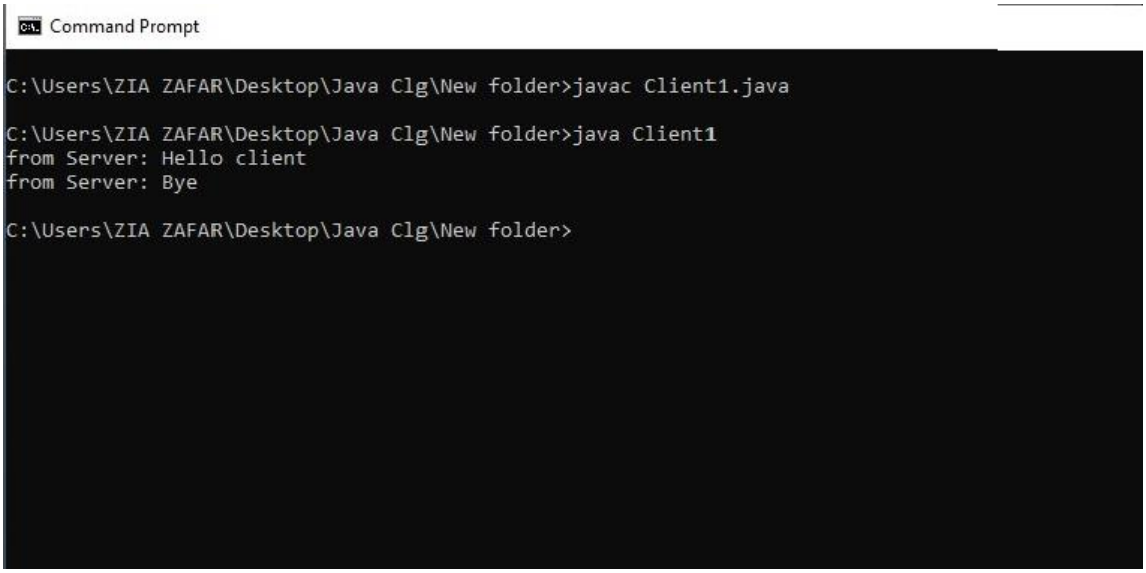
Server Compilation



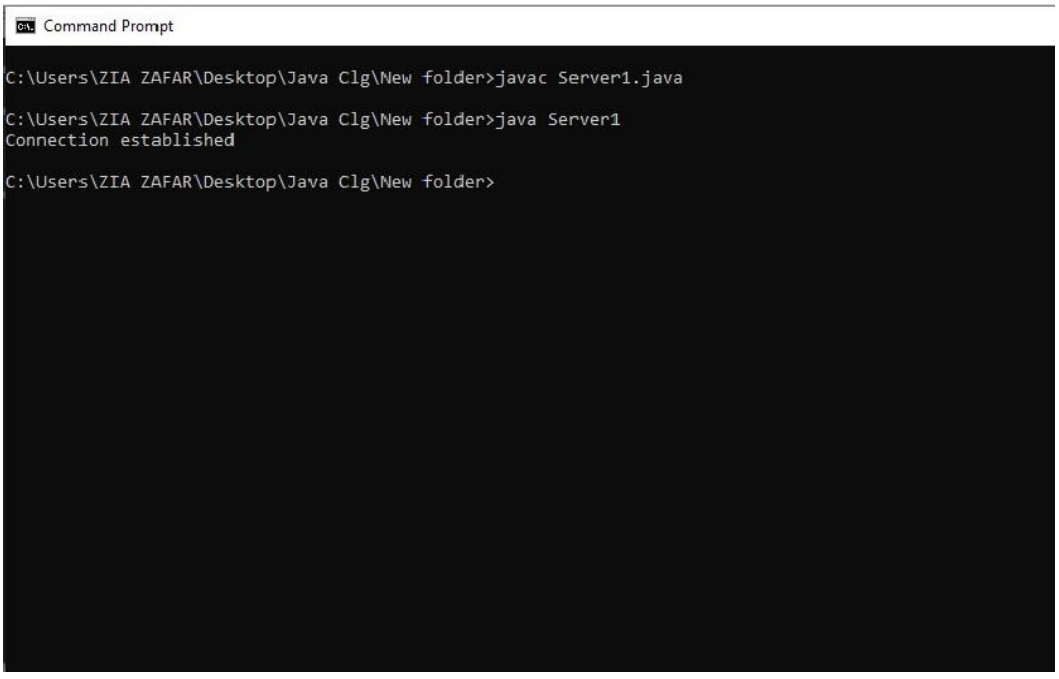
Client Compilation



Client File execution: -



Server File Execution: -

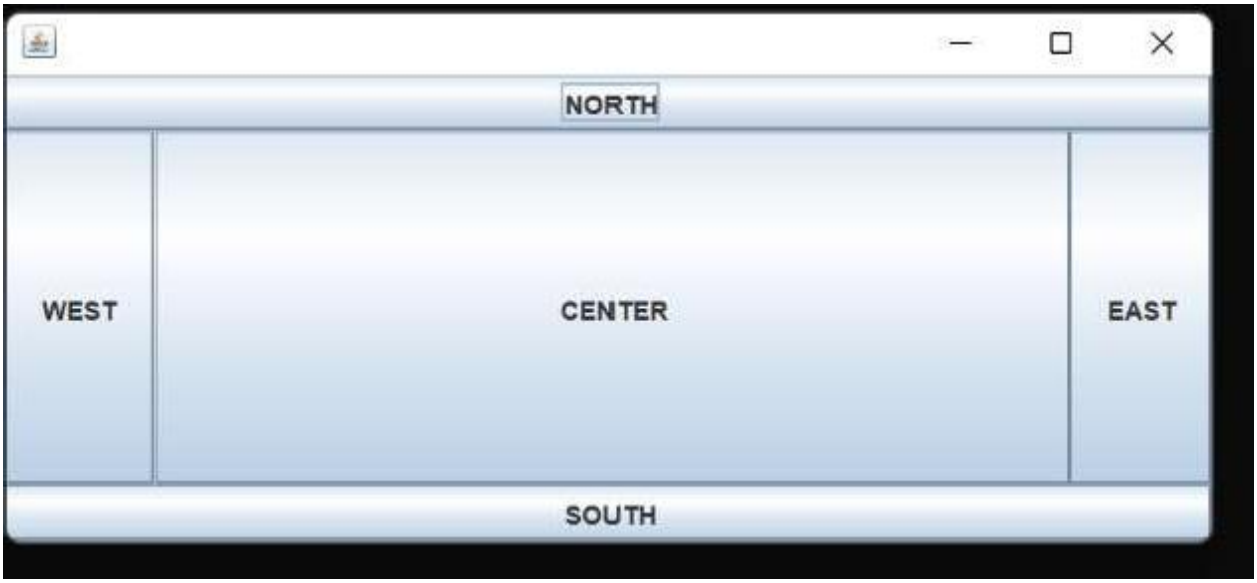


28. Write a program to implement border Layout?

```
import java.awt.*;
import javax.swing.*;
public class Border
{
JFrame f;
Border()
{
f = new JFrame();

// creating buttons
JButton b1 = new JButton("NORTH");; // the button will be labeled as NORTH
JButton b2 = new JButton("SOUTH");; // the button will be labeled as SOUTH
JButton b3 = new JButton("EAST");; // the button will be labeled as EAST
JButton b4 = new JButton("WEST");; // the button will be labeled as WEST

JButton b5 = new JButton("CENTER");; // the button will be labeled as CENTER
f.add(b1, BorderLayout.NORTH); // b1 will be placed in the North Direction
f.add(b2, BorderLayout.SOUTH); // b2 will be placed in the South Direction
f.add(b3, BorderLayout.EAST); // b2 will be placed in the East Direction
f.add(b4, BorderLayout.WEST); // b2 will be placed in the West Direction
f.add(b5, BorderLayout.CENTER); // b2 will be placed in the Center
f.setSize(300, 300);
f.setVisible(true);
}
public static void main(String[] args) {
new Border();
}
```

}

Output□

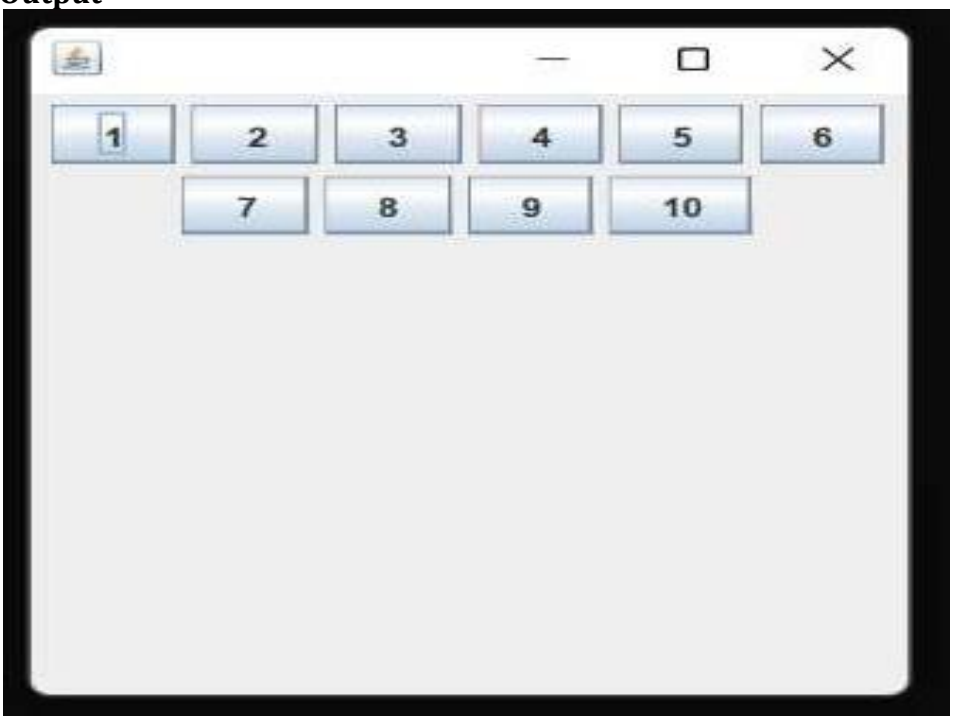


29. write a program to implement flow layout?

```
import java.awt.*;

import javax.swing.*;

public class FlowLayoutExample

{

JFrame frameObj;

// constructor

FlowLayoutExample()

{

// creating a frame object

frameObj = new JFrame();

// creating the buttons

JButton b1 = new JButton("1");

JButton b2 = new JButton("2");

JButton b3 = new JButton("3");

JButton b4 = new JButton("4");

JButton b5 = new JButton("5");

JButton b6 = new JButton("6");

JButton b7 = new JButton("7");

JButton b8 = new JButton("8");

JButton b9 = new JButton("9");
```

```
JButton b10 = new JButton("10");

// adding the buttons to frame

frameObj.add(b1); frameObj.add(b2); frameObj.add(b3); frameObj.add(b4);

frameObj.add(b5); frameObj.add(b6); frameObj.add(b7); frameObj.add(b8);

frameObj.add(b9); frameObj.add(b10);

frameObj.setLayout(new FlowLayout());

frameObj.setSize(300, 300);

frameObj.setVisible(true);

}

// main method

public static void main(String args[])

{

new FlowLayoutExample();}}
```

Output



30. write a program to Demonstrate Key Listener?

```
import java.awt.*;
import java.awt.event.*;
public class MyKeyEvents extends Frame implements KeyListener
{
String str;
public MyKeyEvents()
{
this.addWindowListener(new WindowAdapter()
{
public void windowClosing(WindowEvent we)
{
System.exit(0);
}});
str=new String();
addKeyListener(this);
}
public void keyPressed(KeyEvent k)
{
char ch=k.getKeyChar();
str+=ch;
repaint();
}
public void keyTyped(KeyEvent k)
{       }
public void keyReleased(KeyEvent k)
{       }
public void paint(Graphics g)
```

```
{
g.drawString(str,140,140);
}
public static void main(String[] args)
{
MyKeyEvents f = new MyKeyEvents();
f.setSize(550,550);
f.setVisible(true);

}}
```

Output