Dr.Babasaheb Ambedkar Open University

(Established by Government of Gujarat)

PGDMAD-104

Software Lab For Introduction to JAVA

Post Graduate Diploma in Mobile Application Development

2019

Software Lab for Introduction to Java

Dr. Babasaheb Ambedkar Open University

Software Lab for Introduction to JAVA

Course Writers

Dr. Kamalesh Salunke	Assistant Professor, Department of Computer Science, Gujarat Vidyapith, Ahmedabad
Dr. Vinod Desai	Assistant Professor, Department of Computer Science, Gujarat Vidyapith, Ahmedabad
Dr. Kajal Patel	Assistant Professor, Computer Engineering Department, Vishwakarma Engineering College, Ahmedabad
Dr. Alpaba Rajput	Assistant Professor, School of Computer Science, Dr. Babasaheb Ambedkar Open University, Ahmedabad

Content Reviewer

Prof. (Dr.) Nilesh Modi	Professor and Director, School of Computer Science, Dr. Babasaheb Ambedkar Open University, Ahmedabad

Editors

Prof. (Dr.) Nilesh Modi	Professor and Director, School of Computer Science, Dr. Babasaheb Ambedkar Open University, Ahmedabad
Dr. Himanshu Patel	Assistant Professor, School of Computer Science, Dr. Babasaheb Ambedkar Open University, Ahmedabad

ISBN: 978-81-940577-3-4

Printed and published by: Dr. Babasaheb Ambedkar Open University, Ahmedabad

Software Lab for Introduction to Java

Java Lab Manual

Unit Structure

1.1 LEARNING OBJECTIVES

- To make the student learn an object oriented way of solving problems using java.
- To understand basics of Java programming and execution.
- To implement reusability concepts through packages, inheritance and interfaces.
- To implement error-handling techniques using exceptions handling.
- To implement multithreading concepts for resource utilization in multitasking.
- To make the students to create the Graphical User Interface (GUI) using AWT components and arrangement of AWT components on container using layout manager.
- To make student to use arious I/O operations to read and write.

1.2 INTRODUCTION

This lab manual provides you information about the Software and hardware requirements for practical of java programming and implementation of various object oriented concepts through java programming. We have used notepad text editor which is a simple editor included with the windows operating system to create program code. You can use other advance IDE such as Netbeans, Eclipse for java programming which is convenient and improves your efficiency; since you are on learning phase of java, we recommend you stick to notepad.

1.3 REQUIREMENT SPECIFICATION

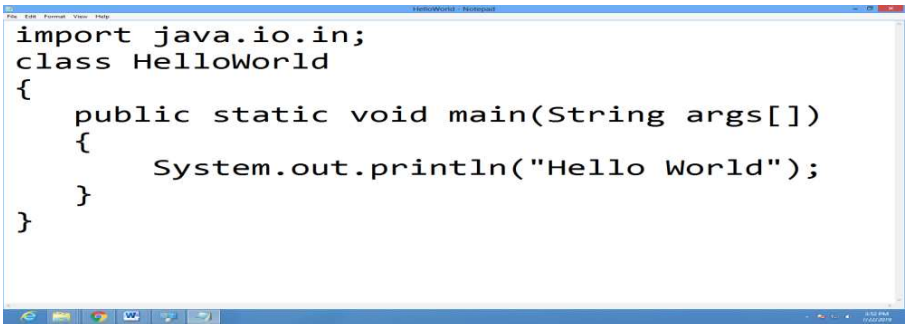
Hardware/Software Requirements

- Operating system: Flavour of any windows
- RAM: 128 MB
- Disk space: 124 MB for JRE; 2 MB for Java Update
- Processor: Minimum Pentium 2 266 MHz processor
- JDK Software Version: JDK1.7

1.4 CREATE YOUR FIRST JAVA PROGRAM

Step 1: Write a Java code in notepad.

For example: Write a simple JAVA Program to print "Hello World".



```
import java.io.in;
class HelloWorld
{
    public static void main(String args[])
    {
        System.out.println("Hello World");
    }
}
```

Step 2: Save file as HelloWorld.java (make sure to select file type as all files while saving the file in our working folder (For example file is saved in E:\java)).

Step 3: Go to command prompt.

Step 4: Use the following command to go the folder where Java program is stored.

cd E:\java

Step 5: Set path of Compiler. It is path in your system where java compiler is available.

For example, set path=C:\Program Files\Java\jdk1.7.0_23\bin;

Press enter

Step 6: Write following command for compilation of program.

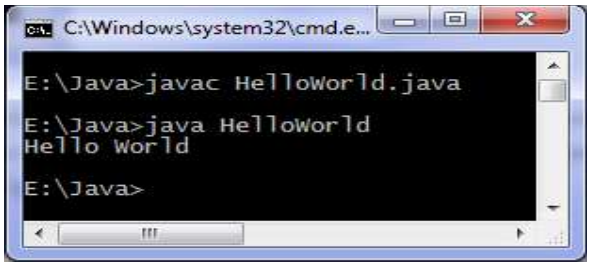
javac HelloWorld.java

Press Enter

Step 7: To run program use following command.

java HelloWorld

Output:



1.5 LIST OF PRACTICAL

1) **Write a JAVA program to implement concept of scanner class to get roll number and name of the student through keyboard and display them as output.**

```
import java.util.Scanner;
class  MyExa1
{
        public static void main(String args[])
        {
            int rno=null;
            String name=null;

             //creating object of Scanner class
            Scanner x=new Scanner(System.in);


            // Take input through keyboard using object of scanner class
            System.out.println("Enter name:");
            name=x.nextLine();
            System.out.println("Enter number:");
            rno=x.nextInt();
            System.out.println("\nRoll number:"+rno);
            System.out.println("\Name:"+name);
        }
}
```

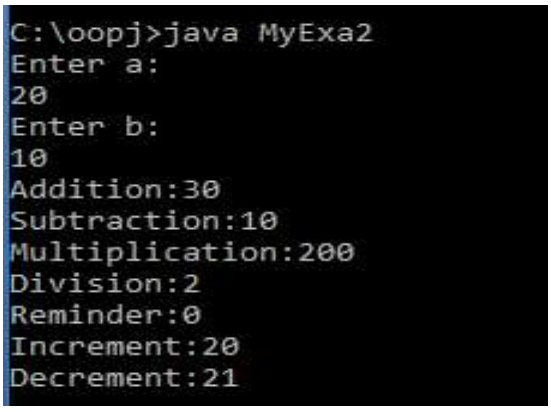
Output:

```
C:\oopj>javac MyExa1.java

C:\oopj>java MyExa1
Enter name:
Manan
Enter roll number:
12

Roll numebr:12

Name:Manan
```

2) Write a JAVA program to implement various Arithmetic Operators.

```
import java.util.Scanner;
public class MyExa2
{
        public static void main(String args[])
        {
            int a,b;
            Scanner sc=new Scanner(System.in);
            System.out.println("Enter a:");
            a=sc.nextInt();
            System.out.println("Enter b:");
            b=sc.nextInt();
           //performing arithmatic operation on input

            System.out.println("Addition:"+(a+b));
            System.out.println("Subtraction:"+(a-b));
            System.out.println("Multiplication:"+(a*b));
            System.out.println("Division:"+(a/b));
            System.out.println("Reminder:"+(a%b));
            System.out.println("Increment:"+(a++));
            System.out.println("Decrement:"+(a--));
        }
}
```

Output:



3) Write a JAVA program to check whether given number is even or odd.

```
import java.util.Scanner;
public class Odd_Even
{
        public static void main(String[] args)
        {
                int n;
                Scanner s = new Scanner(System.in);
                System.out.print("Enter the number to check even/odd:");
                n = s.nextInt();
                if(n % 2 == 0)
                {
                   System.out.println("The given number "+n+" is Even ");
                }
                else
                {
                   System.out.println("The given number "+n+" is Odd ");
                }
        }
}
```

4) Write a JAVA program to find largest number among three numbers.

```
import java.util.Scanner;
public class LargestOfThreeNumbers
{
        public static void main(String args[])
        {
                Scanner sc =new Scanner(System.in);
                System.out.println("Enter 1st number :");
                int a = sc.nextInt();
                System.out.println("Enter 2nd number :");
                int b = sc.nextInt();
                System.out.println("Enter 3rd number :");
                int c = sc.nextInt();
                 if ( a > b && a > c )
```

```
                {
                        System.out.println("Largest number is ::"+ a);
                 }
                else if ( b > a && b > c )
                {
                         System.out.println("Largest number is ::"+ b);
                }
                else if ( c > a && c > b )
                {
                        System.out.println("Largest number is ::"+ c);
                }
                else
                {
                        System.out.println("Cannot validate");
                }
            }
}
```

5) Write a JAVA program to wish "good morning", "good afternoon", "good evening" or "good night" from current time using if…else ladder.

```
import java.util.Calendar;
import java.util.GregorianCalendar;

public class DisplayDateTime
{
        public static void main(String[] args)
        {
                GregorianCalendar time = new GregorianCalendar();
                int hour = time.get(Calendar.HOUR_OF_DAY);
                 int min = time.get(Calendar.MINUTE);
                 int day = time.get(Calendar.DAY_OF_MONTH);
                int month = time.get(Calendar.MONTH) + 1;
                int year = time.get(Calendar.YEAR);
```

```
                System.out.println("The current time is \t" + hour + ":" + min);
                System.out.println("Date is \t"+month +"/"+ day "/"+ year);
                 if (hour < 12)
                        System.out.println("Good Morning!");
                else if (hour < 17 && !(hour == 12))
                        System.out.println("Good Afternoon");
                 else if (hour == 12)
                        System.out.println("Good Noon");
                else
                        System.out.println("Good Evening");
        }
}
```

6) Write a JAVA program which reads two integers and perform the arithmetic operation on them based on user's choice using "switch…case" statement.

```
import java.util.Scanner;
public class Arithmetic_Operators
{
        public static void main(String args[])
        {
                Scanner s = new Scanner(System.in);
                while(true)
                 {
                        System.out.println("");
                        // Read two integers
                        System.out.print("Enter the first number : ");
                        int x = s.nextInt();
                        System.out.print("Enter the second number : ");
                        int y = s.nextInt();
                        // Creating choices
                        System.out.println("Choose the operation");
                        System.out.println("Choose 1 for Addition")
                        System.out.println("Choose 2 for Subtraction");
```

```
System.out.println("Choose 3 for Multiplication");
System.out.println("Choose 4 for Division");
 System.out.println("Choose 5 for Modulus");
 System.out.println("Choose 6 for Exit");
 int n = s.nextInt();
 switch(n)
 {
                case 1:
                     int add;
                     add = x + y;
                     System.out.println("Result : "+add);
                     break;

                case 2:
                     int sub;
                     sub = x - y;
                     System.out.println("Result : "+sub);
                     break;

                case 3:
                     int mul;
                     mul = x * y;
                     System.out.println("Result : "+mul);
                     break;

                case 4:
                     float div;
                     div = (float) x / y;
                     System.out.print("Result : "+div);
                     break;
                case 5:
                     int mod;
                     mod = x % y;
                     System.out.println("Result : "+mod);
                     break;
```

```
                                        case 6:
                                                System.exit(0);
                                }
                        }
                }
}
```

7) Write a JAVA program to find factorial of given number using for and while loop.

```
import java.util.Scanner;
public class Ex_loop
{
        public static void main(String args[])
        {
                long fact=1;
                Scanner sc=new Scanner(System.in);
                System.out.println("Enter n:");
                int n=sc.nextInt();
                //using for loop
                for(int i=1;i<=n;i++)
                fact*=i;
                System.out.println("for:Factorial of "+n+" is :"+fact);
                //using while loop
                fact=1;
                int i=1;
                while(i<=5)
                {
                        fact*=i;
                         i++;
                }
                System.out.println("while:Factorial of "+n+" is :"+fact);
        }
}
```

Output:

```
C:\ajava\oopj>javac Ex_loop.java

C:\ajava\oopj>java Ex_loop
Enter n:
5
for:Factorial of 5 is :120
while:Factorial of 5 is :120
```

8) Write a JAVA program to create following pattern using nesting of loop.

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

```
class Pattern
{
        public static void main(String[] args)
        {
                int rows = 5;
                for(int i = 1; i <= rows; ++i)
                {
                        for(int j = 1; j <= i; ++j)
                        {
                                System.out.print(j + " ");
                        }
                        System.out.println("");
                }
        }
}
```

9) Write a JAVA program to implement break and continue statements.

```
public class BreakAndContinue
{
```

```
    public static void main(String args[])
    {
       int[] numbers= new int[]{101,102,103,104,105,106,107,108,109,110};
       int add = 0;
       for(int i=0; i< numbers.length; i++)
         {
                System.out.println("iteration: " + i);
                 if(i == 5)
                 {
                        System.out.println("calling break statement");
                        break;
                 }
                 if(i%2 != 0)
                 {
                        add = add + numbers[i];
                        System.out.println("callingcontinue statement");
                        continue;
                 }
                System.out.println("Last line of loop executed only for even
                                        number of iterations: " + numbers[i]);
         }
       System.out.println("This is outside the loop, sum: " + add);
    }
}
```

Output:

```
iteration: 0
Last line of loop executed only for even number of iterations: 101
iteration: 1
calling continue statement
iteration: 2
Last line of loop executed only for even number of iterations: 103
iteration: 3
calling continue statement
```

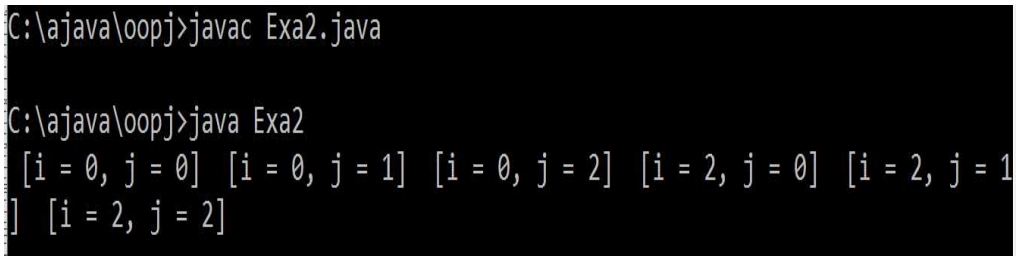
```
iteration: 4
Last line of loop executed only for even number of iterations: 105
iteration: 5
calling break statement
This is outside the loop, sum: 206
```

10) Write a JAVA program to implement labeled continue and break statements.

```
public class Exa2
{
        public static void main(String args[])
        {
                //Label to continue
                first:
                for (int i = 0; i < 3; i++)
                {
                        for (int j = 0; j< 3; j++)
                        {
                                if(i == 1)
                                        continue first;
                                System.out.print(" [i = " + i + ", j = " + j + "] ");
                        }
                  }
                System.out.println();
                //Label to Break
                second:
                for (int i = 0; i < 3; i++)
                {
                        for (int j = 0; j< 3; j++)
                        {
                                if(i == 1)
                                        break second;
                               System.out.print(" [i = " + i + ", j = " + j + "] ");
                        }
```

```
            }
        }
}
```

Output:



11) Write a JAVA program to sort 5 integers in ascending order using array.

```
public class Ex_ary1
{
        public static void main(String args[])
         {
                int[] a={23,45,67,8,3};
                System.out.print("Before Sorting :");
                for(int i=0;i<a.length;i++)
                        System.out.print(" "+a[i]);
                for(int i=0;i<a.length;i++)
                        for(int j=i+1; j<a.length; j++)
                        {
                                if(a[i]>a[j])
                                {
                                        int t=a[i];
                                        a[i]=a[j];
                                        a[j]=t;
                                }
                        }
                        System.out.print("\nAfter Sorting :");
                        for(int i=0;i<a.length;i++)
                                System.out.print(" "+a[i]);
        }
```

```
}
```

Output:

```
C:\ajava\oopj>javac Ex_ary1.java

C:\ajava\oopj>java Ex_ary1
Before Sorting : 23 45 67 8 3
After Sorting : 3 8 23 45 67
```

12) Write a JAVA program to add two 3x3 matrix.

```
public class Ex_ary2
{
        public static void main(String args[])
        {
                //Create two 3x3 array
                int[][] a={{1,1,1},{2,2,2},{3,3,3}};
                int[][] b={{4,4,4},{5,5,5},{6,6,6}};
                int[][] c=new int[3][3];

                for(int i=0;i<3;i++)
                for(int j=0;j<3;j++)
                        c[i][j]=a[i][j]+b[i][j];

                System.out.print("Result matrix:\n");
                for(int i=0;i<3;i++)
                {
                        for(int j=0;j<3;j++)
                                System.out.print(c[i][j]+" ");
                        System.out.print("\n");
                }
        }
}
```

Output:

```
C:\ajava\oopj>javac Ex_ary2.java

C:\ajava\oopj>java Ex_ary2
Result matrix:
5 5 5
7 7 7
9 9 9
```

13) Write a JAVA program to implement simple java class which includes methods to get student information and print the details on screen.

```
class Student
{
        int rollNember;
        String name;
        String course;
        void getData(int r, String n, String c)     //Method
        {
                rollNumber = r;
                name = n;
                course = c;
        }
        void printData()                    //Method
        {
                System.out.println ( rollNumber );
                System.out.println ( name );
                System.out.println ( course );
        }
}

class Exa_Cls
{
        Public static void main(String args[])
        {
                Student s1 = new Student( );    //object s1 is created
                S1.getData(1, ”manan” , ”civil” );
                s1.printData();
```

```
    }
}
```

Output:

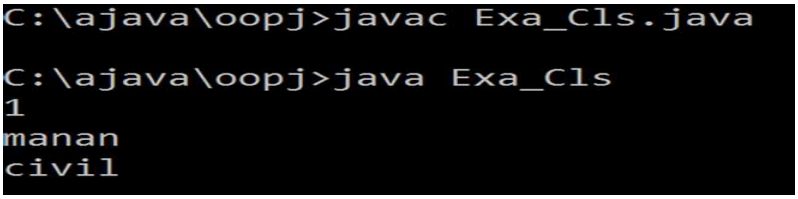
```
C:\ajava\oopj>javac Exa_Cls.java

C:\ajava\oopj>java Exa_Cls
1
manan
civil
```

14) Write a JAVA program to implement default and parameterized constructor in java.

```
class Student
{
    int rollNember;
    String name;
    String course;
    Student()        //default constructor
    {
        rollNumber = 0;
        name = "";
        course = "";
    }
    Student(int r, String n, String c)      //parameterized constructor
    {
        rollNumber = r;
        name = n;
        course = c;
    }
    void printData()   //Method to print data
    {
        System.out.println( rollNumber );
        System.out.println( name );
        System.out.println( course );
    }
```

```
}

class Exa_Cls
{
        Public static void main(String args[])
        {
                Student s1 = new Student(1,”manan”,”civil”);
                s1.printData();
        }
}
```

Output:



15) Write a JAVA program to implement “this” keyword.

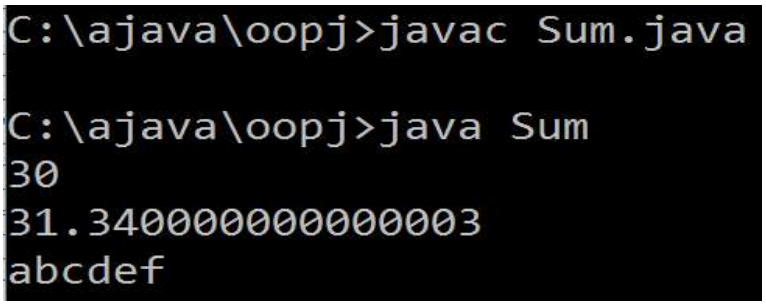
```
class  abc
{
        int a,b,c;
        abc(){ a = 0; b = 0; c = 0;}
        abc( int a,int b, int c)
        {
                this.a = a;
                this.b = b;
                this.c = c;
        }
}

class MyExa
{
        public static void main(String args[])
        {
                abc x = new abc(1,2,3);
        }
```

```
}
```

16) Write a JAVA program to implement concept of method overloading.

```
class Add
{
        int  addition(int a, int b) { return ( a + b ); }
        float addition(float a, float b) { return ( a + b ); }
        String addition(String a, String b)
        {
                return a + b;
         }
}

public class Sum
{
        public static void main(String args[])
        {
                Sum s1 = new Sum();
                System.out.println(s1.addition(10, 20));
                System.out.println(s1.addition(10.56 ,20.78));
                System.out.println(s1.addition("abc", "def"));
        }
}
```

Output:



17) Write a JAVA program which use static variable to count number of object of class created using.

```
class Student
{
```

```
    static int n = 0;
    int rollNumber = 0;
    String name = "";
    String course = "";
    Student()                           //default constructor
    {
        rollNumber = 0;
        name = "";
        course = "";
        Student.n++;
    }
    Student(int r, String n, String c)  //parameterized constructor
    {
        rollNumber = r;
        name = n;
        course = c;
        Student.n++;
    }
    void printData()                    //Method
    {
        System.out.println(rollNumber);
        System.out.println(name);
        System.out.println(course);
    }

}
class Exa_Cls
{
    public static void main(String args[])
    {
        Student s1 = new Student(1,"manan","civil");
        Student s2 = new Student();
        System.out.println("number of objects:"+Student.n);
    }
```

```
}
```

Output:

```
C:\ajava\oopj>javac Exa_Cls.java

C:\ajava\oopj>java Exa_Cls
number of objects:2
```

18) Write a JAVA program to implement static method.

```
class A
{
        static  void sum(int a, int b)
        {
                int c = a + b;
                printA(c);          // must be static
        }
        static void printA(int x)
        {
                System.out.println(" sum : " + x );
        }
}
public class ExStatic2
{
        public static void main(String args[])
        {
                A.sum(10,30);
        }
}
```

Output:

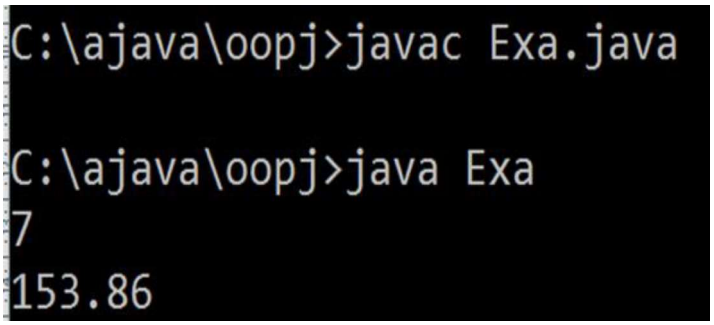
```
C:\ajava\oopj>java ExStatic1
 sum : 40
```

19) Write a JAVA program to implement concept of nesting of methods.

```
import java.util.Scanner;
class Circle
{
        int radius;
```

```
        void getRadius()
        {
                Scanner sc=new Scanner(System.in);
                Radius = sc.nextInt();
        }
        double area()
        {
                getRadius();                    // calling another method
                return(3.14*radius*radius);
        }
}

public class Exa
{
        public static void main(String args[])
        {
                Circle c1 = new Circle();
                System.out.println (c1.area());
        }
}
```

Output:



20) Write a JAVA program to implement concept of vector class.

```
import java.util.Vector;
public class Exa1
{
        public static void main(String args[])
        {
```

```
        Vector v1=new Vector(20);
        v1.add("A");
        v1.add("C");
        v1.add(1,"B");
        System.out.println("size: " + v1.size());
        System.out.println("capacity " + v1.capacity());

        Vector v2=(Vector)v1.clone();
        System.out.println(" Vector v1 " + v1);
        System.out.println(" Vector v2 " + v2);

        v1.addAll(v2);
        System.out.println(" Vector v1.addAll(v2): " + v1);
        System.out.println( " Is B in v1:"+ v1.contains("B"));
        System.out.println(" element at 2 : " + v1.get(2));
        System.out.println(" POsition of A : " + v1.indexOf("A"));
        System.out.println(" Check for empty v1: " + v1.isEmpty());
        System.out.println(" Last index of A: " + v1.lastIndexOf("A"));

        v1.remove("C");
        System.out.println(" After removing C in v1 : " + v1);
        System.out.println("Compare v1 and v2 : " + v1.equals(v2));
        System.out.println( " Fisrt element of v1 : " + v1.firstElement());
        System.out.println(" Last element of v1 : " +  v1.lastElement());
        System.out.println(" v1 to String : " + v1.toString());
    }
}
```

Output:

```
size: 3
capacity 20
Vector v1 [A, B, C]
Vector v2 [A, B, C]
Vector v1.addAll(v2): [A, B, C, A, B, C]
 Is B in v1:true
 element at 2 : C
 POsition of A : 0
Check for empty v1: false
Last index of A: 3
After removing C in v1 : [A, B, A, B, C]
Compare v1 and v2 : false
 Fisrt element of v1 : A
 Last element of v1 : C
 v1 to String : [A, B, A, B, C]
```

21) Write a JAVA program to implement concept of unboxing classes.

```
class Exa3
{
    public static void main(String args[])
    {
        //Autoboxing
        byte a = 10;
        Byte aobj = new Byte(a);

        int b = 289;
        Integer bobj = new Integer(b);

        float c = 508.5f;
        Float cobj = new Float(c);

        double d = 90.3;
        Double dobj = new Double(d);

        char e='x';
        Character eobj=e;


        System.out.println("Autoboxing");
        System.out.println(aobj);
```

```
        System.out.println(bobj);
        System.out.println(cobj);
        System.out.println(dobj);
        System.out.println(eobj);

        //Unboxing
        byte v = aobj;
        int w = bobj;
        float x = cobj;
        double y = dobj;
        char z = eobj;

        System.out.println("Unboxing");
        System.out.println(v);
        System.out.println(w);
        System.out.println(x);
        System.out.println(y);
        System.out.println(z);
    }
}
```

Output:

```
C:\ajava\oopj>javac Exa3.java

C:\ajava\oopj>java Exa3
Autoboxing
10
289
508.5
90.3
x
Unboxing
10
289
508.5
90.3
x
```

22) Write a JAVA program to implement concept of Wrapper classes and its methods.

```
public class ExWrap1
{
        public static void main(String args[])
        {
            // example of valueOf
            System.out.println("valueOf converts String into Wrapper class object");
            Integer a=Integer.valueOf("100");
            Byte b=Byte.valueOf("8");
            Double c=Double.valueOf("10.80");
            System.out.println("Integer: " + a);
            System.out.println("Byte: " + b);
            System.out.println("Double: " + c);

            System.out.println(" valueOf converts String with differnt base into
                                Wrapper class object");
            Integer a1=Integer.valueOf("1110",2);
            System.out.println("Integer: " + a1);

            System.out.println(" valueOf converts primitive data type into
                                Wrapper class object");
            Integer a2 = Integer.valueOf(100);
            Double b2 = Double.valueOf(34.6);
            System.out.println("Integer: " + a2);
            System.out.println("Integer: " + b2);
        }
}
```

Output:

```
C:\ajava\oopj>java ExWrap1
 valueOf converts String into Wrapper class object
Integer: 100
Byte: 8
Double: 10.8
 valueOf converts String with differnt base into Wrapper class object
Integer: 14
 valueOf converts primitive data type into Wrapper class object
Integer: 100
Integer: 34.6
```

23) Write a JAVA program which converts one numeric datatype to other type.

```
public class ExWrap2
{
        public static void main(String args[])
        {
                System.out.println(" xxxValue functions converts one numeric
                                        datatype into other ");
                Integer x = new Integer(122);
                int  y = x.intValue();
                byte z = x.byteValue();          // Convert to byte
                float a = x.floatValue();        // Convert to float
                System.out.println(" int :" + y);
                System.out.println(" byte :" + z);
                System.out.println(" float :" + a);
        }
}
```

Output:

```
C:\ajava\oopj>java ExWrap2
 xxxValue functions converts one numeric datatype into other
 int :122
 byte :122
 float :122.0
```

24) Write a JAVA program which converts String to primitive data type.

```
public class ExWrap3
{
        public static void main(String args[])
        {
                System.out.println(" parseXXX functions converts String to
                                        primitive data type");
                int x  = Integer.parseInt("123");
                double y = Double.parseDouble("123.56");
                boolean z = Boolean.parseBoolean("false");

                System.out.println(" int :"  +  x);
                System.out.println(" double :" + y);
```

```
        System.out.println(" boolean :" + z);

        System.out.println("functions converts a String representation of a
                            number with  base radix into primitive data types.");
        int x1=Integer.parseInt("1111",2);
        System.out.println(" decimal of 1111 is int :" + x1);
    }
}
```

Output:

```
C:\ajava\oopj>java ExWrap3
 parseXXX functions converts String to primitive data type
 int :123
 double :123.56
 boolean :false
 parseXXX functions converts a String representation of a number with  base ra
dix into primitive data types.
 decimal of 1111 is int :15
```

25) Write a JAVA program to convert wrapper class object into String.

```
public class ExWrap4
{
    public static void main(String args[])
    {
        System.out.println( "non static toString functions converts wrapper
                            object to String ");
        Double d = new Double(123.88);
        String s = d.toString();
        System.out.println(" String : "  +  s);

        System.out.println(" static toString functions converts primitive
                            data type into String ");
        String x1 = Double.toString(123.89);
        System.out.println(" String :"  +  x1);
    }
}
```

Output:

```
C:\ajava\oopj>java ExWrap4
non static toString functions converts wrapper object to String
 String : 123.88
static toString functions converts primitive data type into String
 String :123.89
```

26) Write A JAVA program to implement single inheritance.

```
class A
{
        int a;
        A() {a = 0; }
        A( int x ) { a = x; }
        void printA()
        {
                System.out.println(" a = " + a);
        }
}

class B extends A                    \\ Single inheritance
{
        int b;
        B() { a = 0;  b = 0; }
        B(int x, int y) { a = x;  b = y; }
        void printB()
        {
                printA();
                System.out.println(" b = " + b);
        }
}
```

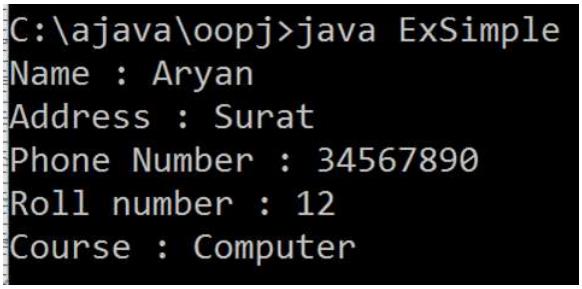
27) Write a JAVA program to implement concept of super keyword.

```
class Person
{
```

```
        String name;
        String address;
        int phno;
        Person(){ name = ""; address = ""; phno = 0;}
        Person(String n, String a, int p){ name = n; address = a; phno = p;}
        void printP()           \\Method
        {
                System.out.println("Name : " + name);
                System.out.println("Address : " + address);
                System.out.println("Phone Number : " + phno);
        }
}

class Student extends Person            \\Inherit class Person
{
         int rollNumber;
         String course;
         Student(){ super(); rollNumber = 0; course = "";}
         Student(String n, String a, int p,int r, String c)
         {
                super(n,a,p);           \\Call parent class constructor
                rollNumber = r;
                course = c;
          }
         void printS()
         {
                  printP();
                  System.out.println("Roll number : " + rollNumber);
                  System.out.println("Course : " + course);
         }
}

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

```
        {
            Student s1=new Student("Aryan","Surat",34567890,12,"Computer");
            s1.printS();
        }
}
```

Output:



28) Write A JAVA program to implement Hierarchical inheritance.

```
class A
{
        int a;
        A() {a = 0; }
        A( int x ) { a = x; }
        void printA() { System.out.println(" a = " + a); }
}

class B extends A                    \\Inherit class A
{
        int b;
        B() {super();  b = 0; }
        B(int x, int y) { super(x);  b = y; }
        void printB()
        {
                super.printA();      \\Call parent class method
                System.out.println(" b = " + b);
          }
  }

  class C extends A                  \\Inherit class A
```

```
{
        int c;
        C() {super();  c = 0; }
        C(int x, int z) { super(x);  c = z; }
        void printB()
        {
                super.printA();          \\Call parent class method
                System.out.println(" c = " + c);
        }
}
public class ExInh
{
        public static void main(String args[])
        {
                B b1 = new B(10,20);
                C c1 = new C(23,34);
                b1. printB();
                c1. printC();
        }
}
```

Output:

```
C:\ajava\oopj>java ExInh
 a = 10
 b = 20
 a = 23
 c = 34
```

29) Write A JAVA program to implement concept of method overriding.

```
class A
{
        int a;
        A() {a = 0; }
```

```
        A( int x ) { a = x; }
        void printData() { System.out.println(" a = " + a); }
}

class B extends A                \\Inherit class A
{
        int b;
        B() {super();  b = 0; }
        B(int x, int y) { super(x);  b = y; }
        void printData()         //the method of parent class is redefined
        {
                System.out.println(" a = " + a);
                System.out.println(" b = " + b);
        }
}

public class ExInh
{
        public static void main(String args[])
        {
                A a1 = new A(10);
                B b1 = new B(23,34);
                a1. printData();
                b1. printData();
        }
}
```

Output:

```
C:\ajava\oopj>java ExInh
 a = 10
 a = 23
 b = 34
```

30) Write A JAVA program to implement concept of final variable.

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

```
        final int x = 80;
        int[] a = new int [ x ];   //we can use x but can not modify it
        x = 90;   //gives compilation error as x is constant
    }
}
```

Output:

```
C:\ajava\oopj>javac ExInh.java
ExInh.java:8: error: cannot assign a value to final variable x
x=90;   //this gives compilation error as x is constant
^
1 error
```

31) Write a JAVA program to implement concept of Abstract class.

```
abstract class Shape
{
    double ar;
    double peri;
    Shape()
    {
        ar = 0.0;
        peri = 0.0;
    }
    final double PI=3.14;
    abstract void area();
    abstract void perimeter();
    void printArea()
    {
        System.out.println("Area : " + ar);
    }
    void printPerimeter()
    {
        System.out.println("Perimeter : " + peri);
    }
```

```
}
class Circle extends Shape   //extends abstract class
{
        int r;
        Circle(){ r = 0; }
        Circle(int r){ this.r = r; }
        void area()
        {
                ar = PI*r*r;
        }
        void perimeter()
        {
                peri = 2*PI*r;
        }
}
class Square extends Shape     //extends abstract class
{
        int s;
        Square(){ s = 0; }
        Square(int s){ this.s = s; }
        void area()
        {
                ar = s * s;
        }
        void perimeter()
        {
                peri = 4 * s;
        }

}
public class ExInh1
{
```

```
        public static void main(String args[])
        {
                Shape c1=new Circle(2);
                c1.area();
                c1.printArea();
                c1.perimeter();
                c1.printPerimeter();
                c1=new Square(2);
                c1.area();
                c1.printArea();
                c1.perimeter();
                c1.printPerimeter();
        }
}
```

Outtput:

```
C:\ajava\oopj>java ExInh1
Area : 12.56
Perimeter : 12.56
Area : 4.0
Perimeter : 8.0
```

32) Write A JAVA program to implement interface.

```
interface Shape
{
        double PI=3.14;
        public double area();
        public double perimeter();
        public void printData();
}

class Circle implements Shape                    \\interface implemented by class
{
         int r;
         Circle(){ r = 0; }
```

```
        Circle(int r){ this.r = r; }
        public double area()
        {
                return PI*r*r;
        }
        public double perimeter()
        {
                return 2*PI*r;
        }
        public void printData()
        {
                System.out.println("Area : " + area());
                System.out.println("Perimeter : " + perimeter());
        }
}

class Square implements Shape          \\interface implemented by class
{
        int s;
        Square(){ s = 0; }
        Square(int s){ this.s = s; }
        public double area()
        {
                return s*s*1.0;
        }
        public double perimeter()
        {
                return 4.0*s;
        }
        public void printData()
        {
                System.out.println("Area : " + area());
                System.out.println("Perimeter : " + perimeter());
        }
```

```
 }

public class ExInf
{
        public static void main(String args[])
        {
                Shape c1=new Circle(5);
                c1.printData();
        }
}
```

Output:

```
C:\ajava\oopj>java ExInf
Area : 78.5
Perimeter : 31.400000000000002
```

33) Write a JAVA program to implement concept of object as function argument.

```
class A
{
        int a;
        A() { a = 0; }
        A(int x) { a = x; }
        void printA()
        {
                System.out.println(“ a  =  “  +  a);
        }
}

public class ExObjArg
{
          public static void main(String args[])
          {
                A a1 = new A(9);
                A a2 = new A(8);
```

```
            add(a1 , a2);
        }
        static void add(A a1, A a2) //function with objects as arguments
        {
            int sum = a1.a + a2.a;
            System.out.println(" sum  =  "  +  sum);
        }
}
```

Output:

```
C:\ajava\oopj>java ExObjArg
 sum   =   17
```

34) Write a JAVA program to implement concept of function return object.

```
class A
{
    int a;
    A() { a = 0; }
    A(int x) { a = x; }
    void printA()
    {
        System.out.println(" a  =  "  +  a);
    }
}
public class ExObjArg
{
    public static void main(String args[])
    {
        A a1 = new A(9);
        A a2 = new A(8);
        A a3 = add(a1 , a2);
        a3.printA();
    }
    //function with objects as arguments and returns object
```

```
    static A add(A a1, A a2)
    {
        A a3 = new A();
        a3.a = a1.a + a2.a;
        return a3;
    }
}
```

Output:

```
C:\ajava\oopj>java ExObjRet
 a   =   17
```

35) Write a JAVA program to find a factorial of given number using recursion.

```
public class Factorial
{
    public static void main(String args[])
    {
        long f = factorial(5);
        System.out.println(f);
    }
    static long  factorial( int n)
    {
        long fact = 1;
        if( n == 1 || n == 0)
            return fact;
        else
        {
            fact = n * factorial(n-1);
            return fact;
        }
    }
}
```

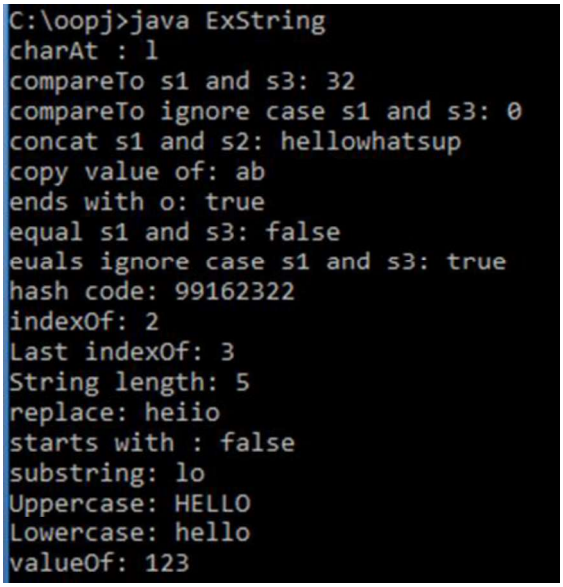
Output:

```
C:\ajava\oopj>java Factorial
120
```

36) Write a JAVA program to implement methods of string class.

```
public class ExString
{
        public static void main( String args [])
        {
                String s1 = "hello";
                String s2 = "whatsup";
                String s3 = new String( "Hello" );
                char[] s4  = { 'a' , 'b' };
                System.out.println( "charAt : " + s1.charAt(2));
                System.out.println( "compareTo s1 and s3: " + s1.compareTo(s3));
                System.out.println( "compareTo ignore case s1 and s3: " +
                                        s1.compareToIgnoreCase(s3));
                System.out.println( "concat s1 and s2: " + s1.concat(s2));
                System.out.println( "copy value of: " + String.copyValueOf(s4));
                System.out.println( "ends with o: " + s1.endsWith("o"));
                System.out.println( "equal s1 and s3: " + s1.equals(s3));
                System.out.println( "euals ignore case s1 and s3: " +
                                        s1.equalsIgnoreCase(s3));
                byte[] b = s1.getBytes();
                System.out.println( "hash code: " + s1.hashCode());
                System.out.println( "indexOf: " + s1.indexOf('l'));
                System.out.println( "Last indexOf: " + s1.lastIndexOf('l'));
                System.out.println( "String length: " + s1.length());
                System.out.println( "replace: " + s1.replace('l','i'));
                System.out.println( "starts with : " + s1.endsWith("h"));
                System.out.println( "substring: " + s1.substring(3));
                                        char ar1 [] = s1.toCharArray();
                System.out.println(" Uppercase: " + s1.toUpperCase());
                System.out.println(" Lowercase: " + s1.toLowerCase());
                System.out.println(" valueOf: " + String.valueOf(123));
        }
}
```

Output:

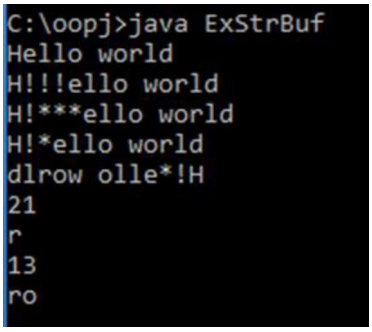


37) Write a JAVA program to implement methods of stringbuffer class.

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

                StringBuffer s1 = new StringBuffer("Hello");
                s1.append( " world" );
                System.out.println( s1 );
                s1.insert ( 1 , "!!!");
                System.out.println( s1 );
                s1.replace ( 2, 4, "***" );
                System.out.println( s1 );
                s1.delete( 2, 4);
                System.out.println( s1 );
                s1.reverse();
                System.out.println( s1 );
                System.out.println( s1.capacity() );
                System.out.println( s1.charAt(2) );
                System.out.println( s1.length() );
                System.out.println( s1.substring(2,4) );
```

```
        }
}
```

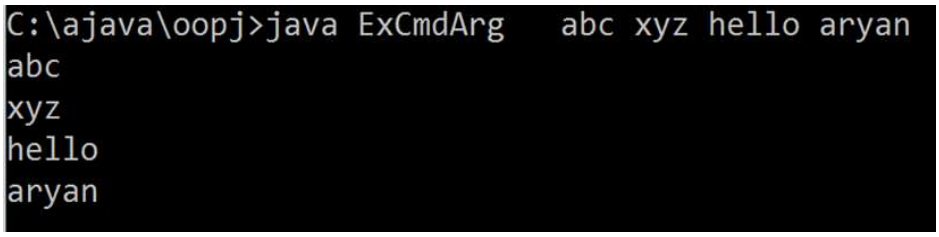
Output:



38) Create a JAVA program to implement concept of command line argument.

```
public class ExCmdArg
{
        public static void main(String args[])
        {
        for( int j = 0; j < args.length ; j++)
         System.out.println(args[j]);          \\print a command line argument
        }
}
```

Output:

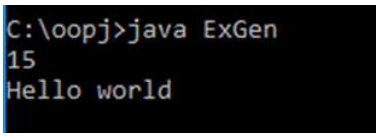


39) Create a JAVA program to implement generic class.

```
class Test<T>                    // generic class
{
          T obj;
          Test(T obj) {  this.obj = obj;  }
          public T getObject()  { return this.obj; }
}
class ExGen
```

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

      Test <Integer> Obj1 =  new Test<Integer>(15);
      System.out.println( Obj1.getObject() );

       Test <String> Obj2 =  new Test<String>( " Hello world " );
      System.out.println( Obj2.getObject() );
   }
}
```

Output:



40) Steps to create simple package.

Step 1) Create directory with package name let say mypack in your current directory.

Step 2) Create a JAVA file MyClass.JAVA in mypack directory.

```
package mypack;          //package declaration
class Hello
{
        public void sayHello ( String nm )
        {
              System. out. println ( " hello from " + nm );
        }
}
public class MyClass
{
        public static void main ( String args[] )
        {
                Hello h1=new Hello();
                h1.sayHello( " Aryu " );
```

```
        }
    }
```

Step 3) Compile MyClass.java file using “javac mypack/MyClass.java”

This will compile MyClass. Java file and create a MyClass.class and Hello.class files in the directory mypack.

Step 4) Run program

```
C:\oopj>java mypack.MyClass
 hello from  Aryu
```

41) Create a package name fruit. In this package create classes named Apple, Grape and Banana with appropriate attributes and methods in them. Compile the java files and create the classes in package fruit. Now prepare jar file containing this package. Put this jar file in class path. Create a java program outside this package which is using this package by importing it. Also create object of each class and call methods in main method. Use appropriate access modifier while creating classes.

// Apple.java file in package fruit

```
package fruit;                          //Package Declaration
public class Apple
{
        int id;
        String color;
        String shape;
        public Apple()
        {
                id = 0;
                color = "";
                shape = "";
        }
        public Apple(int i, String c, String s)
        {
                id = i;
```

```
                color = c;
                shape = s;
        }
        public void printApple()
        {
                System. out. println (" ID : " + id);
                System. out. println (" Color : " + color);
                System. out. println (" Shape : " + shape);
        }

}
```

// Grape.java file in package fruit

```
package fruit;                                  //Package Declaration
public class Grape
{
        int id;
        String color;
        int size;
        public Grape ()
        {
                id = 0;
                color = "";
                size = 0;
        }
        public Grape (int i, String c, int s)
        {
                id = i;
                color = c;
                size = s;
        }
        public void printGrape()
        {
                System. out. println (" ID : " + id);
                System. out. println (" Color : " + color);
```

```
                System. out. println (" Size : " + size);
        }

}
```

// Banana.java file in package fruit

```
package fruit;                          //Package Declaration
public class Banana
{
        int id;
        String color;
        String unit;
        public Banana ()
        {
                id = 0;
                color = "";
                unit = "";
        }
        public Banana (int i, String c, String u)
        {
                id = i;
                color = c;
                unit = u;
        }
        public void printBanana()
        {
                System. out. println (" ID : " + id);
                System. out. println (" Color : " + color);
                System. out. println (" Unit : " + unit);
        }

}
```

Compile :

```
C:\oopj>javac -d . Banana.java

C:\oopj>javac -d . Grape.java

C:\oopj>javac -d . Apple.java

C:\oopj>dir fruit
 Volume in drive C is ACER
 Volume Serial Number is DA72-BF35

 Directory of C:\oopj\fruit

04/04/2019  04:42 PM    <DIR>          .
04/04/2019  04:42 PM    <DIR>          ..
04/04/2019  04:42 PM               953 Apple.class
04/04/2019  04:42 PM               954 Banana.class
04/04/2019  04:42 PM               933 Grape.class
               3 File(s)          2,840 bytes
               2 Dir(s)  237,423,165,440 bytes free
```

```
C:\oopj>jar cf fruit.jar fruit

C:\oopj>set CLASSPATH=%CLASSPATH%;c:\oopj\fruit.jar
```

//ExPack.java

import fruit.Apple; **\\import apple class**

import fruit.Grape; **\\import Grape class**

import fruit.Banana; **\\import Banana class**

```
public class ExPack
{
        public static void main ( String args [] )
        {
                Apple a = new Apple (1, "red", "round"); \\object of apple class
                Grape g = new Grape ( 2, "green", 55 );  \\object of Grape class
                \\object of Banana class
                Banana b = new Banana ( 3, "yellow", "dozon" );
                a. printApple ();
                g. printGrape ();
                b. printBanana ();
        }
}
```

Run:

```
C:\oopj>javac ExPack.java

C:\oopj>java ExPack
 ID : 1
 Color : red
 Shape : round
 ID : 2
 Color : green
 Size : 55
 ID : 3
 Color : yellow
 Unit : dozon
```

42) Write a JAVA program to print "Hello World " using **ArrayList.**

```
import java.util.*;
class ExAList
{
public static void main(String args[])
        {
                ArrayList al = new ArrayList();
                al.add("Hello");
                al.add("World");
                 Iterator itr = al.iterator();
                while(itr.hasNext())
                System. out. println (itr.next());
         }
 }
```

Output:

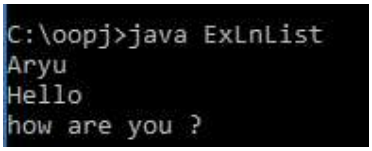
```
C:\oopj>java ExAList
Hello
World
```

43) Write a java program to implement simple LinkedList.

```
import java.util.*;
public class ExLnList
{
        public static void main(String args[])
        {
                LinkedList<String>  al = new LinkedList<String>();
```

```
                al.add("Aryu");
                al.add("Hello");
                al.add("how are you ?");
                Iterator<String>   itr = al.iterator();
                while(itr.hasNext())
                {
                System. out. println (itr.next());
                 }
        }
}
```

Output:

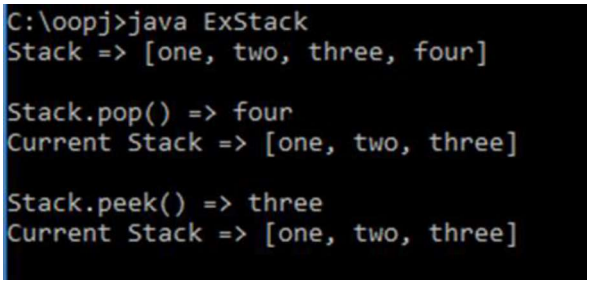


44) Write a JAVA program to implement stack.

```
import java.util.Stack;
public class ExStack
{
        public static void main(String[] args)
        {
                Stack<String> stk = new Stack<>();
                stk.push("one");
                stk.push("two");
                stk.push("three");
                stk.push("four");
                 System. out. println ("Stack => " + stk);
                 System. out. println ();
                 String tp = stk.pop();
                 System. out. println ("Stack.pop() => " + tp);
                 System. out. println ("Current Stack => " + stk);
                 System. out. println ();
                tp = stk.peek();
                System. out. println ("Stack.peek() => " + tp);
```

```
            System. out. println ("Current Stack => " + stk);
        }
}
```

Output:



45) Implement a JAVA program to show various operation of queue.

```
import java.util.*;
class ExQueue
{
    public static void main(String args[])
    {
        PriorityQueue<String> queue=new PriorityQueue<String>();
        queue.add("Hello");
        queue.add("World");
        queue.add("Aryu");
        System. out. println ("head:"+queue.element());
        System. out. println ("head:"+queue.peek());
        System. out. println ("iterating the queue elements:");
        Iterator itr=queue.iterator();
        while(itr.hasNext())
        {
            System. out. println (itr.next());
        }
        queue.remove();
        queue.poll();
        System. out. println ("after removing two elements:");
        Iterator<String> itr2=queue.iterator();
        while(itr2.hasNext())
```

```
            {
                System. out. println (itr2.next());
             }
        }
 }
```

Output:

```
C:\ajava\oopj>java ExQueue
head:Aryu
head:Aryu
iterating the queue elements:
Aryu
World
Hello
after removing two elements:
World
```

46) Write a JAVA program to implement HashSet.

```
import java.util.*;
class ExHSet
{
        public static void main ( String args [] )
        {
                HashSet <String>  al = new HashSet ();
                al.add( "Hello" );
                al.add( "World" );
                al.add( "Aryu" );
                Iterator <String>  itr = al.iterator ();
                 while ( itr.hasNext () )
                {
                        System. out. println  ( itr.next() );
                }
        }
}
```

Output:

```
C:\ajava\oopj>java ExHSet
Aryu
Hello
World
```

47) Write a JAVA program to implement LinkedHashSet.

```
import java.util.*;
public class LHSet
{
        public static void main ( String args[] )
        {
                LinkedHashSet <String> al = new LinkedHashSet();
                al.add ( "Hello" );
                al.add ( "World" );
                al.add ( "Aryu" );
                Iterator <String> itr = al.iterator ();
                While ( itr.hasNext() )
                {
                        System. out. println  ( itr.next() );
                }
        }
}
```

Output:

```
C:\oopj>java LHSet
Hello
World
Aryu
```

48) Write a JAVA program to implement TreeSet.

```
import java.util.*;
class ExTreeSet
 {
        public static void main( String args[] )
        {
                TreeSet <String> al = new TreeSet <String>  ();
```

```
                al.add ( "Hello" );
                al.add ( "World" );
                al.add ( "Aryu" );
                Iterator <String> itr = al.iterator();
                While ( itr.hasNext() )
                 {
                        System. out. println  ( itr.next ( ) );
                }
        }
}
```

Output:

```
C:\oopj>java ExTreeSet
Aryu
Hello
World
```

49) Write a JAVA program to implement Exception Handling.

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

public class ExErr
 {
        public static void main(String args[]) throws Exception
        {
                  int a = 0;
                  int b = 0;


                 try
                 {
                        //  Block of code to execute
                        Scanner sc = new Scanner ( System.in );
                        System. out. println  ( " a = ");
                        a = sc.nextInt();
                        System. out. println  ( " b = ");
                        b = sc.nextInt();
```

```
        }
        catch (InputMismatchException e )   \\Catch the error
        {
            //  Block of code to handle errors
            System. out. println  ( "Error occured as the value
                                    entered is a character ");
        }
        finally          \\ will execute at end
        {
            System. out. println  ( " a + b = " + ( a + b ));
        }
    }
}
```

Output:

```
C:\oopj>notepad ExErr.java

C:\oopj>javac ExErr.java

C:\oopj>java ExErr
 a =
3
 b =
4
 a + b = 7

C:\oopj>java ExErr
 a =
3
 b =
a
Error occured as the value entered is a character
 a + b = 3
```

50) Write a JAVA program to implement throw keyword to explicitly raise an exception.

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

public class ExErr
{
    public static void main(String args[]) throws Exception
    {
```

```
        int a[] = { 3, 4, 5, 6, 7, 8};
        int i;
        try
        {
                Scanner sc = new Scanner ( System.in );
                System. out. println  ( " index = ");
                i = sc.nextInt();
                if ( i >= 6)
                {
                    ArrayIndexOutOfBoundsException ex = new
                            ArrayIndexOutOfBoundsException();
                   Throw ex;        \\Throw exception
                }
                else
                        System. out. println  ( " a[i] = " +  a[i]);

        }
        catch (ArrayIndexOutOfBoundsException e )
        {
                System. out. println  ( "Error occured as the value of i is >=6 ");
        }
    }
}
```

Output:

```
C:\ajava\oopj>java ExErr
 index =
2
 a[i] = 5

C:\ajava\oopj>java ExErr
 index =
6
Error occured as the value of i is >=6
```

52) Write a JAVA program to implement user defined exception.

```
import java.util.Scanner;
class NegativeException extends Exception
 {
        private int x;
        NegativeException(int a)
        {
                x=a;
         }
         public String toString()
         {
                return "NegativeException[" + x +"] : value is less than zero";
         }
 }

public class UDException
{
        public static void main ( String args[])
         {
                int a;
                Scanner sc = new Scanner ( System.in );
                Try
                {
                        System. out. println ( "Enter a: ");
                        a = sc.nextInt();
                         if( a < 0 )
                                throw (new NegativeException(2) );
                        else
                                System. out. println  ( " a = " + a );
                 }
                 catch (Exception e) { System. out. println  (e); }
         }
}
```

51) Write a JAVA program to implement chained exception.

```
import java.io.*;
public class LinkedException
{
  static void raiseLinkedException()
  {
      ArithmeticException e = new ArithmeticException( " top most exception ");
      e.initCause( new IOException( " cause " ));
      throw e;
  }
  public static void main ( String args[] )
  {
          Try {raiseLinkedException();}
           catch ( ArithmeticException ex)
           {
                        System. out. println ( " caught  : " + ex );
                        System. out. println ( " cause  : " + ex.getCause() );
           }
}
}
```

Output:

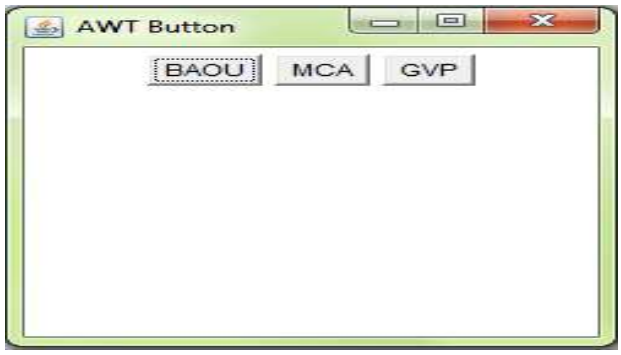
```
C:\oopj>javac LinkedException.java

C:\oopj>java LinkedException
 caught  : java.lang.ArithmeticException:  top most exception
 cause  : java.io.IOException:  cause
```

52) Write a JAVA program to working with Button Control in AWT.

```
import java.awt.*;
public class buttonTest extends Frame
{
          Button first, second, third;
          buttonTest(String str)
          {
                  super(str);
                  setLayout(new FlowLayout());      \\set layout of frame
```

```
                \\create button objects
                first = new Button("BAOU");
                second = new Button("MCA");
                third = new Button("GVP");
                // add Button to Frame
                add(first);
                add(second);
                add(third);
        }
        public static void main(String arg[])
        {
                Frame frm=new buttonTest("AWT Button");
                frm.setSize(250,250);
                frm.setVisible(true);
        }
}
```

Output:



53) Write a JAVA program to working with Label Control in AWT.

```
import java.awt.*;
public class labelTest extends Frame
{
        labelTest(String str)
        {
                super(str);
                setLayout(new FlowLayout());      \\set layout of frame
                \\create Label objects
```

```
                Label one = new Label("BAOU");
                Label two = new Label("MCA");
                Label three = new Label("GVP");
                // add labels to Frame
                add(one);
                add(two);
                add(three);
        }
        public static void main(String arg[])
         {
                Frame frm=new labelTest("AWT Label");
                frm.setSize(250,200);
                frm.setVisible(true);
        }
}
```

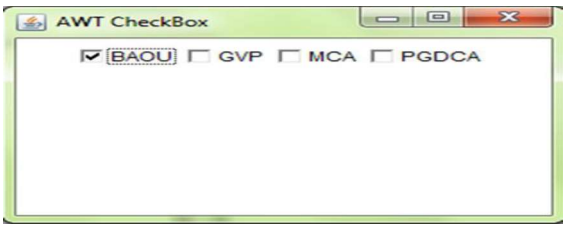
Output:

54) Write a JAVA program to working with Checkbox Control in AWT.

```
import java.awt.*;
public class checkBoxTest extends Frame
{
        Checkbox MCA, BCA, MscIT, Bsc;
        checkBoxTest(String str)
        {
                super(str);
                setLayout( new FlowLayout());     \\set layout of frame
```

```
                    \\create Checkbox objects
                    MCA = new Checkbox("BAOU", null, true);
                    BCA = new Checkbox("GVP");
                    MscIT = new Checkbox("MCA");
                    Bsc = new Checkbox("PGDCA");
                    // add Checkbox to Frame
                    add(MCA);
                    add(BCA);
                    add(MscIT);
                    add(Bsc);
            }
            public static void main(String arg[])
            {
                    Frame frm=new checkBoxTest("AWT CheckBox");
                    frm.setSize(300,200);
                    frm.setVisible(true);
            }
    }
```

Output:

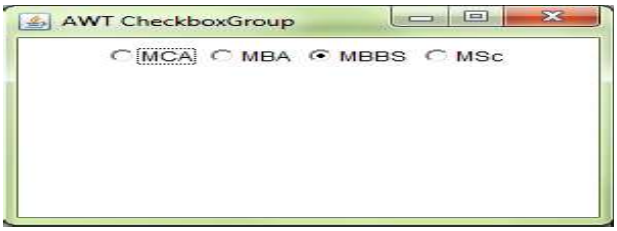


55) Write a JAVA program to working with CheckboxGroup Control in AWT.

```
import java.awt.*;
public class ChBoxGroup extends Frame
{
        Checkbox mca, mba, mbbs, msc;
        CheckboxGroup cbg;
        ChBoxGroup(String str)
        {
                super(str);
                setLayout(new FlowLayout());      \\set layout of frame
```

```
                cbg = new CheckboxGroup();      \\create object
                mca = new Checkbox("MCA", cbg, false);
                mba = new Checkbox("MBA", cbg, false);
                mbbs= new Checkbox("MBBS", cbg, true);
                msc = new Checkbox("MSc", cbg, false);
                add(mca);
                add(mba);
                add(mbbs);
                add(msc);
        }
        public static void main(String arg[])
        {
                Frame frm=new ChBoxGroup("AWT CheckboxGroup");
                 frm.setSize(300,200);
                 frm.setVisible(true);
        }
}
```

Output:

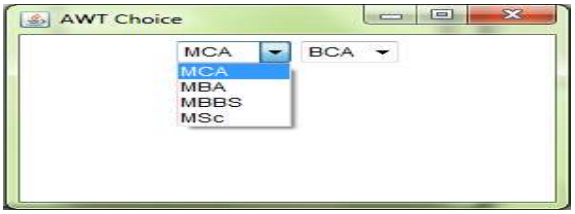


56) Write a java program to working with Choice Control in AWT.

```
import java.awt.*;
public class choiceTest extends Frame
{
        Choice master, bachelor;
        choiceTest(String str)
        {
                super(str);
                setLayout(new FlowLayout());      \\set layout of frame
                master = new Choice();
```

```
                bachelor = new Choice();
                master.add("MCA");
                master.add("MBA");
                master.add("MBBS");
                master.add("MSc");
                bachelor.add("BCA");
                bachelor.add("BBA");
                bachelor.add("BSc");
                add(master);
                add(bachelor);
        }
        public static void main(String arg[])
        {
                Frame frm=new choiceTest("AWT Choice");
                frm.setSize(300,200);
                frm.setVisible(true);
         }
}
```

Output:



57) Write a JAVA program to working with TextField Control in AWT.

```
import java.awt.*;
public class txtFieldTest extends Frame
{
        TextField txtname, txtpass;
        txtFieldTest(String str)
        {
                super(str);
                setLayout(new FlowLayout());      \\set layout of frame
                Label name = new Label("Name: ", Label.RIGHT);
```

```
                Label pass = new Label("Password: ", Label.RIGHT);
                txtname = new TextField(12);
                txtpass = new TextField(8);
                txtpass.setEchoChar('*');
                add(name);
                add(txtname);
                add(pass);
                add(txtpass);
        }
        public static void main(String arg[])
        {
                Frame frm=new txtFieldTest("AWT TextField");
                frm.setSize(250,200);
                frm.setVisible(true);
        }
}
```

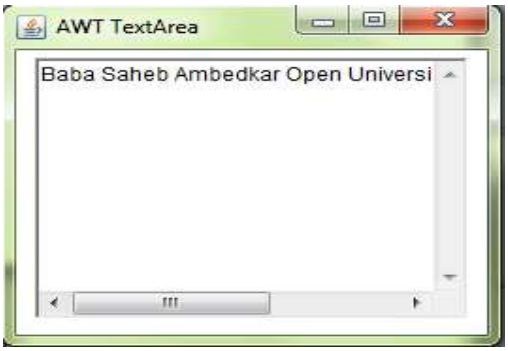
Output:

58) Write a JAVA program to working with TextArea Control in AWT.

```
import java.awt.*;
public class txtAreaTest extends Frame
{
        txtAreaTest(String str)
        {
                super(str);
                setLayout(new FlowLayout());
                String val ="Baba Saheb Ambedkar Open University”
                TextArea text = new TextArea(val, 10, 30);
```

```
            add(text);
        }
        public static void main(String arg[])
        {
            Frame frm=new txtAreaTest("AWT TextArea");
            frm.setSize(250,200);
             frm.setVisible(true);
         }
}
```

Output:

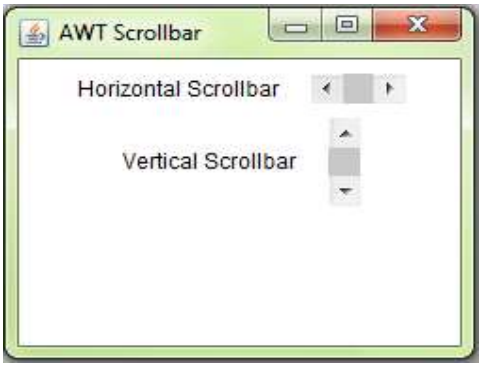


59) Write a JAVA program to working with Scrollbar Control in AWT.

```
import java.awt.*;
class scrollBarTest extends Frame
{
   scrollBarTest(String str)
   {
        super(str);
        setLayout(new FlowLayout());

        //Horizontal Scrollbar
        Label Horzlbl =new Label("Horizontal Scrollbar");
        Scrollbar hzsb = new  Scrollbar (Scrollbar.HORIZONTAL ,50,10,0,200);

        //Vertical Scrollbar
        Label vertlbl =new Label("Vertical Scrollbar");
        Scrollbar vtsb = new Scrollbar (Scrollbar.VERTICAL, 30,15,0,255);
```

```
            \\Add Scrollbars to frame
            add(Horzlbl);
            add(hzsb);
            add(vertlbl);
            add(vtsb);
    }
    public static void main(String arg[])
     {
                    Frame frm=new scrollBarTest("AWT Scrollbar");
                    frm.setSize(250,200);
                    frm.setVisible(true);
      }
}
```

Output:

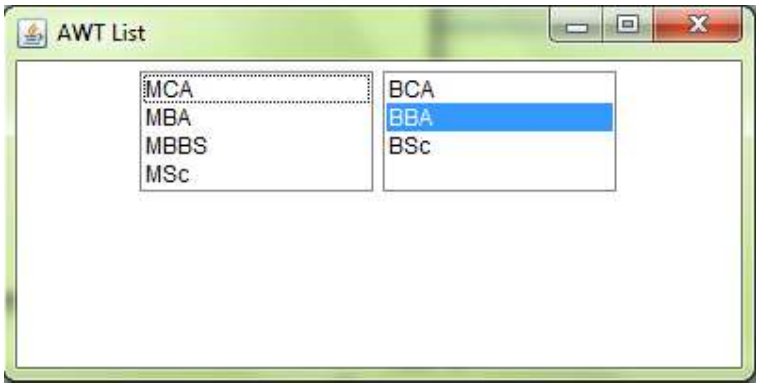


60) Write a JAVA program to working with List Control in AWT.

```
import java.awt.*;
public class ListTest extends Frame
{
        List master, bachelor;
        ListTest(String str)
        {
                super(str);
                setLayout(new FlowLayout());      \\set layout of frame
                master = new List(13, true);
                bachelor = new List(13, false);
```

```
                master.add("MCA");
                master.add("MBA");
                master.add("MBBS");
                master.add("MSc");
                bachelor.add("BCA");
                bachelor.add("BBA");
                bachelor.add("BSc");
                 bachelor.select(1);

                //add lists to Frame
                add(master);
                add(bachelor);
        }
        public static void main(String arg[])
        {
                Frame frm=new ListTest("AWT List");
                frm.setSize(1300,200);
                frm.setVisible(true);
         }
}
```

Output:

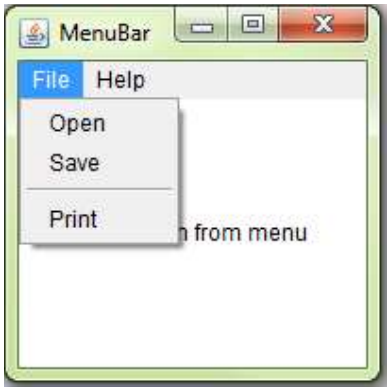


61) Write a JAVA program to working with Menu in AWT.

```
import java.awt.*;
import java.lang.*;
import java.util.*;

public class menuTest extends Frame
{
        MenuBar  mbar;
        Menu     file, help;
        MenuItem op, os, pr, sa, mc;
        Label   msg = new Label("Select an option from menu");
        menuTest(String str)
         {
                super(str);
                setLayout(new BorderLayout());    \\set layout of frame
                 add("Center", msg);
                mbar = new MenuBar();             \\create object of menubar

                mbar.add(file = new Menu("File"));              \\Add Menu File
                 mbar.add(help = new Menu("Help"));             \\AddMenu Help
                mbar.setHelpMenu(help);

                \\Add menu items in Menus
                file.add(op = new MenuItem("Open"));
                file.add(os = new MenuItem("Save"));
                file.addSeparator();
                file.add(pr = new MenuItem("Print"));

                help.add(sa = new MenuItem("Save As"));
                 help.add(mc = new MenuItem("close"));

                setMenuBar(mbar);                       \\set menubar
        }
          public static void main(String arg[])
          {
                  Frame frm=new menuTest("MenuBar");
                  frm.setSize(200,200);
```

```
                frm.setVisible(true);
            }
}
```

Output:

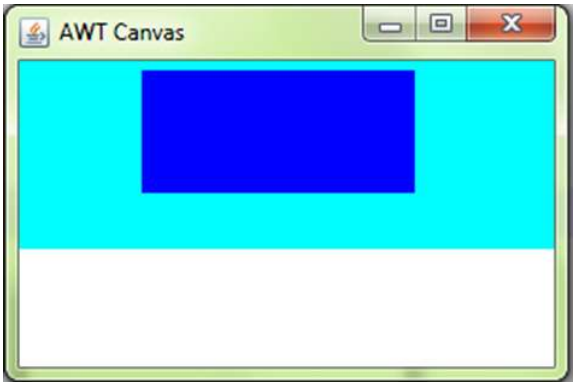


62) Write a JAVA program to working with Canvas in AWT.

```
import java.awt.*;
public class canvasDraw extends Frame
{
        public canvasDraw(String str)
        {
                super(str);
                CanvasTest ct = new CanvasTest();          \\ Object
                ct.setSize(125, 100);
                ct.setBackground(Color.cyan);
                add(ct, "North");
                setSize(300, 200);
                setVisible(true);
        }
        public static void main(String args[])
        {
                new canvasDraw("AWT Canvas");
        }
}
class CanvasTest extends Canvas
{
```

```
        public void paint(Graphics g)
        {
                g.setColor(Color.blue);
                 g.fillRect(65, 5, 1135, 65);
        }
}
```

Output:



63) Write a JAVA program to working with Panel in AWT.

```
import java.awt.*;
public class PanelTest extends Frame
{
        public PanelTest(String str)
        {
                super(str);
                setLayout(new BorderLayout());

                \\Create Panel
                Panel p1 = new Panel();
                Panel p2 = new Panel();

                p1.setBackground(Color.cyan);
                p2.setLayout(new GridLayout(1, 3, 20, 0));

               Button b1 = new Button("BAOU");
                Button b2 = new Button("GVP");
                Button b3 = new Button("MCA");
```

```
                Button b13 = new Button("BCA");
                Button b5 = new Button("MBA");
                Button b6 = new Button("BBA");

                p1.add(b1);
                p1.add(b2);
                 p1.add(b3);
                p2.add(b13);
                p2.add(b5);
                p2.add(b6);

                add(p1, "North");
                add(p2, "South");
        }
        public static void main(String args[])
        {
                Frame fm=new PanelTest("AWT Panel");
                fm.setSize(300, 200);
                fm.setVisible(true);
        }
}
```

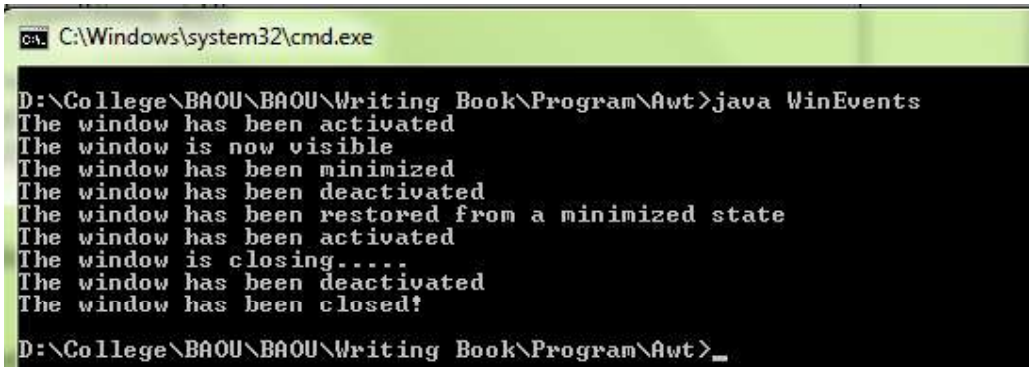
Output:

64) Write a program to implement all the window event methods.

```
import java.awt.*;
import java.awt.event.*;
public class WinEvents extends Frame implements WindowListener
{
        public WinEvents(String str)
        {
                super(str);
                addWindowListener(this);
        }
        public static void main(String[] args)
        {
                Frame fm = new WinEvents("WindowEvent_Example");
                fm.setSize(250, 250);
                 fm.setVisible(true);
        }
        public void windowClosing(WindowEvent we)
        {
                 System.out.println("The window is closing.....");
                ((Window)we.getSource()).dispose();
         }
        public void windowClosed(WindowEvent we)
        {
                System.out.println("The window has been closed!");
                System.exit(0);
         }
        public void windowActivated(WindowEvent we)
        {
                System.out.println("The window has been activated");
         }
        public void windowDeactivated(WindowEvent we)
        {
                 System.out.println("The window has been deactivated");
        }
        public void windowDeiconified(WindowEvent we)
```

```
        {
        System.out.println("The window has been restored from a minimized state");
         }
        public void windowIconified(WindowEvent we)
        {
                System.out.println("The window has been minimized");
         }
        public void windowOpened(WindowEvent we)
        {
                System.out.println("The window is now visible");
         }
}
```

Output:



65) Write a program to implement all the key event methods.

```
import java.awt.BorderLayout;
import java.awt.event.KeyEvent;
import java.awt.event.KeyListener;
import java.awt.Frame;
import java.awt.TextField;
import java.awt.TextArea;
import java.awt.Label;
public class keyListenerTest extends Frame implements KeyListener
{
        TextArea text;
```

```
TextField txtF;
Label l1;
keyListenerTest(String str)
 {
        super(str);
        setLayout(null);
        l1=new Label("Enter Key:");
        l1.setBounds(50,50,100,30);
        txtF= new TextField();
        text = new TextArea();
        txtF.addKeyListener(this);
        txtF.setBounds(160,50,100,30);
        text.setBounds(20,100,300,300);
        add(l1);
        add(txtF);
        add(text);
 }
public void keyPressed(KeyEvent ke)
{
        text.append("Key is Pressed\n");
}

public void keyReleased(KeyEvent ke)
{
        text.append("Key is Released\n");
}
public void keyTyped(KeyEvent ke)
{
        text.append("Key is Typed\n");
}
public static void main(String args[])
{
          Frame frame = new keyListenerTest("KeyListener");
          frame.setSize(350,1400);
```

```
                frame.setVisible(true);
        }
}
```

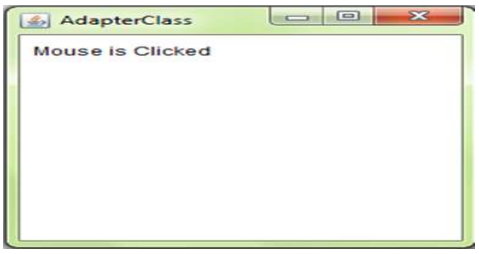
Output:

66) Write a program to implement adapter classes.

```
import java.awt.*;
import java.awt.event.*;
public class adapterTest extends Frame
{
        Label lblTest;
        adapterTest(String str)
        {
                super(str);
                setLayout(new FlowLayout(FlowLayout.LEFT));
                lblTest = new Label();
                add(lblTest);
                addMouseListener(new MyAdapter(lblTest));
        }
        public static void main(String str[])
        {
                Frame at=new adapterTest("AdapterClass");
                at.setSize(250,250);
                at.setVisible(true);
```

```
        }
}
class MyAdapter extends MouseAdapter
{
        Label lblTest;
        MyAdapter(Label lbl)
        {
                lblTest = lbl;
        }
        public void mouseClicked(MouseEvent me)
        {
                lblTest.setText("Mouse is Clicked");
        }
}
```

Output:

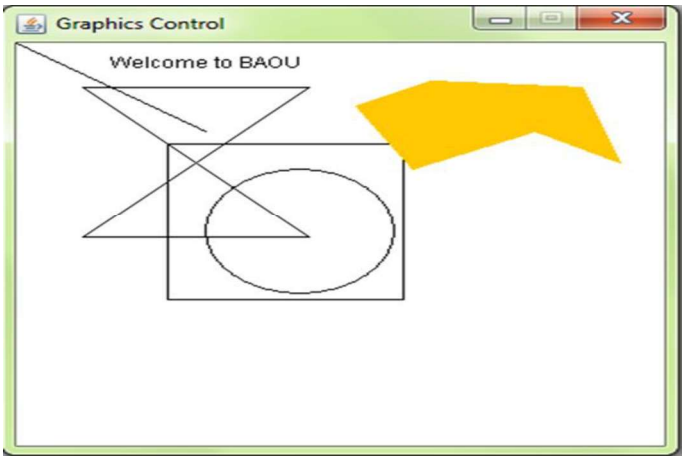


67) Write a JAVA program to implements methods of the Graphics class.

```
import java.awt.Frame;
import java.awt.Panel;
import java.awt.Graphics;
import java.awt.Polygon;
import java.awt.Color;
public class PictureDraw extends Panel
{
        public void paint(Graphics g)
        {
                //Print a String message
                g.drawString("Welcome to BAOU", 20, 20);
                //draw a Line
```

```
                g.drawLine(0, 0, 100, 70);
                //draw a Oval
                g.drawOval(100, 100, 100, 100);
                //draw a rectangle
                g.drawRect(80, 80, 125, 125);
                //draw a Polygon
                int x[] = {35, 155, 35, 155, 35};
                int y[] = {35, 35, 155, 155, 35};
                g.drawPolygon(x,y,5);   //points = 5

                g.setColor(Color.orange);
                Polygon pg = new Polygon();
                pg.addPoint(220, 30);
                pg.addPoint(300, 35);
                pg.addPoint(320, 95);
                pg.addPoint(275, 70);
                pg.addPoint(210, 100);
                pg.addPoint(180, 50);
                g.drawPolygon(pg);
                g.fillPolygon(pg);
        }
        public static void main(String[] args)
        {
                Frame f= new Frame("Graphics Control");
                f.add(new PictureDraw());
                f.setSize(600, 1500);
                f.setVisible(true);
                f.setResizable(false);
        }
}
```

Output:

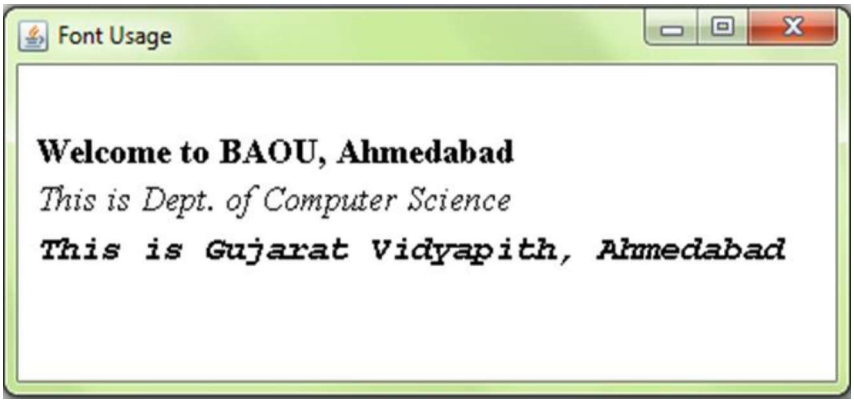


68) Write a java program to implement Font class and methods.

```
import java.awt.Font;
import java.awt.Frame;
import java.awt.Panel;
import java.awt.Graphics;
public class FontClass extends Panel
{
   public void paint(Graphics g)
         {
            Font f = new Font("Arial", Font.PLAIN, 18);
            Font fb = new Font("TimesRoman", Font.BOLD, 18);
            Font fi = new Font("Serif", Font.ITALIC, 18);
            Font fbi = new Font("Monospaced", Font.BOLD + Font.ITALIC, 18);
            g.setFont(f);
            g.setFont(fb);
            g.drawString("Welcome to BAOU, Ahmedabad", 10, 50);
            g.setFont(fi);
            g.drawString("This is Dept. of Computer Science", 10, 75);
            g.setFont(fbi);
            g.drawString("This is Gujarat Vidyapith, Ahmedabad", 10, 100);
         }
         public static void main(String s[])
         {
               Frame f= new Frame("Font Usage");
```

```
            f.add(new FontClass());
            f.setVisible(true);
            f.setSize(1550,200);
        }
}
```

Output:

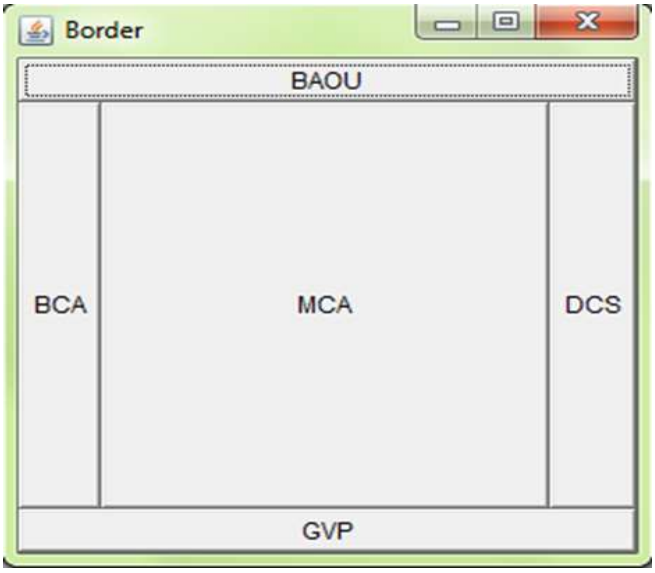


69) Write a java program depicts the use of Border Layout.

```
import java.awt.*;
public class BorderLout extends Frame
{
        BorderLout(String title)
        {
            super(title);
            Button b1=new Button("BAOU");;
            Button b2=new Button("GVP");;
            Button b3=new Button("DCS");;
            Button b15=new Button("BCA");;
            Button b5=new Button("MCA");;

            add(b1,BorderLayout.NORTH);
            add(b2,BorderLayout.SOUTH);
            add(b3,BorderLayout.EAST);
            add(b15,BorderLayout.WEST);
            add(b5,BorderLayout.CENTER);
        }
```

```
        public static void main(String[] args)
        {
                Frame bly=new BorderLout("Border");
                bly.setSize(300,300);
                bly.setVisible(true);
        }
}
```

Output:

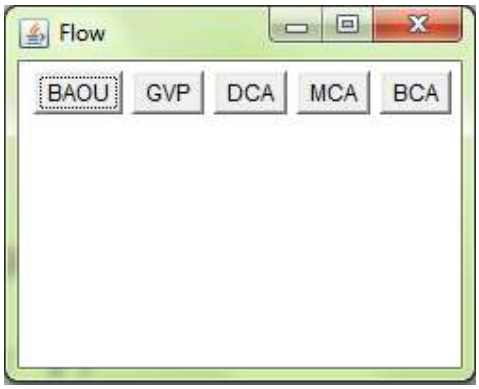


70) Write a java program depicts the use of Flow Layout.

```
import java.awt.*;
public class FlowLout extends Frame
{
        FlowLout(String title)
        {
                super(title);
                Button b1=new Button("BAOU");
                Button b2=new Button("GVP");
                Button b3=new Button("DCA");
                Button b15=new Button("MCA");
                Button b5=new Button("BCA");
```

```
                add(b1);add(b2);add(b3);add(b15);add(b5);
                //setting flow layout of right alignment
                setLayout(new FlowLayout(FlowLayout.RIGHT));
        }
        public static void main(String[] args)
        {
            Frame fly=new FlowLout("Flow");
            fly.setSize(250,200);
            fly.setVisible(true);
        }
}
```

Output:

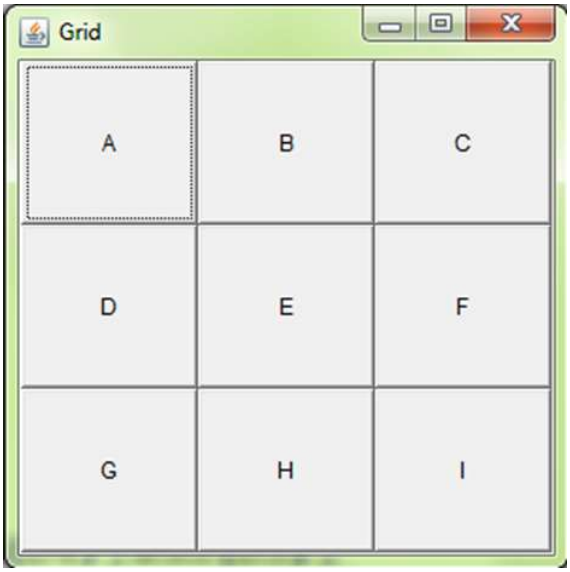


71) Write a JAVA program depicts the use of Grid Layout.

```
import java.awt.*;
public class GridLout extends Frame
{
        GridLout(String title)
        {
                super(title);
                Button ba=new Button("A");
                Button bb=new Button("B");
                Button bc=new Button("C");
                Button bd=new Button("D");
                Button be=new Button("E");
                Button bf=new Button("F");
```

```
                Button bg=new Button("G");
                Button bh=new Button("H");
                Button bi=new Button("I");

                add(ba);add(bb);
                add(bc);add(bd);
                add(be);add(bf);
                add(bg);add(bh);add(bi);
                 //setting gridlayout of 3 rows and 3 columns
                setLayout(new GridLayout(3,3));
        }
        public static void main(String[] args)
        {
                        Frame fyl=new GridLout("Grid");
                        fyl.setSize(300,300);
                        fyl.setVisible(true);
        }
}
```

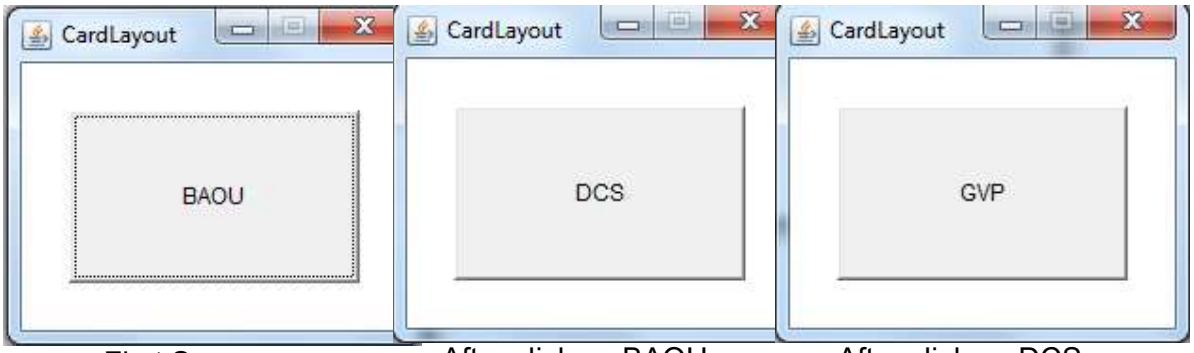
Output:



72) Write a JAVA program depicts the use of Card Layout.

```
Import java.awt.*;
import java.awt.event.*;
class CardLout extends Frame implements ActionListener
{
    CardLayout cardlt = new CardLayout(25,25);
    CardLout(String str)
    {
                super(str);
                setLayout(cardlt);
                Button Panel1 = new Button("BAOU");
                Button Panel2 = new Button ("DCS");
                Button Panel3 = new Button("GVP");
                add(Panel1,"BAOU");
                add(Panel2,"DCS");
                add(Panel3,"GVP");
                Panel1.addActionListener(this);
                Panel2.addActionListener (this);
                Panel3.addActionListener(this);
    }
     public void actionPerformed(ActionEvent e)
     {
                  cardlt.next(this);
     }

     public static void main(String args[])
     {
          CardLout frame = new CardLout("CardLayout");
          frame.setSize(210,170);
          frame.setResizable(false);
          frame.setVisible(true);
     }
}
```

Output:



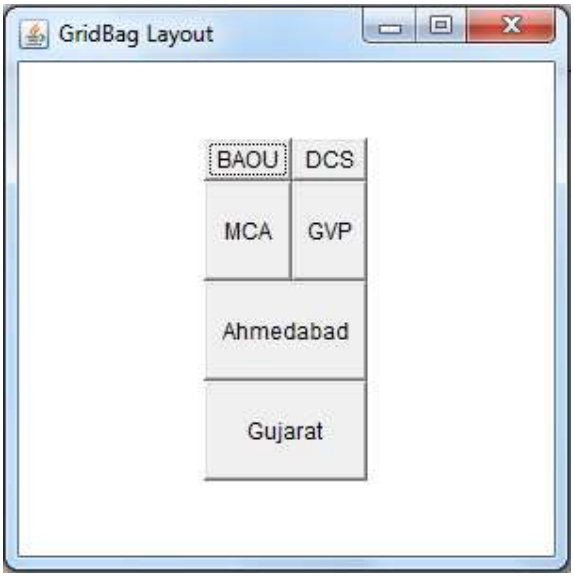
First Screen | After click on BAOU | After click on DCS

73) Write a JAVA program depicts the use of GridBag Layout.

```
import java.awt.*;
import java.awt.GridBagConstraints;
import java.awt.GridBagLayout;

public class gridBagLout extends Frame
{
        Button first, second,third,forth,fifth,sixth;
        public static void main(String[] args)
        {
                 Frame gbl = new gridBagLout(“GridBag Layout”);
                gbl.setSize(300, 300);
                gbl.setVisible(true);
        }
        public gridBagLout(String str)
        {
                super(str);
                first=new Button("BAOU");
                second=new Button("DCS");
                third=new Button("MCA");
                forth=new Button("GVP");
                fifth=new Button("Ahmedabad");
                sixth=new Button("Gujarat");
                GridBagConstraints gbc = new GridBagConstraints();
                GridBagLayout layout = new GridBagLayout();
```

```
        setLayout(layout);

        gbc.fill = GridBagConstraints.HORIZONTAL;
        gbc.gridx = 0;
        gbc.gridy = 0;
        add(first, gbc);
        gbc.gridx = 1;
        gbc.gridy = 0;
        add(second, gbc);
        gbc.fill = GridBagConstraints.HORIZONTAL;
         gbc.ipady = 30;
        gbc.gridx = 0;
        gbc.gridy = 1;
        add(third, gbc);
        gbc.gridx = 1;
        gbc.gridy = 1;
        add(forth, gbc);
        gbc.gridx = 0;
        gbc.gridy = 2;
        gbc.fill = GridBagConstraints.HORIZONTAL;
        gbc.gridwidth = 2;  //Merge two columns
        add(fifth, gbc);
         gbc.gridx = 0;
        gbc.gridy = 3;
        gbc.gridwidth = 2;  //Merge two columns
        add(sixth, gbc);
    }
}
```

Output:

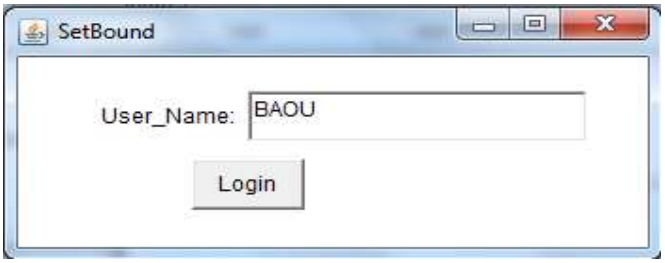


74) Write a JAVA program depicts the use of Setbound().

```
import java.awt.*;
public class Setbound extends Frame
{
        Label name;
        TextField user;
        Button login;
        Setbound(String str)
        {
                super(str);
                setLayout(null);
                name=new Label("User_Name:");
                user=new TextField(10);
                login=new Button("Login");

                name.setBounds(50, 50, 75, 30 );
                add(name);
                user.setBounds(130, 50, 180,30 );
                add(user);
                login.setBounds(100, 90, 60, 30 );
                add(login);
         }
```

```
        public static void main(String[] args)
        {
                Frame sb=new Setbound("SetBound");
                sb.setSize(350,150);
                sb.setVisible(true);
         }
}
```

Output:



75) Write a JAVA Program to write a String and character array using OutputStreamWriter and reading back the same file using InputStreamReader.

```
import java.io.*;
public class Outstreamwriter
{
        public static void main(String[ ] arg)
        {
                String str=" BAOU";
                char[] arrdata= {'V','I','D','Y','A','P','I','T','H'};
                try
                {
                        FileOutputStream fos= new FileOutputStream("Test1.txt");
                        OutputStreamWriter osw= new OutputStreamWriter(fos);
                        // writing each character of character array
                        for(char ch : arrdata)
                        {
                                osw.write(ch);
                        }
```

```
                    //writing a String
                    osw.write(str);
                    osw.flush();
                    osw.close();
                }
                catch(IOException e)
                {
                    System.out.println(e);
                }
                try
                {
                    FileInputStream fis= new FileInputStream("Test1.txt");
                    InputStreamReader isr= new InputStreamReader(fis);
                    int data;
                        while( (data=isr.read())!=-1)
                        {
                            System.out.print((char)data);
                        }
                }
                catch(IOException e){System.out.println(e);}
            }
}
```

76) Write a JAVA Program to writes a data in a file using BufferedWriter and reads the content back from the same file using BufferedReader.

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;

public class buferReadWriter
{
```

```
public static void main(String[] args)
{
        File buffile = new File("buff.txt");

        /*Writing file using BufferedWriter*/
        FileWriter filewrite = null;
        BufferedWriter buffwrite =null;
        try
        {
                filewrite=new FileWriter(buffile);
                buffwrite =new BufferedWriter(filewrite);
                buffwrite.write("Babasaheb Ambedkar Open University \n");
                buffwrite.write("Gujarat Vidyapith \n");
                buffwrite.write("Dept. of Computer Science");
                buffwrite.flush();
        }
        catch (IOException ioe)
        {
                System.out.println(ioe);
        }
        finally {
                try
                 {
                        if(filewrite!=null)
                        {
                                filewrite.close();
                        }
                        if(buffwrite!=null)
                        {
                                buffwrite.close();
                        }
                }
                catch (IOException ioe)
                {
```

```
                System.out.println(ioe);
        }
}
```

/*Reading file using BufferedReader*/

```
FileReader fileread=null;
BufferedReader buffRead=null;
try {
        fileread =new FileReader(buffile);
        buffRead=new BufferedReader(fileread);
        String data=null;
        while((data=buffRead.readLine())!=null)
        {
                System.out.println(data);
        }
}
catch (IOException ioe)
{
        System.out.println(ioe);
}
finally {
        try {
                if(fileread!=null)
                {
                        fileread.close();
                }
                if(buffRead!=null)
                {
                        buffRead.close();
                }
        }
        catch (IOException ioe)
        {
                System.out.println(ioe);
```

```
                    }
                }
            }
}
```

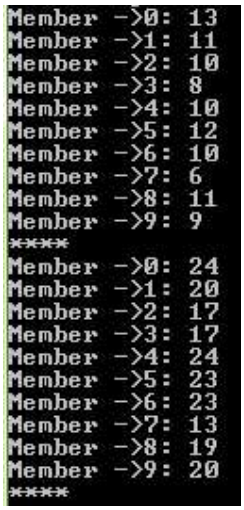
77) Write a JAVA program in which one thread takes off the behavior of scores for 10 members by generating random numbers between 0 to 10. Another thread keeps track of the total votes per members using PipedWriter and PipedReader class.

```
import java.util.*;
import java.io.*;
public class PipeThreadRdWtr
{
        public static void main(String[] args) throws Exception
        {
                PipedWriter owner = new PipedWriter();
                PipedReader user = new PipedReader(owner);
                DigitOwner dit = new DigitOwner(owner);
                DigitUser du = new DigitUser(user);
                 dit.start();
                du.start();
         }
}
class DigitOwner extends Thread
{
        BufferedWriter bw;
        public DigitOwner(Writer w)
         {
                 this.bw = new BufferedWriter(w);
         }
         // thread continually generates random votes
         public void run()
        {
```

```
                try
                {
                        Random r = new Random();
                        while (true)
                        {
                                String vote = "" + Math.abs((r.nextInt() % 10));
                                bw.write(vote);
                                bw.newLine();
                                bw.flush();
                                sleep(20);
                         }
                }
                catch(IOException e)
                {
                                System.err.println(e);
                }
                catch(InterruptedException e)
                {
                                System.err.println(e);
                 }
        }
}
class DigitUser extends Thread
{
        BufferedReader br;
        int[] votes = new int[10];
        public DigitUser(Reader r)
        {
                br = new BufferedReader(r);
         }
        public void run()
        {
                try
                {
```

```
                String data;
                 int count = 0;
                while ((data = br.readLine()) != null)
                {
                        int member = Integer.parseInt(data);
                        votes[member]++;
                        count++;
                        if (count % 100 == 0)
                         {
                                for (int i=0; i<votes.length; i++)
                                {

                                    System.out.println("Member ->" + i + ": "
                                    + votes[i]);
                                }
                                System.out.println("****");
                        }
                 }
              }
              catch(IOException e)
               {
                    System.err.println(e);
              }
        }
}
```

Output:



78) Write a java program to write to and read from the file using FileInputStream and FileOutputStream .

```
import java.io.*;
class fileInoutStream
{
        public static void main(String args[])
        {
                FileInputStream fin;
                BufferedReader br = null;
                try
                {
                        //Writing in to file
                        FileOutputStream fout= new FileOutputStream("foutest.txt");
                        fout.write(50);
                        fout.write('V');
                        fout.write('D');
                        fout.close();
                        //Reading from the file
                        fin=new FileInputStream("foutest.txt");
                        BufferedInputStream bin=new BufferedInputStream (fin);
                        int i;
                        while((i=bin.read())!=-1)
                        {
```

```
                    System.out.print((char)i);
                }
                bin.close();
                fin.close();
            }
            catch(Exception ex)
            {
                System.out.println(ex);
            }
        }
}
```

79) Write a JAVA program to write to and read from the file using DataOutputStream and DatainputStream.

```
import java.io.*;
public class DataInOut
{
   public static void main(String[] args) throws IOException
        {
            //Writing to the file
            FileOutputStream datafile = new FileOutputStream("dataout.txt");
            DataOutputStream data = new DataOutputStream(datafile);
            data.write(50);
            data.write('V');
            data.write('L');
            data.write('D');
            data.flush();
            data.close();

            //Reading from the file
            InputStream inputdata = new FileInputStream("dataout.txt");
            DataInputStream datainst = new DataInputStream(inputdata);
            int count = inputdata.available();
            byte[] arydata = new byte[count];
```

```
                datainst.read(arydata);
                for (byte vd : arydata)
                {
                        char ch = (char) vd;
                        System.out.print("->"+ch);
                }
        }
}
```

80) Write a JAVA program to implement File handling using File class.

```
import java.io.File;
import java.io.*;
public class Filehandling
{
        public static void main(String[] args)
        {
                File f1 = new File("D:\\College\\BAOU\\BAOU\\Writing
                                        Book\\Program\\Book\\Test.txt");
                System.out.println("Folder Name is       : "+f1.getName());
                System.out.println("Full Path  is        : "+f1.getPath());
                System.out.println("Parent of  file      : "+f1.getParent());
                System.out.println("Book Folder is    : "+f1.exists());
                System.out.println("Book is a File    : "+f1.isFile());

                System.out.println("Test.txt is writeable   : "+f1.canWrite());
                System.out.println("Test.txt is readable    : "+f1.canRead());
                System.out.println("Test.txt size in Bytes  : "+f1.length());
                System.out.println("Absolute Location is   : "+f1.toString());
                System.out.println("Test.txt is Hidden file     : "+f1.isHidden());

                //Creating a new Directory
                File f2 = new File("D:\\College\\BAOU\\BAOU\\Writing
                                        Book\\Program\\newDir");
                if(f2.mkdir())
                {
```

```
                System.out.println("Directory Created : Success");
            }
            else
            {
                System.out.println("Directory Created : Unsuccess");
            }
            //New file creation
            File f3 = new File("D:\\College\\BAOU\\BAOU\\Writing
                                Book\\Program\\new.txt");
            try
            {
                if(f3.createNewFile())
                {
                    System.out.println("File Created : Success");
                }
                else
                {
                    System.out.println("File Created : Unsuccess");
                }
            }
            catch (IOException io){}
        }
    }
```

81) Write a java program to implement console class.

```
import java.io.*;
import java.util.*;
public class ConsoleReadWrite
{
    public static void main(String[] args) throws IOException
    {
```

```
                Console console = System.console();
                 if (console == null)
                {
                        System.out.println("Console is not supported");
                        System.exit(1);
                 }
                String Stuname = console.readLine("student name: ");
                String Stuage = console.readLine("student Age: ");
                String Stucity = console.readLine("student Address? ");
                //console.format to print Data
                console.printf("%s, a %s year-old student is living in %s",
                                Stuname, Stuage, Stucity);
         }
}
```

82) Write a JAVA program which prints even numbers after every one second interval using thread class.

```
class EvenThread extends Thread
{
        EvenThread(String name)
        {
                super(name);
        }
        public void run()
        {
              for(int i=1; i&lt;11; i++)
              {
                        if(i%2==0)
                                System.out.println(this.getName() + &quot; :&quot; + i);
                        try
                        {
                                Thread.sleep(1000);
                        }
                        catch (InterruptedException e)
```

```
                    {
                        System.out.println ("Thread is Interrupted");
                    }
                }
            }
        }

        class ThreadDemoOne
        {
            public static void main(String [] args)
            {
                EvenThread et1 = new EvenThread("Thread 1 :");
                et1.start();
                EvenThread et2 = new EvenThread("Thread 2 :");
                et2.start();
                while(et1.isAlive() || et2.isAlive()){}
            }
        }
```

83) Write a JAVA program to implement Runnable interface.

```
        class EvenRunnable implements Runnable
        {
            String name="";
            EvenRunnable (String name)
            {
                this.name = name;
            }
            public void run()
            {
                for(int i=1; i&lt;11; i++)
                {
                    if(i%2==0)
                        System.out.println(Thread.currentThread().getName() + i);
                    try
                    {
```

```
                    Thread.sleep(1000);
                }
                catch (InterruptedException e)
                {
                    System.out.println ("Thread is interrupted");
                }
            }
        }
}

class ThreadDemoTwo
{
    public static void main(String [] args)
    {
        EvenThread et1 = new EvenThread("Thread 1 :");
        Thread t1 = new Thread(et1);
        t1.start();
        EvenThread et2 = new EvenThread("Thread 2 :");
        Thread t2 = new Thread(et2);
        t2.start();
        while(t1.isAlive() || t2.isAlive()){}
    }
}
```

84) Write a JAVA program to set priority of thread.

```
class ThreadPriorityDemo
{
  public static void main (String [] args)
  {
    try
    {
        Thread t1 = new Thread("Thread1");
        Thread t2 = new Thread("Thread2");
```

```
        System.out.println ("Before any change")
        System.out.println("The Priority of"+ t1.getName() +"is"+ t1.getPriority());
        System.out.println("The Priority of"+ t1.getName() +"is"+ t2.getPriority());
        //change in priority
       t1.setPriority(7);
       t2.setPriority(8);
       System.out.println ("After changing in Priority");
       System.out.println("The Priority of"+ t1.getName() +"is"+ t1.getPriority());
       System.out.println("The Priority of"+t1.getName() +" is"+ t2.getPriority());
    }
    catch (Exception e)
    {
       System.out.println("0main thread interrupted");
    }
  }
}
```

85) Write a JAVA program to control access of resource using wait() and notify () methods.

```
class WaitNotify implements Runnable
{
        WaitNotify ()
        {
                Thread th = new Thread (this);
                th.start();
        }
        synchronized void notifyThat ()          \\Synchronized Method
        {
                System.out.println (&amp;quot;Notify the threads waiting&amp;quot;);
                this.notify();
        }
        synchronized public void run()
        {
                try
```

```
                {
                        System.out.println("Thead is waiting....");
                        this.wait ();
                }
                catch (InterruptedException e){}
                System.out.println (&amp;quot;Waiting thread notified&amp;quot;);
          }
}
Class RunWaitNotify
{
        public static void main (String args[])
        {
                WaitNotify wait_not = new WaitNotify();
                Thread.yield ();
                wait_not.notifyThat();
        }
}
```

1.6 LIST OF LAB ASSIGNMENTS

1. Write a JAVA program to print number in reverse order.
2. Write a JAVA program to print first 10 Fibonacci number.
3. Write a JAVA program to Change UpperCase to LowerCase and Lowercase to Uppercase of All the Characters in the String without Using Built in Java function.
4. Write a JAVA program to find greatest common divisor (gcd) and least common multiple (LCM) of two integers.
5. Write a JAVA program using StringBuffer to delete, remove character from given string.
6. Create a class name employee which stores information like emp id, name, phone number, address, designation etc. Write a function which accepts an object of employee to add a new employee in existing list of employee. While adding check for emp id. The emp id should be of 5 character long. Implement this check using user define exception class.

7. Write a JAVA program to create a single ball bouncing inside a Panel using multithreading concept.
8. Write a JAVA program to display the digital watch.
9. Write a JAVA program that display the x and y position of the cursor movement using Mouse.
10. write a JAVA program that creates multiple threads using Runnable interface where First thread display “Hello“every 1 sec, the second thread displays “ Welcoming You In“ every 2 seconds and the third display “Dr. Babasaheb Ambedkar Open University” every 3 seconds.

1.7 FURTHER READINGS

1. https://www.guru99.com/install-java.html
2. https://www.javatpoint.com/java-tutorial

BAOU
Education for All

યુનિવર્સિટી ગીત

સ્વાધ્યાયઃ પરમં તપઃ
સ્વાધ્યાયઃ પરમં તપઃ
સ્વાધ્યાયઃ પરમં તપઃ

શિક્ષણ, સંસ્કૃતિ, સદ્‌ભાવ, દિવ્યબોધનું ધામ
ડૉ. બાબાસાહેબ આંબેડકર ઓપન યુનિવર્સિટી નામ;
સૌને સૌની પાંખ મળે, ને સૌને સૌનું આભ,
દશે દિશામાં સ્મિત વહે હો દશે દિશે શુભ-લાભ.

અભણ રહી અજ્ઞાનના શાને, અંધકારને પીવો ?
કહે બુદ્ધ આંબેડકર કહે, તું થા તારો દીવો;
શારદીય અજવાળા પહોંચ્યાં ગુર્જર ગામે ગામ
ધ્રુવ તારકની જેમ ઝળહળે એકલવ્યની શાન.

સરસ્વતીના મયૂર તમારે ફળિયે આવી ગહેકે
અંધકારને હડસેલીને ઉજાસના ફૂલ મહેંકે;
બંધન નહીં કો સ્થાન સમયના જવું ન ઘરથી દૂર
ઘર આવી મા હરે શારદા દૈન્ય તિમિરના પૂર.

સંસ્કારોની સુગંધ મહેંકે, મન મંદિરને ધામે
સુખની ટપાલ પહોંચે સૌને પોતાને સરનામે;
સમાજ કેરે દરિયે હાંકી શિક્ષણ કેરું વહાણ,
આવો કરીયે આપણ સૌ
ભવ્ય રાષ્ટ્ર નિર્માણ...
દિવ્ય રાષ્ટ્ર નિર્માણ...
ભવ્ય રાષ્ટ્ર નિર્માણ

9 788194 057734

DR. BABASAHEB AMBEDKAR OPEN UNIVERSITY
(Established by Government of Gujarat)
'Jyotirmay' Parisar,
Sarkhej-Gandhinagar Highway, Chharodi, Ahmedabad-382 481
Website : www.baou.edu.in