Java Programming Lab Manual

(GR-20)

Index

Sno	Task No	Page Nos
1	Task1	
2	Task2	
3	Task3	
4	Task4	
5	Task5	
6	Task6	
7	Task7	
8	Task8	
9	Task9	
10	Task10	
11	Task11	
12	Task12	
13	Task13	

Syllabus

Week-1

Write java programs that implement the following

a) Class and Object

b) Constructor

c) Parameterized constructor

d) Method overloading

e) Constructor overloading.

Week-2

a) Write a Java program that checks whether a given string is a palindrome or not.

b) Write a Java program for sorting a given list of names in ascending order.

c) Write a Java Program that reads a line of integers, and then displays each integer and the sum of all the integers (Use StringTokenizer class of java.util)

Week-3

Write java programs that uses the following keywords

a) this

b) super

c) static

d) final

Week-4

a) Write a java program to implement method overriding

b) Write a java program to implement dynamic method dispatch.

c) Write a Java program to implement multiple inheritance.

d) Write a java program that uses access specifiers.

Week-5

a) Write a Java program that reads a file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.

b) Write a Java program that reads a file and displays the file on the screen, with a line number before each line.

c) Write a Java program that displays the number of characters, lines and words in a text file.

Week-6

a) Write a Java program for handling Checked Exceptions.

b) Write a Java program for handling Unchecked Exceptions.

Week-7

a) Write a Java program that creates three threads. First thread displays “Good Morning” every one second, the second thread displays “Hello” every two seconds and the third thread displays “Welcome” every three seconds.

b) Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.

Week-8

Write a program illustrating following collections framework

a) ArrayList

b) Vector

c) HashTable

d) Stack

Week-9

a) Develop an applet that displays a simple message.

b) Develop an applet that receives an integer in one text field, and computes its factorial value and returns it in another text field, when the button named “Compute” is clicked.

c) Write a Java program that works as a simple calculator. Use a grid layout to arrange button for the digits and for the +, -,*, % operations. Add a text field to display the result.

Week-10

a) Write a Java program for handling mouse events.

b) Write a Java program for handling key events.

Week-11

a) Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields Num1 and Num 2.

b) The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1or Num2 were not an integer, the program would throw Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception and display the exception in a message dialog box.

Week-12

a) Write a java program that simulates traffic light. The program lets the user select one of three lights: red, yellow or green. When a radio button is selected, the light is turned on, and only one light can be on at a time No Light is on when the program starts.

b) Write a Java program that allows the user to draw lines, rectangles and ovals.

Week-13

Create a table in Table.txt file such that the first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using JTable component.

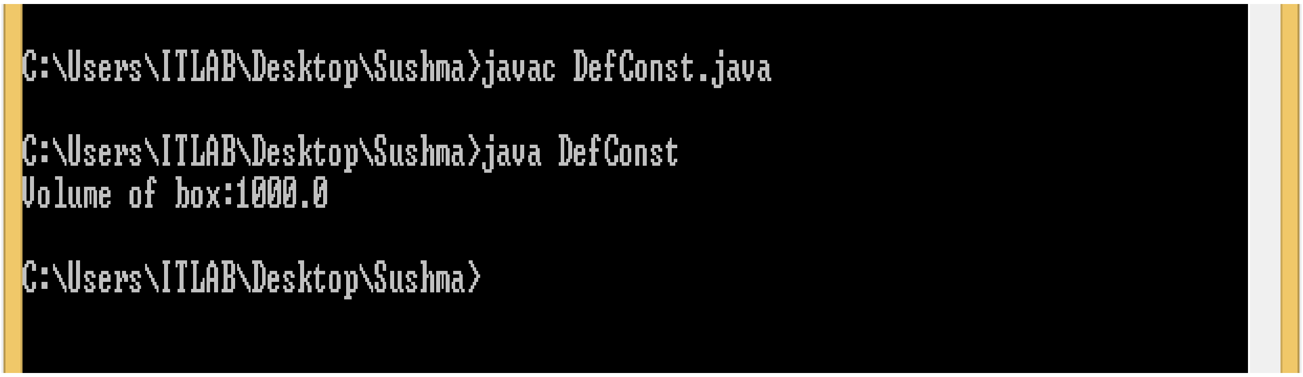
Week-1

Write java programs that implement the following

- **Constructor**

```
class Box
{
        double l;
        Box()
        {
                l=10;
        }
        double vol()
        {
                return l*l*l;
        }
}
class DefConst
{
        public static void main(String ar[])
        {
                Box b=new Box();
                double v=b.vol();
                System.out.println("Volume of box:"+v);
        }
}
```

Output

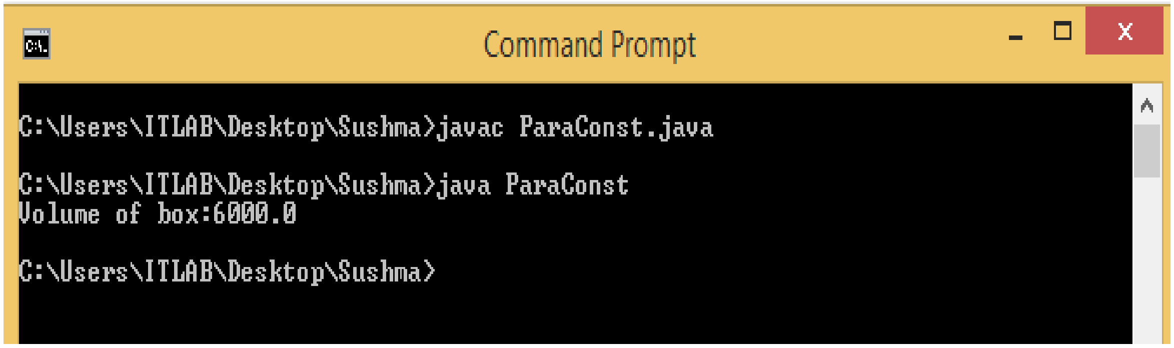


```
C:\Users\ITLAB\Desktop\Sushma>javac DefConst.java

C:\Users\ITLAB\Desktop\Sushma>java DefConst
Volume of box:1000.0

C:\Users\ITLAB\Desktop\Sushma>
```

- **Parameterized constructor**

```
class Box1
{
        double l,b,h;
        Box(double l, double b, double h)
        {
                this.l=l;
                this.b=b;
                this.h=h;
        }
        double vol()
        {
                return l*b*h;
        }
}
class ParaConst
{
        public static void main(String ar[])
        {
                Box1 b=new Box1(10,20,30);
                double v=b.vol();
                System.out.println("Volume of box:"+v);
        }
}
```

Output

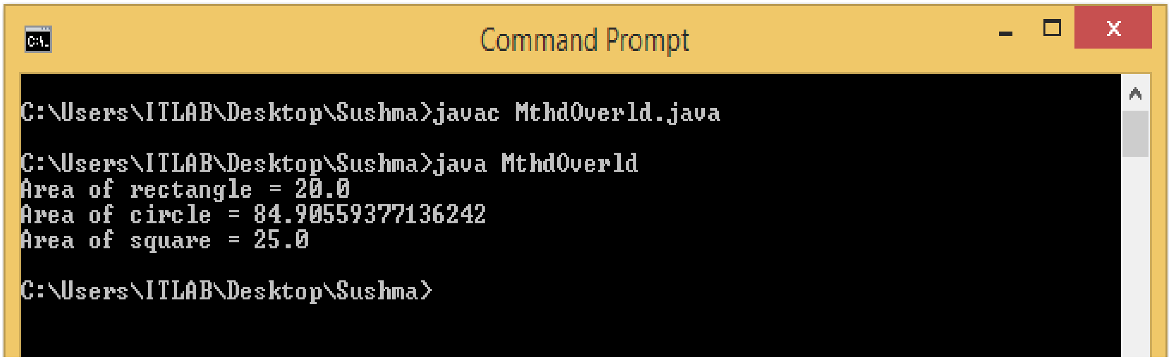


```
C:\Users\ITLAB\Desktop\Sushma>javac ParaConst.java

C:\Users\ITLAB\Desktop\Sushma>java ParaConst
Volume of box:6000.0

C:\Users\ITLAB\Desktop\Sushma>
```

- **Method overloading**

```
class Ar
{
        double l;
        double b;
        void Area(double x,double y)
        {
                l=x;
                b=y;
                System.out.println("Area of rectangle = "+(l*b));
        }
        void Area(double x)
        {
                l=x;
                System.out.println("Area of square = "+(l*l));
        }
        void Area(float x)
        {
                l=x;
                System.out.println("Area of circle = "+(l*l)*(3.14));
        }
 }
class MthdOvrld
{
        public static void main(String args[])
        {
                Ar a1=new Ar();
                a1.Area(4.0d,5.0d);
                a1.Area(5.2f);
                a1.Area(5.0d);
        }
}
```

Output

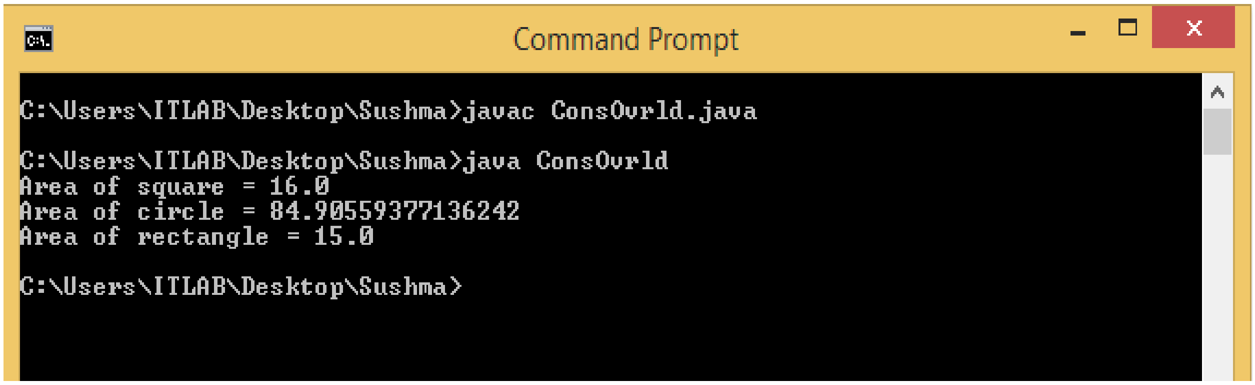


```
C:\Users\ITLAB\Desktop\Sushma>javac MthdOverld.java

C:\Users\ITLAB\Desktop\Sushma>java MthdOverld
Area of rectangle = 20.0
Area of circle = 84.90559377136242
Area of square = 25.0

C:\Users\ITLAB\Desktop\Sushma>
```

- **Constructor overloading**

```
class Area
{
        double l;
        double b;
        Area(double x,double y)
        {
                l=x;
                b=y;
                System.out.println("Area of rectangle = "+(l*b));
        }
        Area(double x)
        {
                l=x;
                System.out.println("Area of square = "+(l*l));
        }
        Area(float x)
        {
                l=x;
                System.out.println("Area of circle = "+(l*l)*(3.14));
        }
 }
class ConsOvrld
{
        public static void main(String args[])
        {
                Area r=new Area(4.0d);
                Area c=new Area(5.2f);
                Area s=new Area(5.0d,3.0d);
        }
}
```

Output



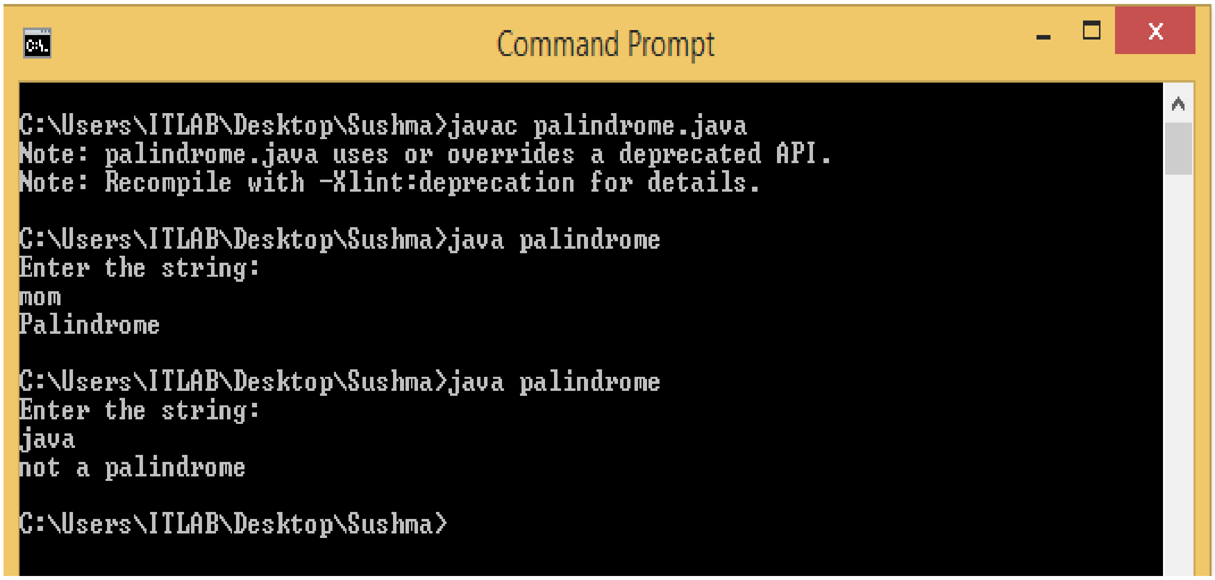
```
C:\Users\ITLAB\Desktop\Sushma>javac ConsOvrld.java

C:\Users\ITLAB\Desktop\Sushma>java ConsOvrld
Area of square = 16.0
Area of circle = 84.90559377136242
Area of rectangle = 15.0

C:\Users\ITLAB\Desktop\Sushma>
```

Week-2

- **Write a Java program that checks whether a given string is a palindrome or not.**

```
import java.io.*;
class palindrome
{
 public static void main(String args[]) throws Exception
{
  DataInputStream dis=new DataInputStream(System.in);
  System.out.println("Enter the string:");
  String Str=dis.readLine();
  int n=Str.length();
  char c[]=new char[n];
  int size=n;
  for(int i=0;i<size;i++)
 {
        c[i]=Str.charAt(--n);
 }
 String Str1=new String(c);
 if(Str.compareTo(Str1)==0)
{
        System.out.println("Palindrome");
 }
 else
{
        System.out.println("not a palindrome");
 }
}
}
```

Output

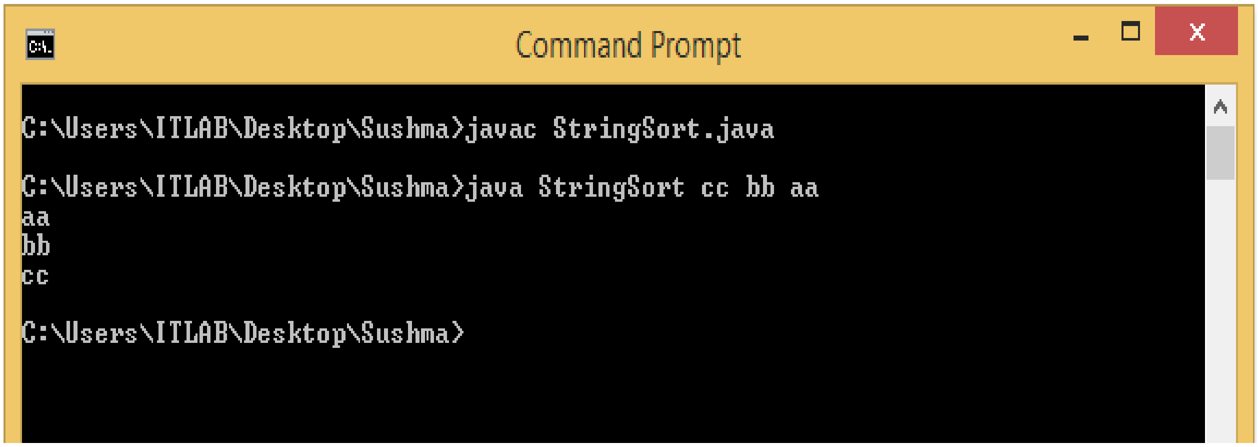


```
C:\Users\ITLAB\Desktop\Sushma>javac palindrome.java
Note: palindrome.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

C:\Users\ITLAB\Desktop\Sushma>java palindrome
Enter the string:
mom
Palindrome

C:\Users\ITLAB\Desktop\Sushma>java palindrome
Enter the string:
java
not a palindrome

C:\Users\ITLAB\Desktop\Sushma>
```

- **Write a Java program for sorting a given list of names in ascending order.**

```
class StringSort
{
  public static void main(String a[])
  {
    String s1;
    for(int i=0;i<a.length-1;i++)
    {
      for(int j=i+1;j<a.length;j++)
     {
       if(a[i].compareTo(a[j])>0)
       {
         s1=a[i];
         a[i]=a[j];
         a[j]=s1;
      }
    }
  }
  for(int j=0;j<a.length;j++)
  {
    System.out.println(a[j]);
  }
 }
}
```

Output

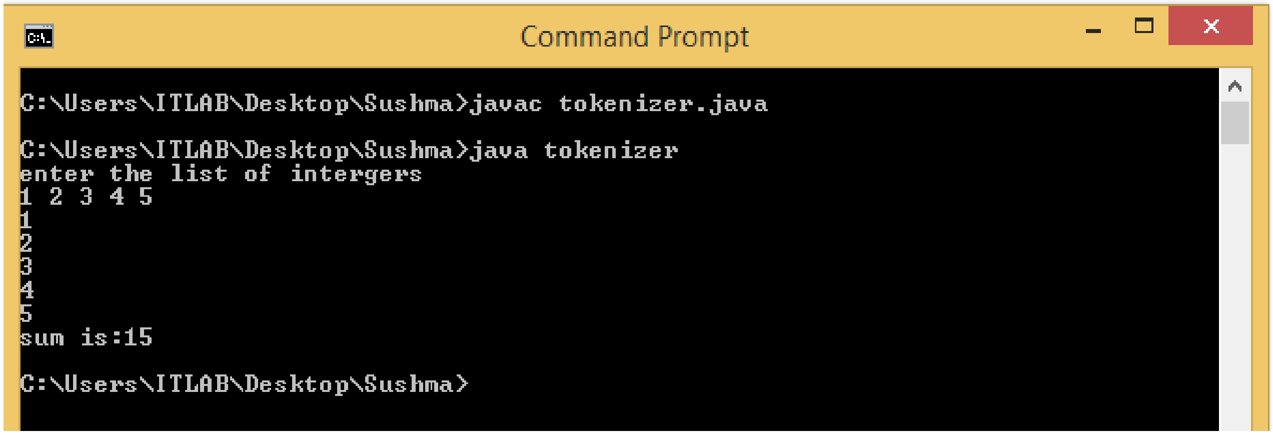


```
C:\Users\ITLAB\Desktop\Sushma>javac StringSort.java

C:\Users\ITLAB\Desktop\Sushma>java StringSort cc bb aa
aa
bb
cc

C:\Users\ITLAB\Desktop\Sushma>
```

- **Write a Java Program that reads a line of integers, and then displays each integer and the sum of all the integers (Use StringTokenizer class of java.util)**

```
import java.util.StringTokenizer;
import java.util.Scanner;
public class tokenizer
{
public static void main(String args[])
{
String s;
int sum=0,x;
Scanner sc=new Scanner(System.in);
System.out.println("enter the list of intergers");
s=sc.nextLine();
StringTokenizer st=new StringTokenizer(s);
while(st.hasMoreTokens())
{
x=Integer.parseInt(st.nextToken());
System.out.println(x);
sum=sum+x;
}
System.out.println("sum is:"+sum);
 }
}
```

Output



```
C:\Users\ITLAB\Desktop\Sushma>javac tokenizer.java

C:\Users\ITLAB\Desktop\Sushma>java tokenizer
enter the list of intergers
1 2 3 4 5
1
2
3
4
5
sum is:15

C:\Users\ITLAB\Desktop\Sushma>
```

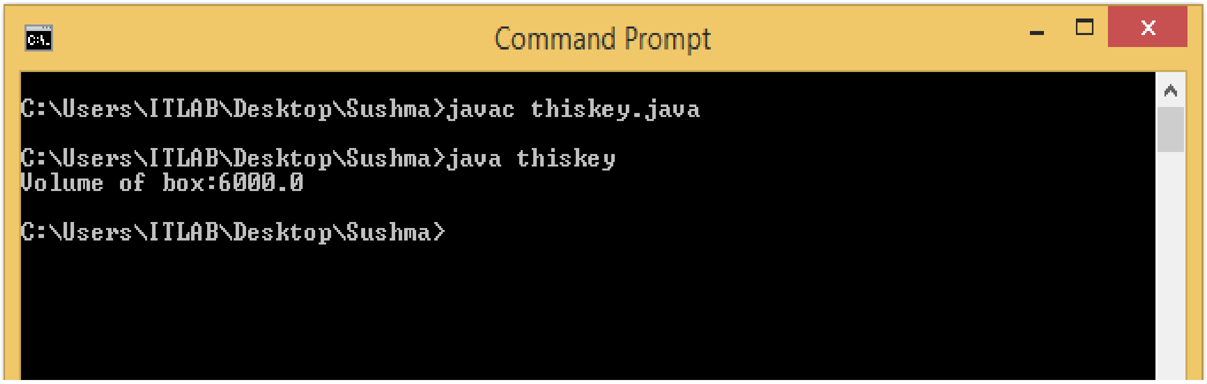
Week-3

Write java programs that uses the following keywords

- **This**

```
class Box1
{
        double l,b,h;
                Box(double l, double b, double h)
                {
                        this.l=l;
                        this.b=b;
                        this.h=h;
        }
        double vol()
        {
                return l*b*h;
        }
}
class thiskey
{
        public static void main(String ar[])
        {
                Box1 b=new Box1(10,20,30);
                double v=b.vol();
                System.out.println("Volume of box:"+v);
        }
}
```

Output

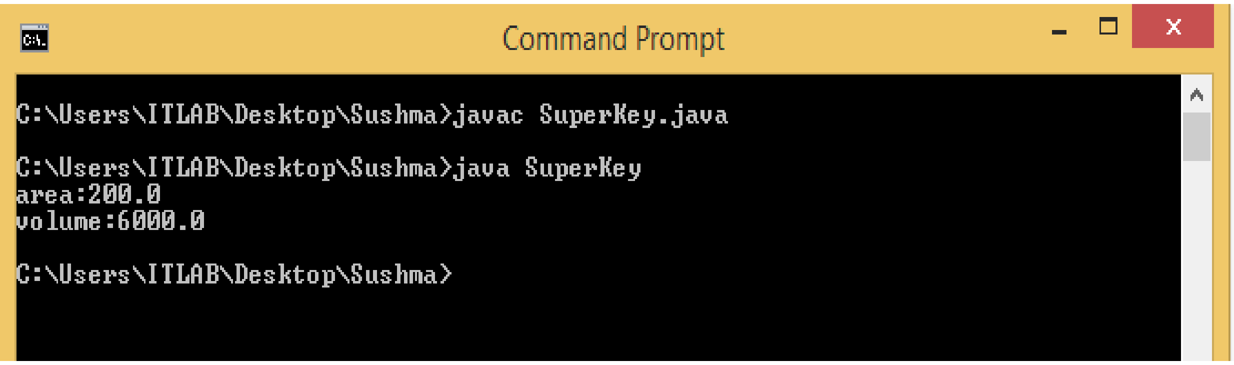


```
C:\Users\ITLAB\Desktop\Sushma>javac thiskey.java

C:\Users\ITLAB\Desktop\Sushma>java thiskey
Volume of box:6000.0

C:\Users\ITLAB\Desktop\Sushma>
```

- **Super**

```
class rect
{
        double length;
        double breadth;
        rect()
        {
                length=1;breadth=1;
        }
        rect(double l,double b)
        {
                length=l;breadth=b;
        }
        double area()
        {
                return length*breadth;
        }
}
class box extends rect
{
        double height;
        box()
        {
                super();
                height=1;
        }
        box(double l,double b,double h)
        {
                super(l,b);
                height=h;
        }
        void volume()
        {
                System.out.println("volume:"+(length*breadth*height));
        }
}
class SuperKey
{
        public static void main(String args[])
        {
                rect r=new rect(20,10);
                System.out.println("area:"+r.area());
                box b=new box(30,40,5);
                b.volume();
        }
}
```

Output

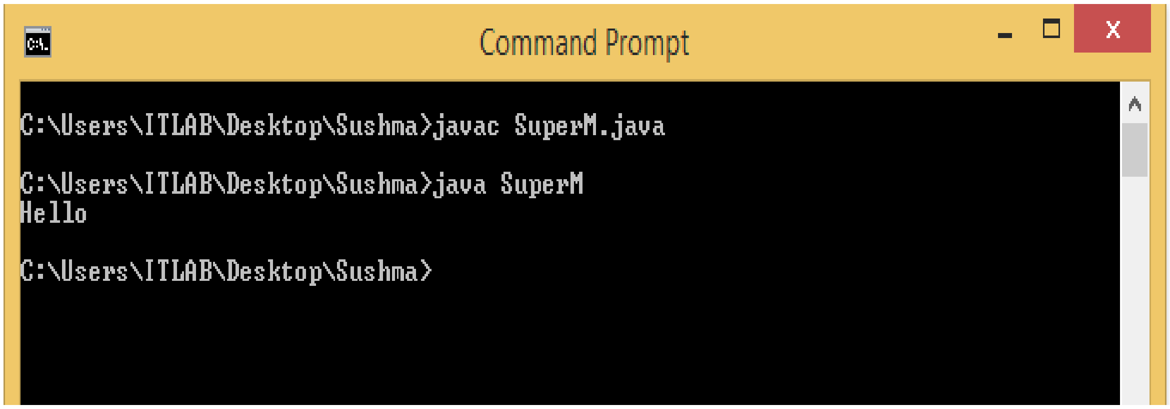


```
C:\Users\ITLAB\Desktop\Sushma>javac SuperKey.java

C:\Users\ITLAB\Desktop\Sushma>java SuperKey
area:200.0
volume:6000.0

C:\Users\ITLAB\Desktop\Sushma>
```

- **Super Method**

```
class ABB
{
 void show()
 {
  System.out.println("Hello");
 }
}
class ABCC extends ABB
{
 void get()
 {
  super.show();
 }
}
class SuperM
{
public static void main(String args[])
{
 ABCC abc=new ABCC();
 abc.get();
}
}
```

Output

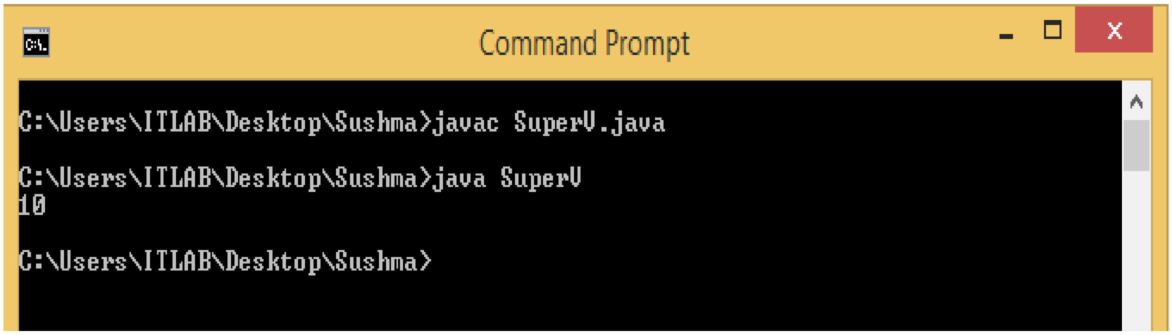


```
C:\Users\ITLAB\Desktop\Sushma>javac SuperM.java

C:\Users\ITLAB\Desktop\Sushma>java SuperM
Hello

C:\Users\ITLAB\Desktop\Sushma>
```

- **Super Variable**

```
class AB
{
 int a=10;
}
class ABC extends AB
{
 int a=20;
 void show()
 {
  System.out.println(super.a);
 }
}
class SuperV
{
public static void main(String args[])
{
 ABC abc=new ABC();
 abc.show();
}
}
```

Output

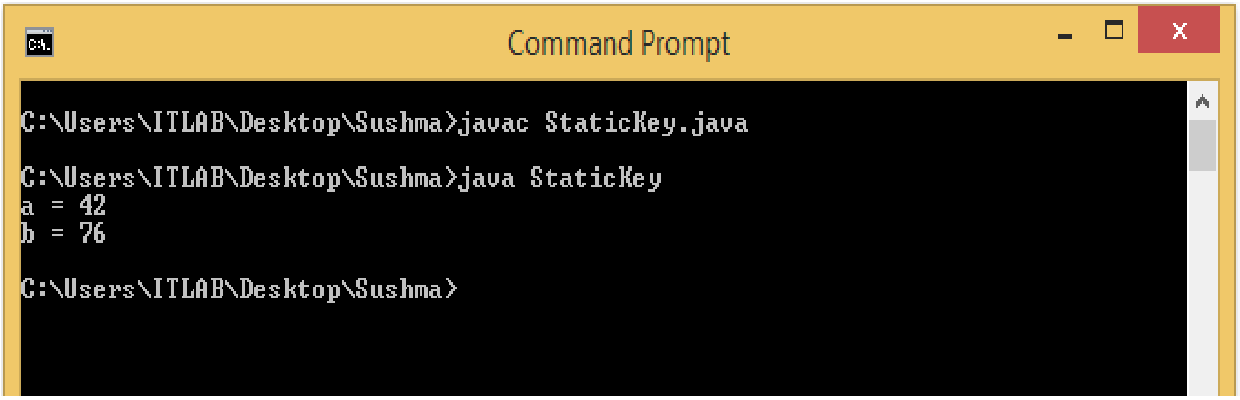


```
C:\Users\ITLAB\Desktop\Sushma>javac SuperV.java

C:\Users\ITLAB\Desktop\Sushma>java SuperV
10

C:\Users\ITLAB\Desktop\Sushma>
```

- **Static**

```
class StaticDemo
{
 static int a=42;
 static int b=76;
 static void callme()
 {
  System.out.println("a = "+a);
 }
}
class StaticKey
{
 public static void main(String args[])
 {
   StaticDemo.callme();
   System.out.println("b = "+StaticDemo.b);
 }
}
```

Output



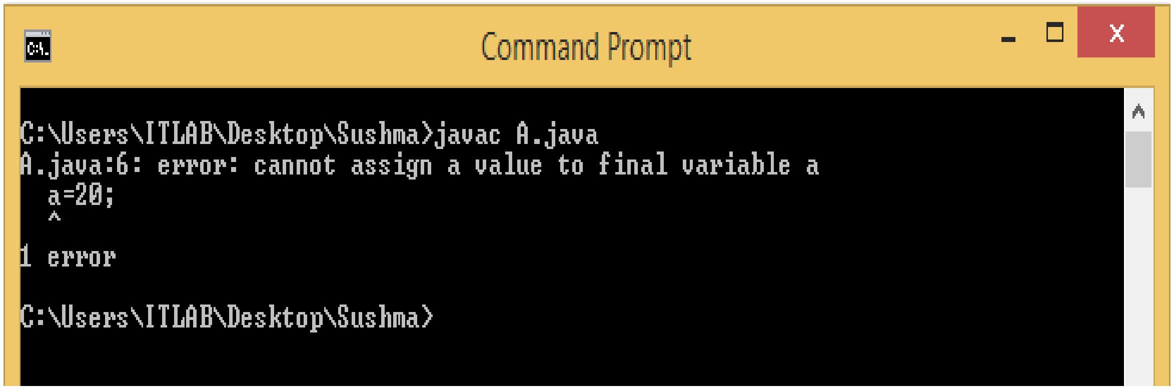
```
C:\Users\ITLAB\Desktop\Sushma>javac StaticKey.java

C:\Users\ITLAB\Desktop\Sushma>java StaticKey
a = 42
b = 76

C:\Users\ITLAB\Desktop\Sushma>
```

- **Final**

Final Variable

```
class A
{
 final int a=10;
 void run()
 {
  a=20;
 }
 public static void main(String args[])
 {
  A a1=new A();
  a1.run();
 }
}
```

Output

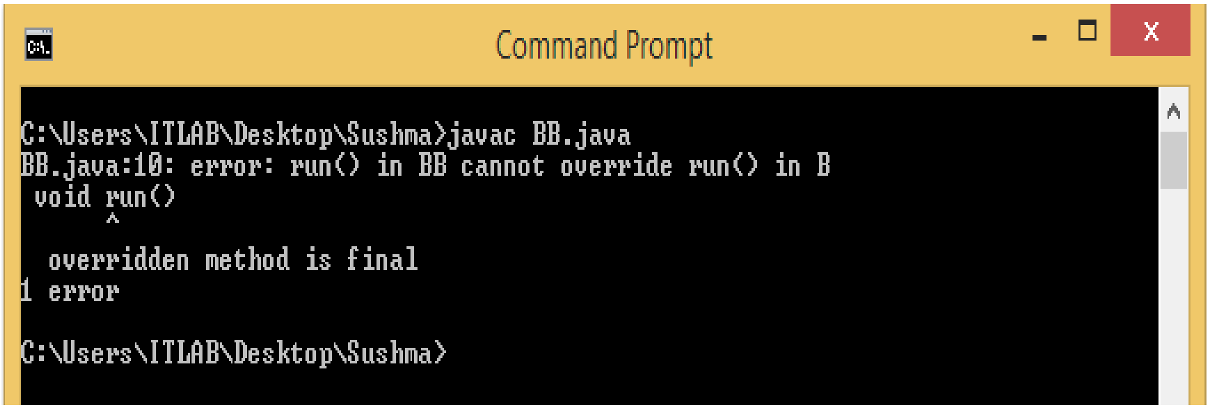


```
C:\Users\ITLAB\Desktop\Sushma>javac A.java
A.java:6: error: cannot assign a value to final variable a
    a=20;
    ^
1 error

C:\Users\ITLAB\Desktop\Sushma>
```

- **Final Method**

```
class B
{
 final void run()
 {
  System.out.println("hello");
 }
}
class BB extends B
{
 void run()
 {
  System.out.println("world");
 }
 public static void main(String args[])
 {
  BB b=new BB();
  b.run();
 }
}
```

Output

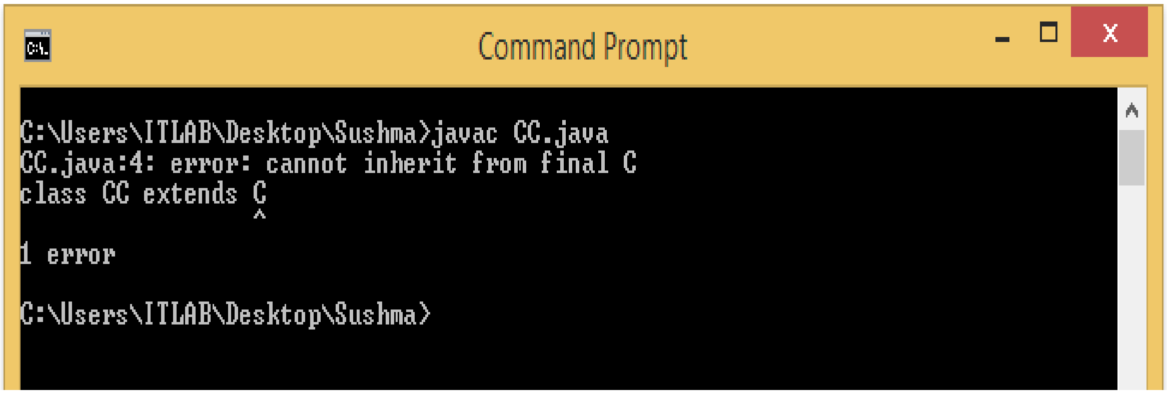


```
C:\Users\ITLAB\Desktop\Sushma>javac BB.java
BB.java:10: error: run() in BB cannot override run() in B
 void run()
      ^
  overridden method is final
1 error

C:\Users\ITLAB\Desktop\Sushma>
```

- **Final Class**

```
final class C
{
}
class CC extends C
{
 void run()
 {
  System.out.println("world");
 }
 public static void main(String args[])
 {
  CC c=new CC();
  c.run();
 }
}
```

Output



```
C:\Users\ITLAB\Desktop\Sushma>javac CC.java
CC.java:4: error: cannot inherit from final C
class CC extends C
                 ^
1 error

C:\Users\ITLAB\Desktop\Sushma>
```

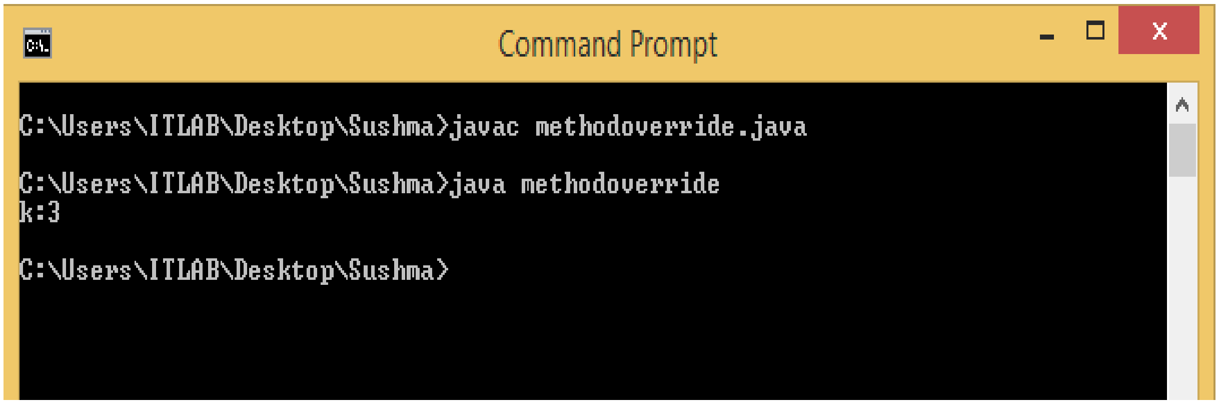
Week-4

- **Write a java program to implement method overriding**

```
class A
{
int i,j;
A(int a,int b)
{
i=a;j=b;
}
void show()
{
System.out.println("i and j="+i+" "+j);
}
}
class B extends A
{
int k;
B(int a,int b,int c)
{
super(a,b);
k=c;
}
void show()
{
System.out.println("k:"+k);
}
}
class methodoverride
{
public static void main(String ar[])
{
```

```
B subOb=new B(1,2,3);
subOb.show();
}
}
```

Output



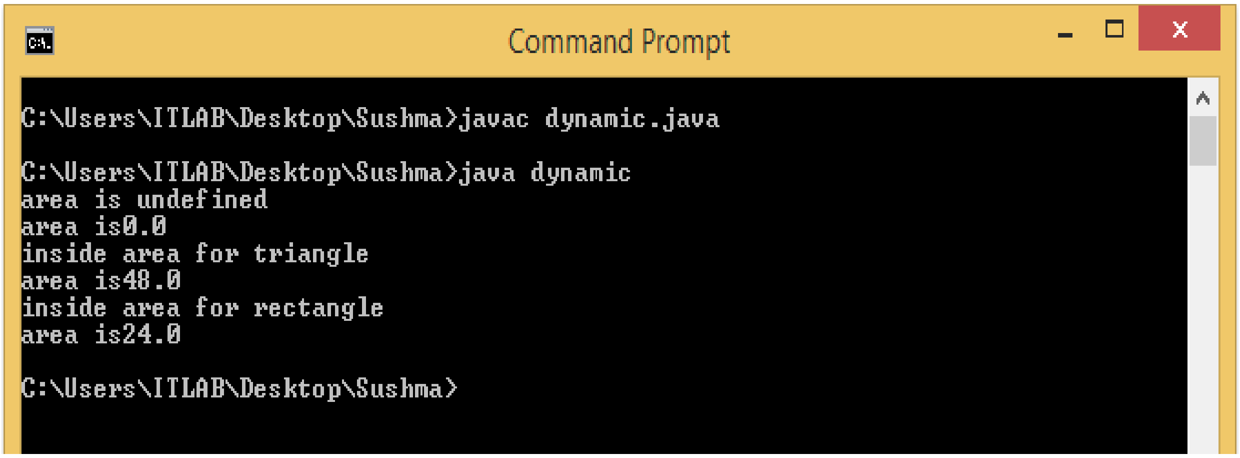
```
C:\Users\ITLAB\Desktop\Sushma>javac methodoverride.java

C:\Users\ITLAB\Desktop\Sushma>java methodoverride
k:3

C:\Users\ITLAB\Desktop\Sushma>
```

- **Write a java program to implement dynamic method dispatch.**

```
class draw
{
 double d1,d2;
 draw(double a,double b)
{
 d1=a;d2=b;
}
double area()
{
 System.out.println("area is undefined");
 return 0;
 }
}
class triangle extends draw
{
 triangle(double a,double b)
 {
 super(a,b);
}
double area()
{
System.out.println("inside area for triangle");
return d1*d2/2;
 }
}
class rectangle extends draw
```

```
{
 rectangle(double a,double b)
{
 super(a,b);
}
double area()
{
 System.out.println("inside area for rectangle");
 return d1*d2;
 }
}
class dynamic
{
 public static void main(String ar[])
  {
 draw d=new draw(10,20);
 triangle t=new triangle(12,8);
 rectangle r=new rectangle(6,4);
 draw ref;
 ref=d;
 System.out.println("area is"+ref.area());
 ref=t;
 System.out.println("area is"+ref.area());
 ref=r;
 System.out.println("area is" +ref.area());
 }
}
```

Output

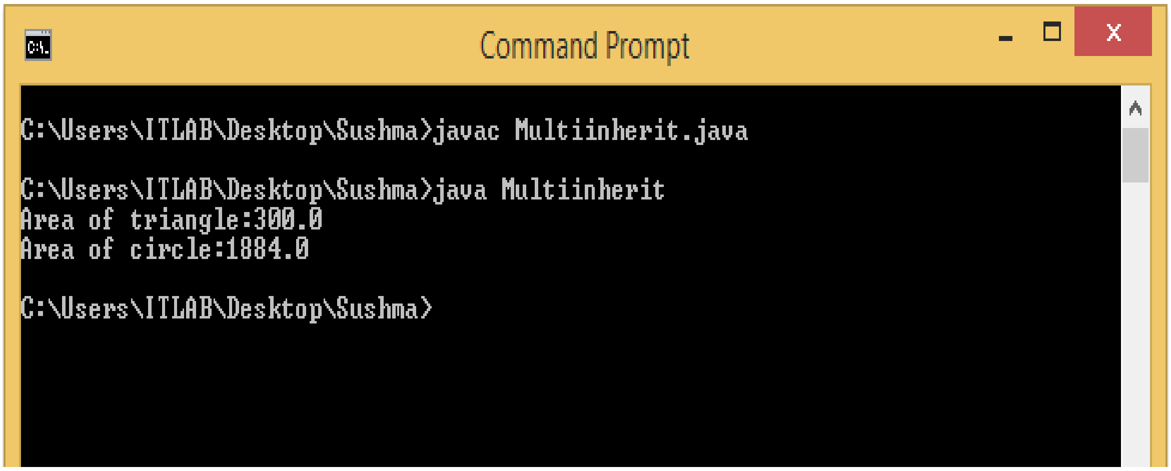


```
C:\Users\ITLAB\Desktop\Sushma>javac dynamic.java

C:\Users\ITLAB\Desktop\Sushma>java dynamic
area is undefined
area is0.0
inside area for triangle
area is48.0
inside area for rectangle
area is24.0

C:\Users\ITLAB\Desktop\Sushma>
```

- **Write a Java program to implement multiple inheritance.**

```
interface draw
{
final static double PI=3.14;
double area(double d1,double d2);
}
class triangle implements draw
{
public double area(double d1,double d2)
{
return (d1*d2)/2;
}
}
class circle implements draw
{
public double area(double d1,double d2)
{
return PI*(d1*d2);
}
}
class Multiinherit
{
 public static void main(String args[])
 {
  triangle t=new triangle();
  circle c=new circle();
  draw d;
  d=t;
  System.out.println("Area of triangle:"+d.area(20,30));
  d=c;
 System.out.println("Area of circle:"+d.area(20,30));
 }
}
```

Output



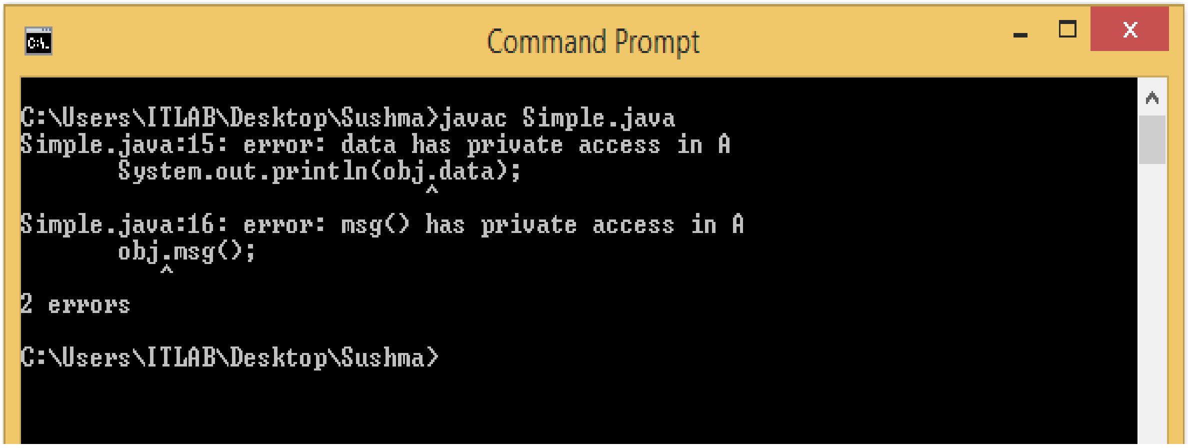
```
C:\Users\ITLAB\Desktop\Sushma>javac Multiinherit.java

C:\Users\ITLAB\Desktop\Sushma>java Multiinherit
Area of triangle:300.0
Area of circle:1884.0

C:\Users\ITLAB\Desktop\Sushma>
```

- **Write a java program that uses access specifiers.**

1. Private

```
 class A
 {
  private int data=40;
  private void msg()
  {
    System.out.println("Hello java");
  }
 }
public class Simple
{
  public static void main(String args[])
  {
    A obj=new A();
    System.out.println(obj.data);
    obj.msg();
  }
 }
```

Output



```
C:\Users\ITLAB\Desktop\Sushma>javac Simple.java
Simple.java:15: error: data has private access in A
       System.out.println(obj.data);
                             ^
Simple.java:16: error: msg() has private access in A
       obj.msg();
          ^
2 errors

C:\Users\ITLAB\Desktop\Sushma>
```

2. default access modifier

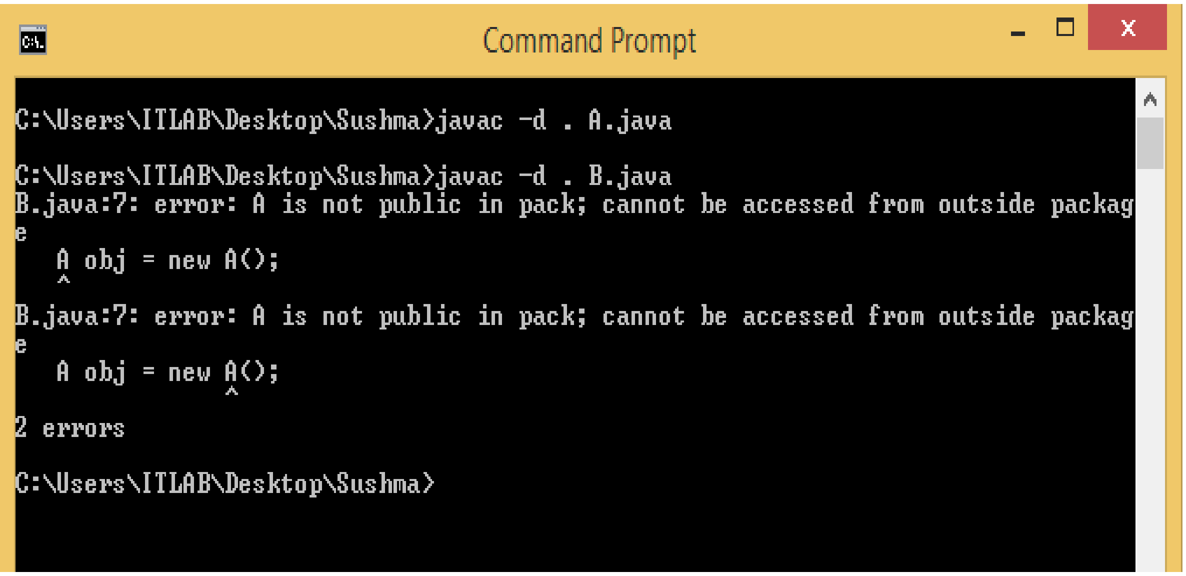
A.java

```
package pack;
class A
{
  void msg()
 {
   System.out.println("Hello");
  }
}
```

B.java

```
package mypack;
import pack.*;
class B
{
  public static void main(String args[])
{
   A obj = new A();
   obj.msg();
 }
}
```

Output



```
C:\Users\ITLAB\Desktop\Sushma>javac -d . A.java

C:\Users\ITLAB\Desktop\Sushma>javac -d . B.java
B.java:7: error: A is not public in pack; cannot be accessed from outside packag
e
   A obj = new A();
   ^
B.java:7: error: A is not public in pack; cannot be accessed from outside packag
e
   A obj = new A();
               ^
2 errors

C:\Users\ITLAB\Desktop\Sushma>
```

3. protected access modifier

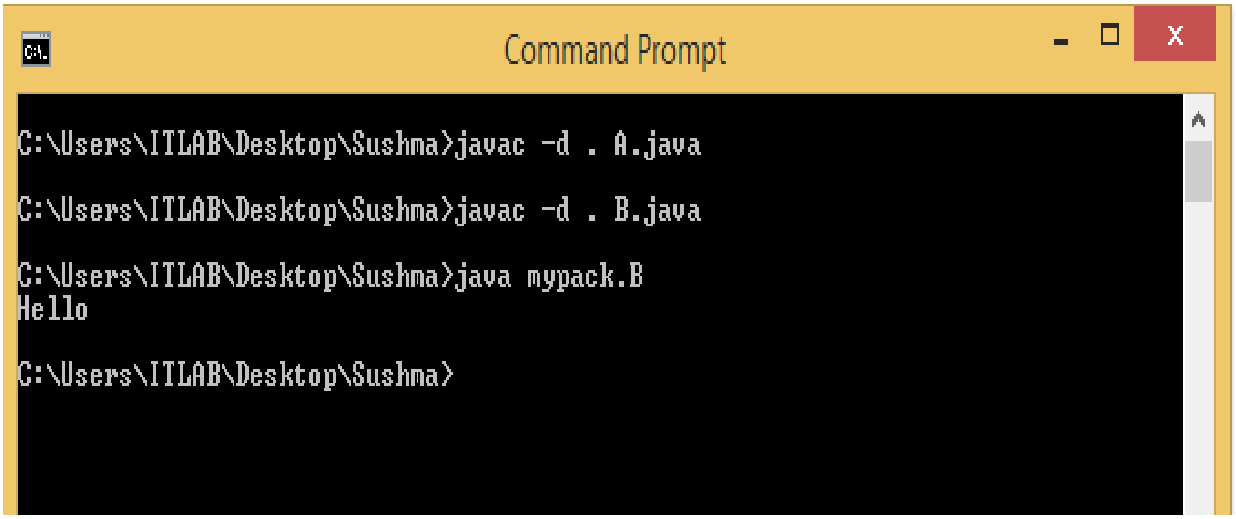
A.java

```
package pack;
public class A
{
 protected void msg()
 {System.out.println("Hello");}
}
```

B.java

```
package mypack;
import pack.*;
class B extends A
{
  public static void main(String args[])
 {
   B obj = new B();
   obj.msg();
  }
}
```

Output



```
C:\Users\ITLAB\Desktop\Sushma>javac -d . A.java

C:\Users\ITLAB\Desktop\Sushma>javac -d . B.java

C:\Users\ITLAB\Desktop\Sushma>java mypack.B
Hello

C:\Users\ITLAB\Desktop\Sushma>
```

4. public access modifier

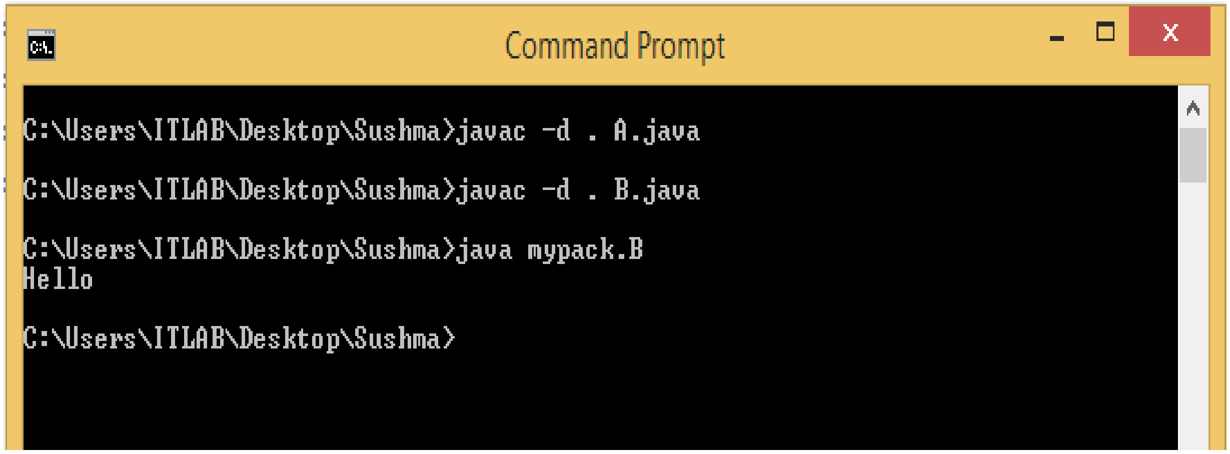
A.java

```
package pack;
public class A
{
        public void msg()
        {System.out.println("Hello");}
}
```

B.java

```
package mypack;
import pack.*;
class B
{
  public static void main(String args[])
{
   A obj = new A();
   obj.msg();
  }
}
```

Output



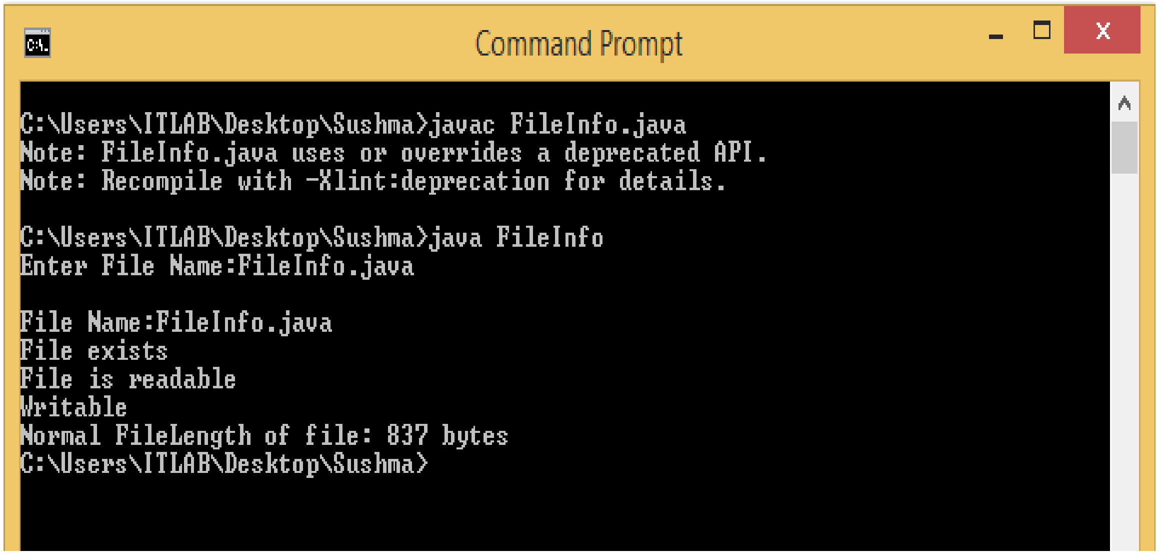
```
C:\Users\ITLAB\Desktop\Sushma>javac -d . A.java

C:\Users\ITLAB\Desktop\Sushma>javac -d . B.java

C:\Users\ITLAB\Desktop\Sushma>java mypack.B
Hello

C:\Users\ITLAB\Desktop\Sushma>
```

Week-5

- **Write a Java program that reads a file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.**

```
import java.io.*;
class FileInfo
{
public static void main(String ar[]) throws Exception
{
DataInputStream dis=new DataInputStream(System.in);
System.out.printf("Enter File Name:");
String fname =dis.readLine();
File f1=new File(fname);
System.out.print("\nFile Name:"+fname);
if(f1.exists())
{
System.out.printf("\nFile exists");
if(f1.canRead())
System.out.printf("\nFile is readable");
else
System.out.printf("\nNot readable");
if(f1.canWrite())
System.out.printf("\nWritable");
else
System.out.printf("\nNot Writable");
if(f1.isDirectory())
System.out.printf("\nDirectory File");
else if(f1.isAbsolute())
System.out.printf("\nAbsolute File");
else
System.out.printf("\nNormal File");
System.out.printf("Length of file: "+f1.length()+" bytes");
}
else
System.out.printf("\nFile does not exist.");
}
}
```

Output

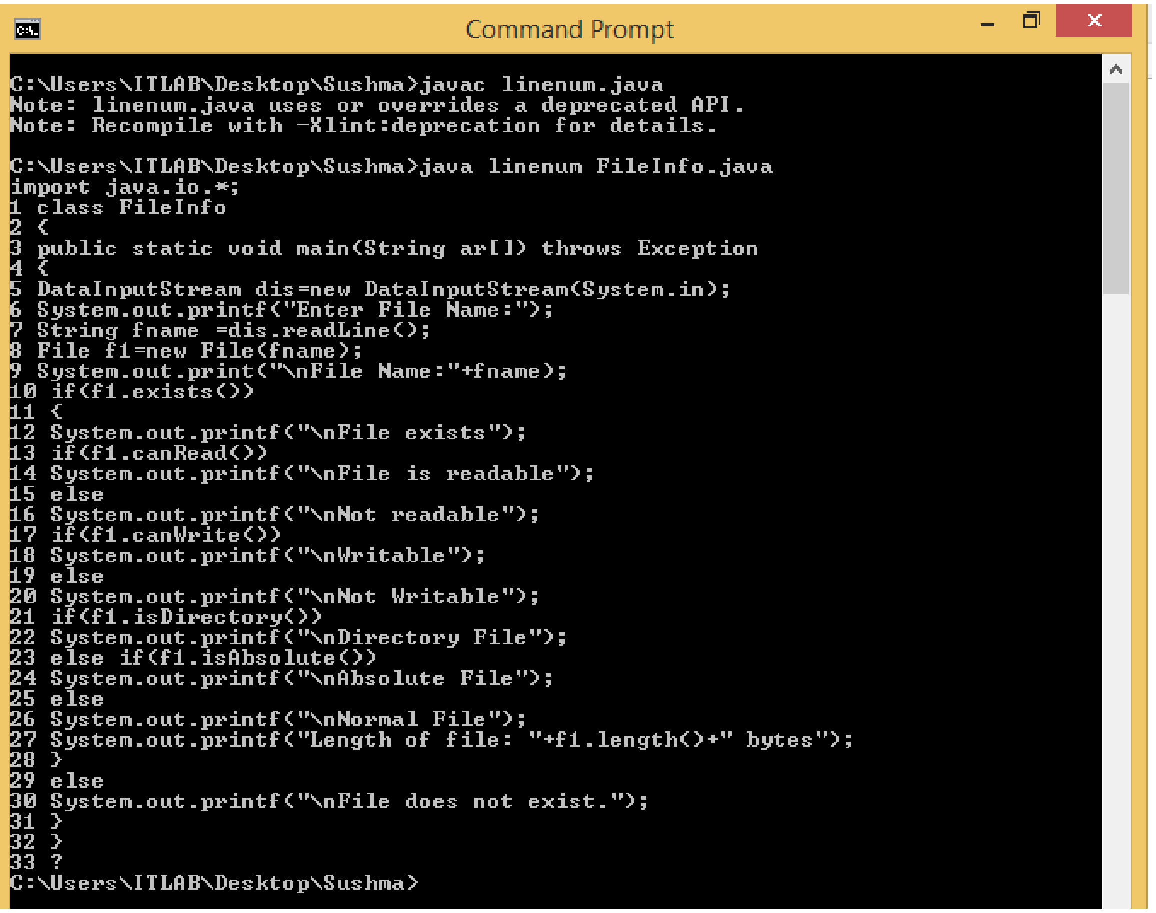


```
C:\Users\ITLAB\Desktop\Sushma>javac FileInfo.java
Note: FileInfo.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

C:\Users\ITLAB\Desktop\Sushma>java FileInfo
Enter File Name:FileInfo.java

File Name:FileInfo.java
File exists
File is readable
Writable
Normal FileLength of file: 837 bytes
C:\Users\ITLAB\Desktop\Sushma>
```

- **Write a Java program that reads a file and displays the file on the screen, with a line number before each line.**

```
import java.io.*;
class linenum
{
public static void main(String ar[]) throws IOException
{
 FileInputStream fil;
LineNumberInputStream line;
int i;
try
{
fil=new FileInputStream(ar[0]);
line= new LineNumberInputStream(fil);
}
catch(FileNotFoundException e)
{
System.out.println("No such file found");
return;
}
do
{
i=line.read();
if(i=='\n')
{
System.out.println();
System.out.print(line.getLineNumber()+" ");
}
else
System.out.print((char)i);
}while(i!=-1);
fil.close();
line.close();
}
}
```

Output

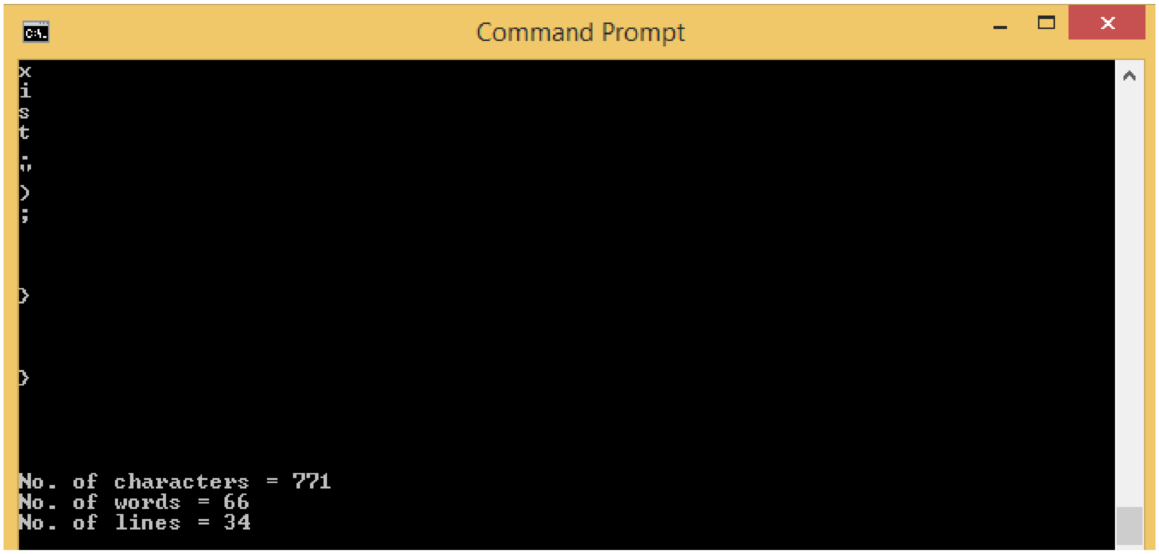


```
C:\Users\ITLAB\Desktop\Sushma>javac linenum.java
Note: linenum.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.

C:\Users\ITLAB\Desktop\Sushma>java linenum FileInfo.java
import java.io.*;
1 class FileInfo
2 {
3 public static void main(String ar[]) throws Exception
4 {
5 DataInputStream dis=new DataInputStream(System.in);
6 System.out.printf("Enter File Name:");
7 String fname =dis.readLine();
8 File f1=new File(fname);
9 System.out.print("\nFile Name:"+fname);
10 if(f1.exists())
11 {
12 System.out.printf("\nFile exists");
13 if(f1.canRead())
14 System.out.printf("\nFile is readable");
15 else
16 System.out.printf("\nNot readable");
17 if(f1.canWrite())
18 System.out.printf("\nWritable");
19 else
20 System.out.printf("\nNot Writable");
21 if(f1.isDirectory())
22 System.out.printf("\nDirectory File");
23 else if(f1.isAbsolute())
24 System.out.printf("\nAbsolute File");
25 else
26 System.out.printf("\nNormal File");
27 System.out.printf("Length of file: "+f1.length()+" bytes");
28 }
29 else
30 System.out.printf("\nFile does not exist.");
31 }
32 }
33 ?
C:\Users\ITLAB\Desktop\Sushma>
```

- **Write a Java program that displays the number of characters, lines and words in a text file.**

```
import java.io.*;
public class FileCount
{
public static void main(String args[]) throws Exception
{
DataInputStream dis=new DataInputStream(System.in);
System.out.println("Enter file name:");
String fname=dis.readLine();
try
{
FileInputStream fis=new FileInputStream(fname);
int cchar=0,cwords=0,clines=1;
int c;
System.out.println("The contents of file:");
while((c=fis.read())!=-1)
{
cchar++;
if((char)c==' ')
{
++cwords;
--cchar;
}
if((char)c=='\n')
{
++clines;
++cwords;
cchar=cchar-1;
}
System.out.println((char)c);
}
System.out.println();
System.out.println("No. of characters = "+cchar);
System.out.println("No. of words = "+cwords);
System.out.println("No. of lines = "+clines);
}
catch(FileNotFoundException fnot)
{
System.out.println("File Not Found");
}
}
}
```

Output



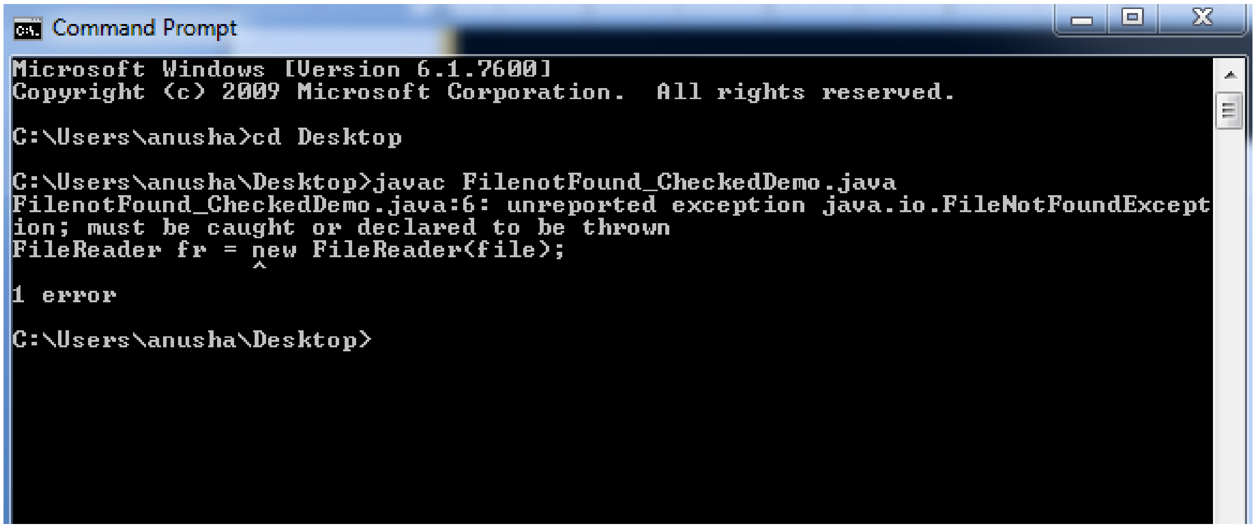
```
x
i
s
t
.
"
)
;



}



}




No. of characters = 771
No. of words = 66
No. of lines = 34
```

Week-6

- Write a Java program for handling Checked Exceptions.

```
import java.io.File;
 import java.io.FileReader;
public class FilenotFound_CheckedDemo
{ public static void main(String args[])
{ File file = new File("E://file.txt");
FileReader fr = new FileReader(file);
}
 }
```

Output

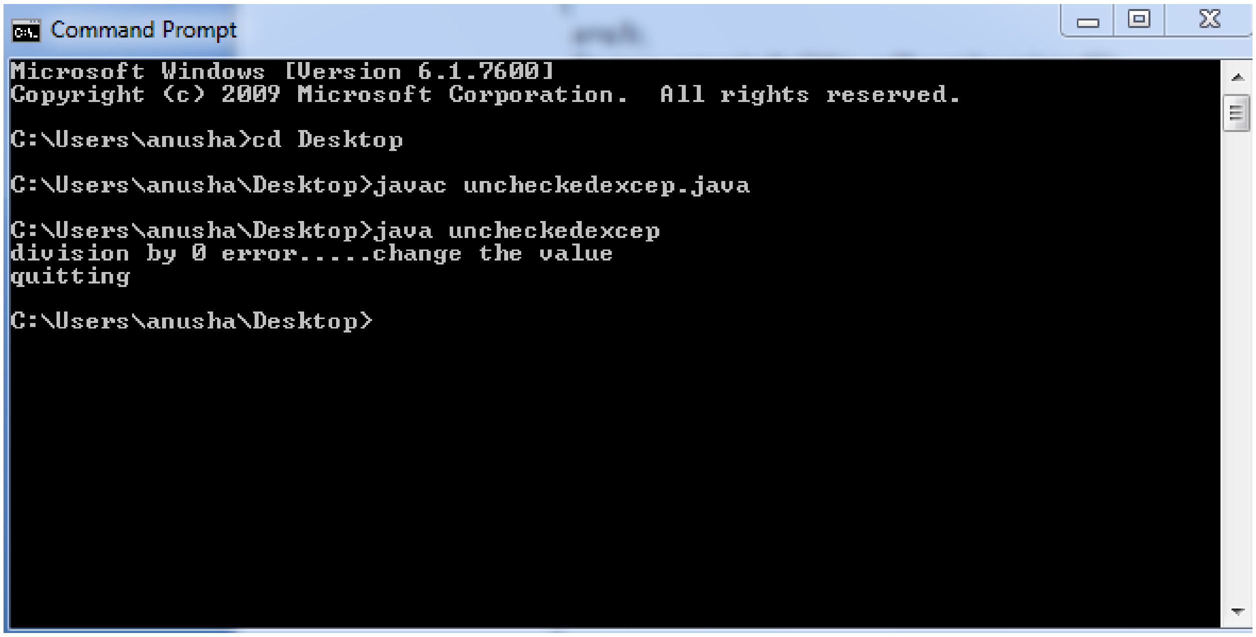


```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac FilenotFound_CheckedDemo.java
FilenotFound_CheckedDemo.java:6: unreported exception java.io.FileNotFoundExcept
ion; must be caught or declared to be thrown
FileReader fr = new FileReader(file);
                ^
1 error

C:\Users\anusha\Desktop>
```

- **Write a Java program for handling Unchecked Exceptions.**

```
class uncheckedexcep
{
  public static void main(String args[])
  {
    int a=10;
    int b=0;
    try
       {
        a=a/b;
        System.out.println("this will not be printed");
       }
     catch(ArithmeticException ae)
      {
       System.out.println("division by 0 error.....change the value");
      }
      System.out.println("quitting");
  }
}
```

Output



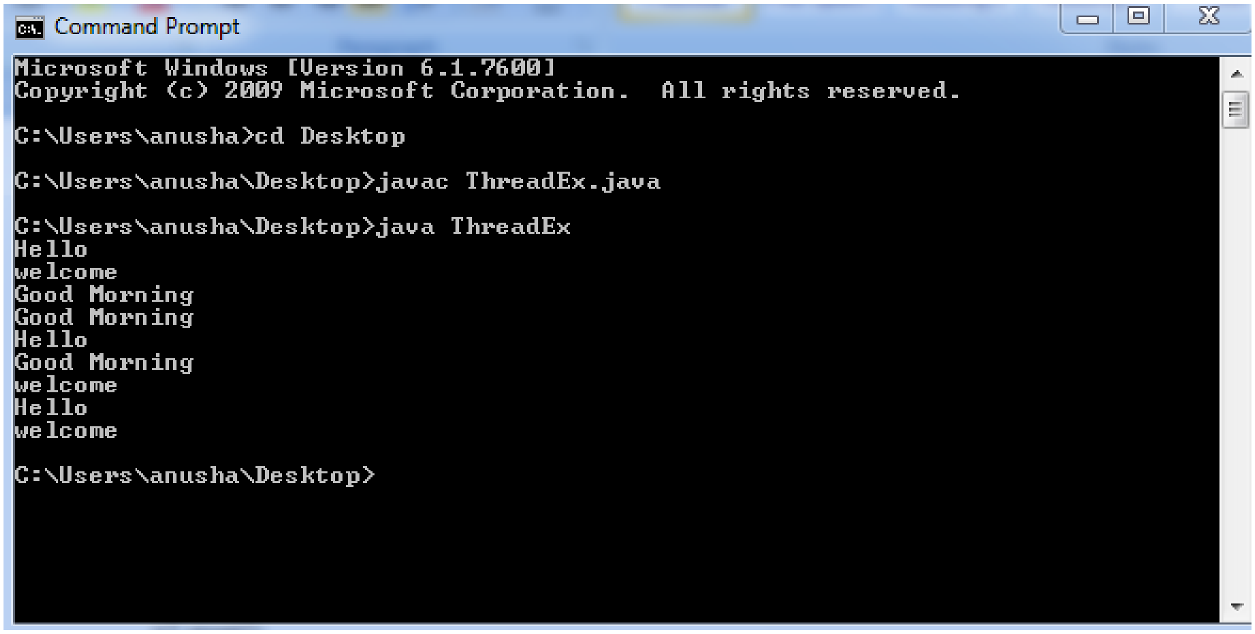
```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac uncheckedexcep.java

C:\Users\anusha\Desktop>java uncheckedexcep
division by 0 error.....change the value
quitting

C:\Users\anusha\Desktop>
```

Week-7

- Write a Java program that creates three threads. First thread displays "Good Morning" every one second, the second thread displays "Hello" every two seconds and the third thread displays "Welcome" every three seconds.

```
class Thread1 extends Thread
{
 public void run()
 {
  for(int i=0;i<3;i++){
   System.out.println("Good Morning ");
try{Thread.sleep(1000);}catch(Exception e){}}

 }
}
class Thread2 extends Thread
{
 public void run()
 {
  for(int i=0;i<3;i++){
   System.out.println("Hello ");
try{Thread.sleep(2000);}catch(Exception e){}}

 }
}
class Thread3 extends Thread
{
 public void run()
 {for(int i=0;i<3;i++){

   System.out.println("welcome");
try{Thread.sleep(3000);}catch(Exception e){}
  }
 }
}
class ThreadEx
{
 public static void main(String args[])
 {
  Thread1 t1=new Thread1();
  Thread2 t2=new Thread2();
  Thread3 t3=new Thread3();
  t1.start();
  t2.start();
  t3.start();

 }
}
```

Output



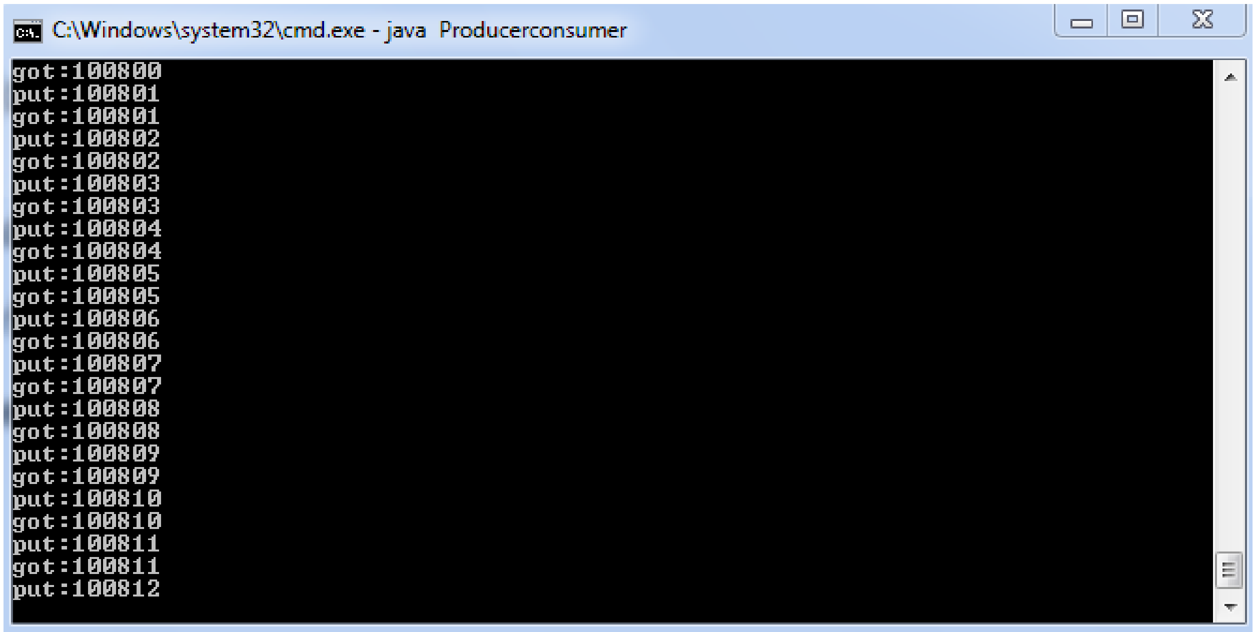
```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac ThreadEx.java

C:\Users\anusha\Desktop>java ThreadEx
Hello
welcome
Good Morning
Good Morning
Hello
Good Morning
welcome
Hello
welcome

C:\Users\anusha\Desktop>
```

- Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.

```
class Q
{
int n;
boolean flag=false;

synchronized int get()
{
if(!flag)
try
{
wait();
}
catch(InterruptedException e)
{
System.out.println("Exception caught");
}
System.out.println("got:"+n);
flag=false;
notify();
return n;
}
synchronized void put(int n)
{
if(flag)
try
{
wait();
}
catch(InterruptedException e)
{
System.out.println("Exception caught");
}
this.n=n;
flag=true;
System.out.println("put:"+n);
notify();
}
}
class Producer implements Runnable
{

Q q;
Producer(Q q)
{
this.q=q;
new Thread(this,"Producer").start();
}
public void run()
{
int i=0;
while(true)
{
q.put(i++);
}
}
}
class Consumer implements Runnable
{
Q q;
```

```
Consumer(Q q)
{
this.q=q;
new Thread(this,"Consumer").start();
}
public void run()
{
while(true)
{
q.get();
}
}
}
class Producerconsumer
{
public static void main(String args[])
{
Q q=new Q();
new Producer(q);
new Consumer(q);
System.out.println("press control-c to stop");
}
}
```

Output



```
got:100800
put:100801
got:100801
put:100802
got:100802
put:100803
got:100803
put:100804
got:100804
put:100805
got:100805
put:100806
got:100806
put:100807
got:100807
put:100808
got:100808
put:100809
got:100809
put:100810
got:100810
put:100811
got:100811
put:100812
```

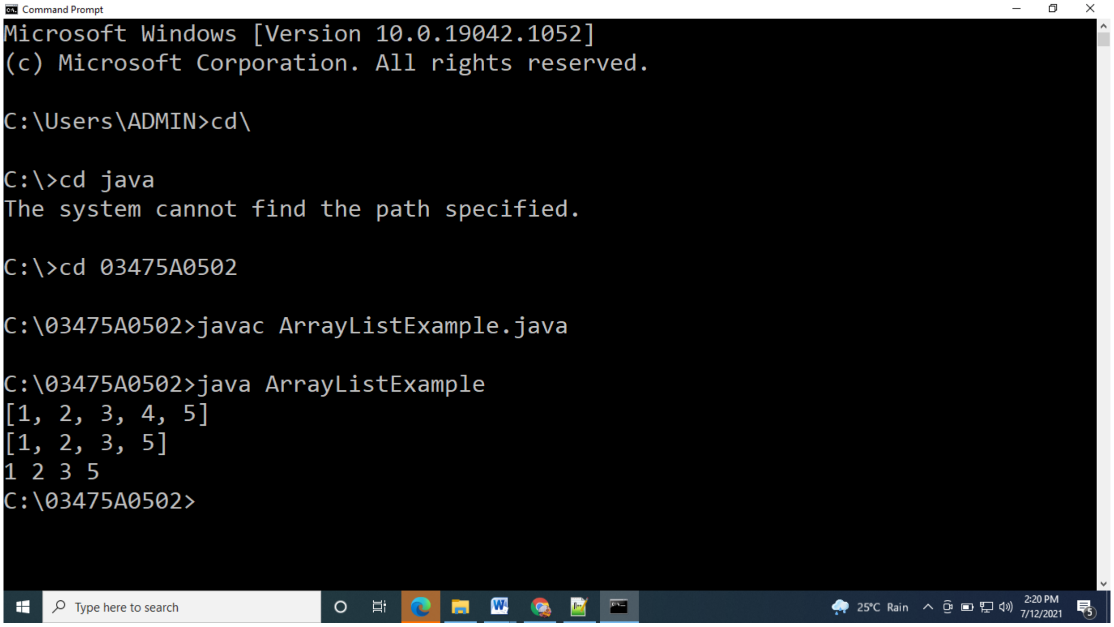
Week-8

- Java program to demonstrate the working of ArrayList in Java

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

class ArrayListExample {
    public static void main(String[] args)
    {
        // Size of the
        // ArrayList
        int n = 5;

        // Declaring the ArrayList with
        // initial size n
        ArrayList<Integer> arrli
            = new ArrayList<Integer>(n);

        // Appending new elements at
        // the end of the list
        for (int i = 1; i <= n; i++)
            arrli.add(i);

        // Printing elements
        System.out.println(arrli);

        // Remove element at index 3
        arrli.remove(3);

        // Displaying the ArrayList
        // after deletion
        System.out.println(arrli);
```

```
                // Printing elements one by one
                for (int i = 0; i < arrli.size(); i++)
                        System.out.print(arrli.get(i) + " ");
        }
}
```

Output:

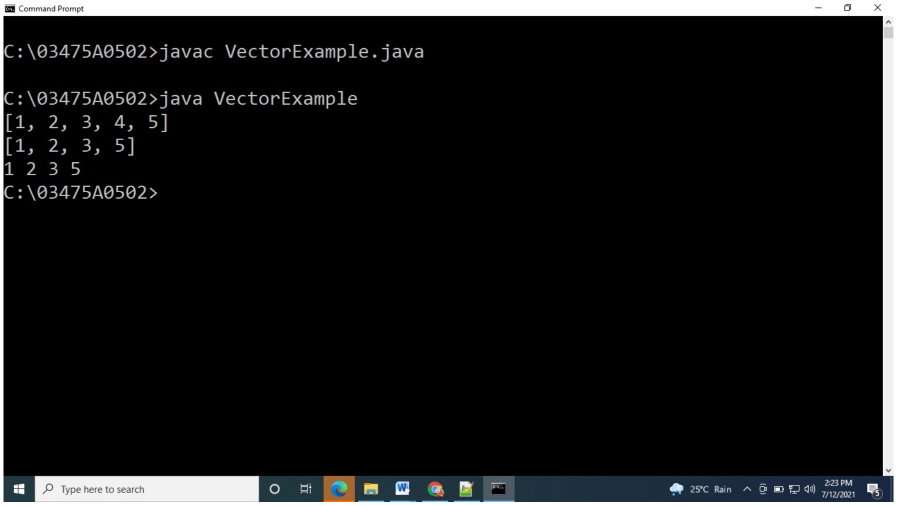


```
Microsoft Windows [Version 10.0.19042.1052]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ADMIN>cd\

C:\>cd java
The system cannot find the path specified.

C:\>cd 03475A0502

C:\03475A0502>javac ArrayListExample.java

C:\03475A0502>java ArrayListExample
[1, 2, 3, 4, 5]
[1, 2, 3, 5]
1 2 3 5
C:\03475A0502>
```

- Java program to demonstrate the working of Vector

```
import java.io.*;
import java.util.*;
class VectorExample {
    public static void main(String[] args)
    {
            // Size of the
            // Vector
            int n = 5;
            // Declaring the Vector with
            // initial size n
            Vector<Integer> v = new Vector<Integer>(n);
            // Appending new elements at
            // the end of the vector
            for (int i = 1; i <= n; i++)
                    v.add(i);
            // Printing elements
            System.out.println(v);

            // Remove element at index 3
            v.remove(3);

            // Displaying the vector
            // after deletion
            System.out.println(v);

            // Printing elements one by one
            for (int i = 0; i < v.size(); i++)
                    System.out.print(v.get(i) + " ");
    }
}
```

Output:

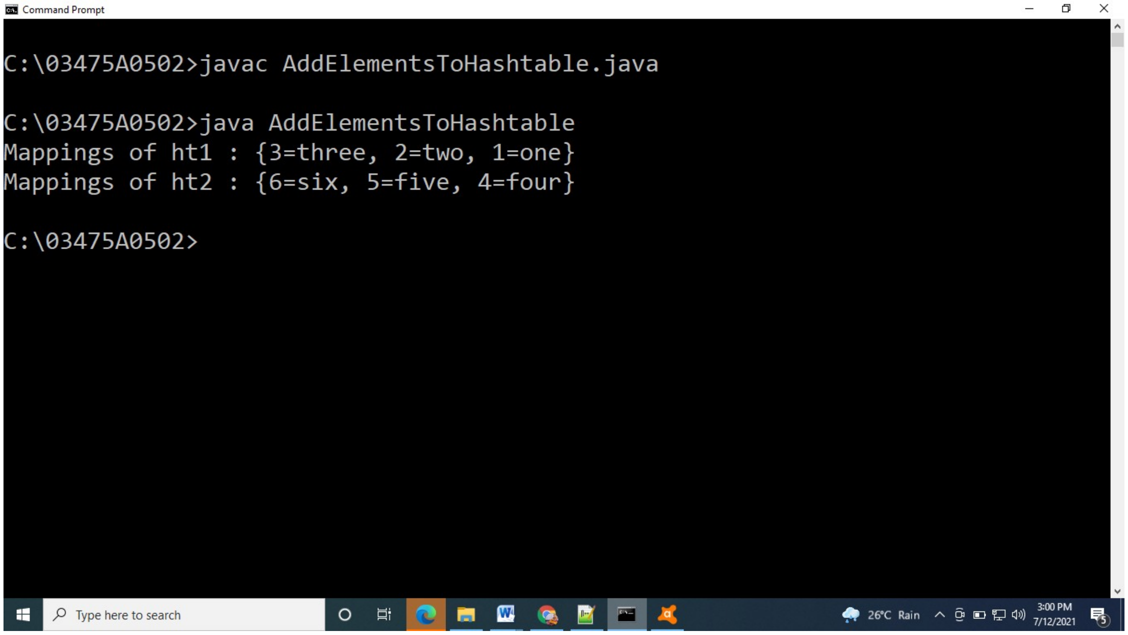


```
C:\03475A0502>javac VectorExample.java

C:\03475A0502>java VectorExample
[1, 2, 3, 4, 5]
[1, 2, 3, 5]
1 2 3 5
C:\03475A0502>
```

- Java program to demonstrate adding elements to Hashtable

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

class AddElementsToHashtable {
    public static void main(String args[])
    {
        // No need to mention the
        // Generic type twice
        Hashtable<Integer, String> ht1 = new Hashtable<>();

        // Initialization of a Hashtable
        // using Generics
        Hashtable<Integer, String> ht2
            = new Hashtable<Integer, String>();

        // Inserting the Elements
        // using put() method
        ht1.put(1, "one");
        ht1.put(2, "two");
        ht1.put(3, "three");

        ht2.put(4, "four");
        ht2.put(5, "five");
        ht2.put(6, "six");

        // Print mappings to the console
        System.out.println("Mappings of ht1 : " + ht1);
        System.out.println("Mappings of ht2 : " + ht2);
    }
}
```

Output:

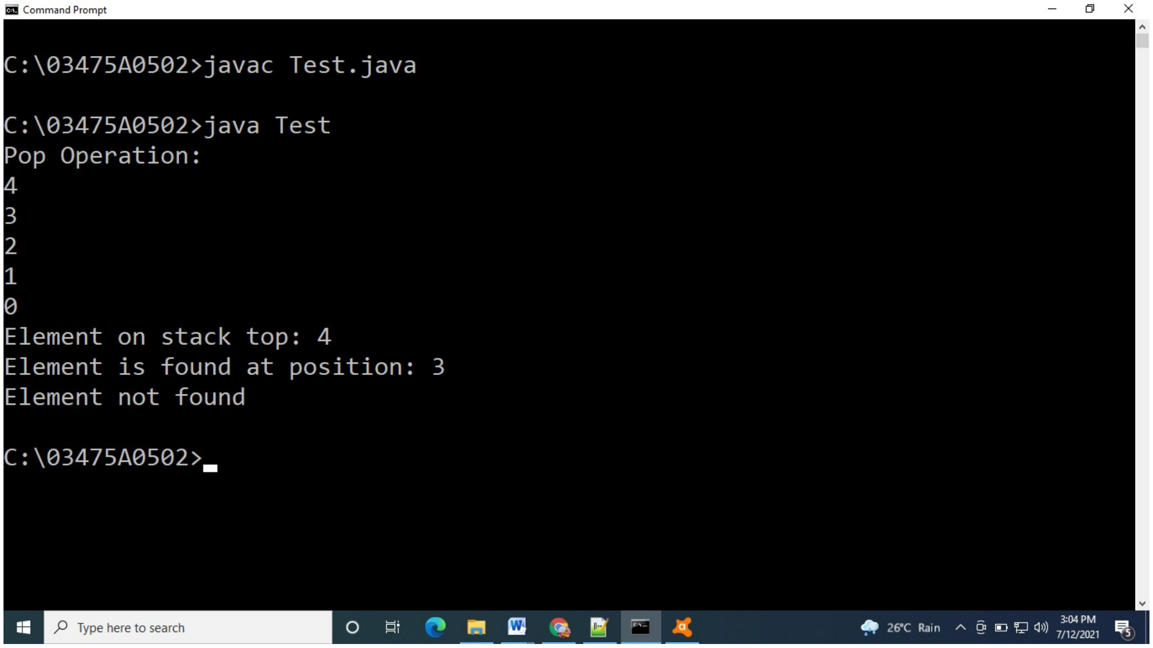


```
C:\03475A0502>javac AddElementsToHashtable.java

C:\03475A0502>java AddElementsToHashtable
Mappings of ht1 : {3=three, 2=two, 1=one}
Mappings of ht2 : {6=six, 5=five, 4=four}

C:\03475A0502>
```

- Java code for stack implementation

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

class Test
{
    // Pushing element on the top of the stack
    static void stack_push(Stack<Integer> stack)
    {
        for(int i = 0; i < 5; i++)
        {
            stack.push(i);
        }
    }

    // Popping element from the top of the stack
    static void stack_pop(Stack<Integer> stack)
    {
        System.out.println("Pop Operation:");

        for(int i = 0; i < 5; i++)
        {
            Integer y = (Integer) stack.pop();
            System.out.println(y);
        }
    }

    // Displaying element on the top of the stack
    static void stack_peek(Stack<Integer> stack)
    {
        Integer element = (Integer) stack.peek();
        System.out.println("Element on stack top: " + element);
    }

    // Searching element in the stack
    static void stack_search(Stack<Integer> stack, int element)
    {
        Integer pos = (Integer) stack.search(element);

        if(pos == -1)
            System.out.println("Element not found");
        else
            System.out.println("Element is found at position: " + pos);
    }



    public static void main (String[] args)
    {
        Stack<Integer> stack = new Stack<Integer>();

        stack_push(stack);
        stack_pop(stack);
        stack_push(stack);
        stack_peek(stack);
        stack_search(stack, 2);
        stack_search(stack, 6);
    }
}
```

Output:



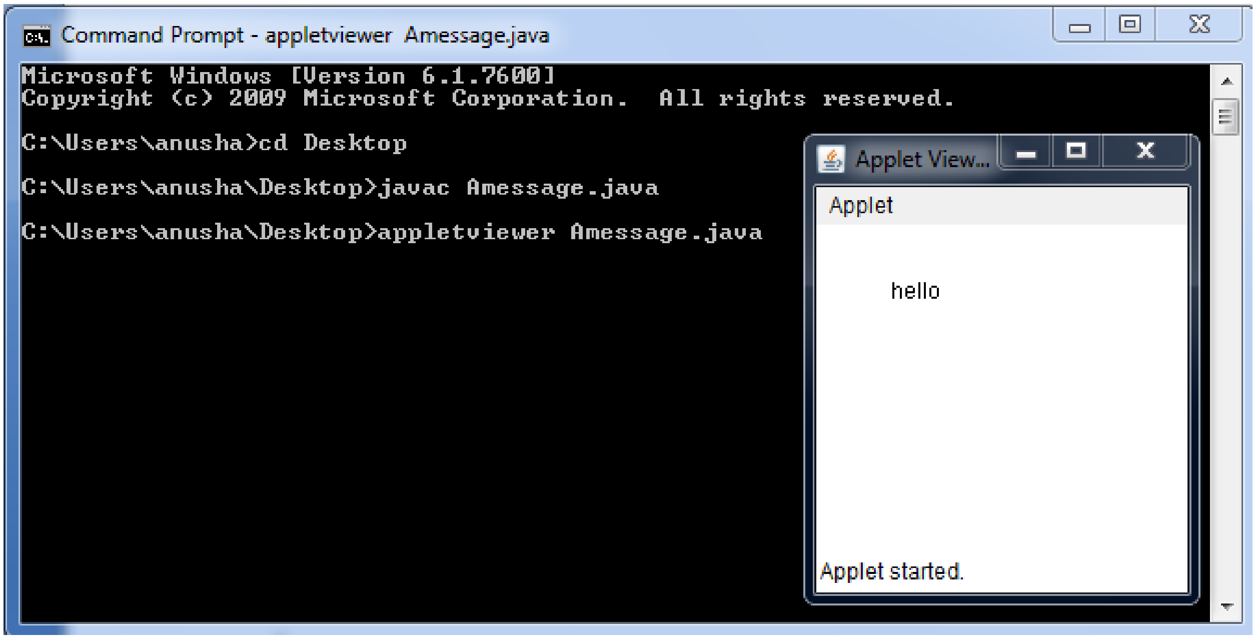
```
C:\03475A0502>javac Test.java

C:\03475A0502>java Test
Pop Operation:
4
3
2
1
0
Element on stack top: 4
Element is found at position: 3
Element not found

C:\03475A0502>
```

Week-9

a) Develop an applet that displays a simple message.

```
import java.awt.*;
import java.applet.*;
/*<applet code="Amessage.class" width=200 height=300>
</applet>*/
public class Amessage extends Applet
{
public void paint(Graphics g)
{
g.drawString("hello",40,40);
}
}
```

Output:



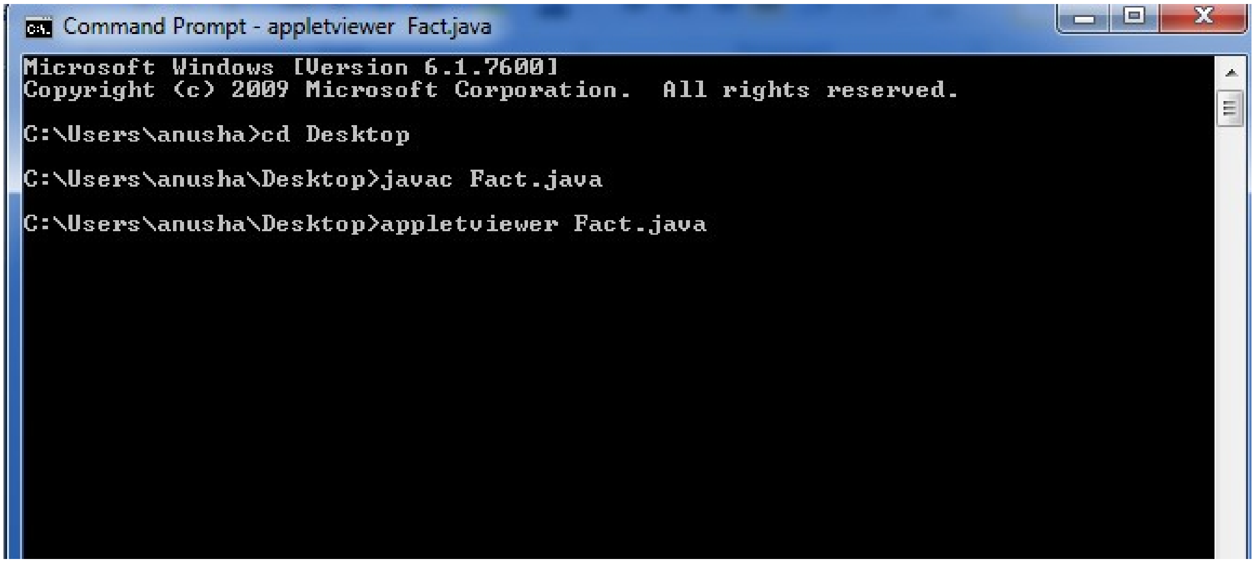
```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac Amessage.java

C:\Users\anusha\Desktop>appletviewer Amessage.java
```

```
Applet

        hello

Applet started.
```

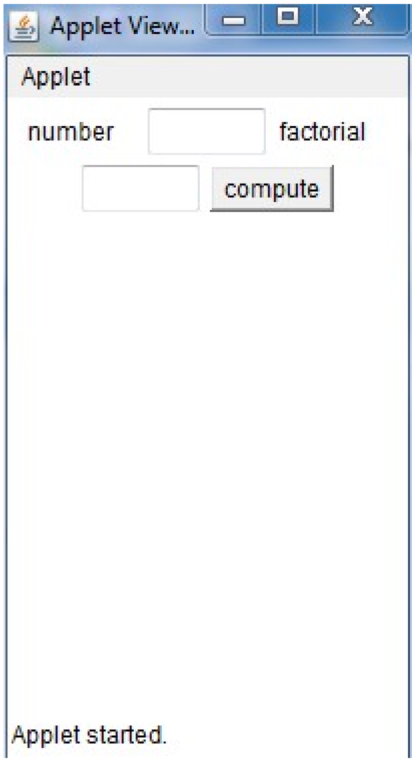
- Develop an applet that receives an integer in one text field , and computes its factorial value and returns it in another text field , when button named “Compute” is clicked.

```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
/*<applet code="Fact.class" width=200 height=300>
</applet>*/
public class Fact extends Applet implements ActionListener
{
 TextField x1,x2;
 public void init()
 {
  Label L1=new Label("number");
  Label L2=new Label("factorial");
  x1=new TextField(5);
  x2=new TextField(5);
  Button b=new Button("compute");
  add(L1);
  add(x1);
  add(L2);
  add(x2);
  add(b);
  x1.addActionListener(this);
  x2.addActionListener(this);
  b.addActionListener(this);
 }
 public void actionPerformed(ActionEvent j)
 {
  int fac=1,m;
  String m1=x1.getText();
  m=Integer.parseInt(m1);
  for(int i=1;i<=m;i++)
  {
   fac=fac*i;
  }
  x2.setText(" "+fac);
 }
}
```

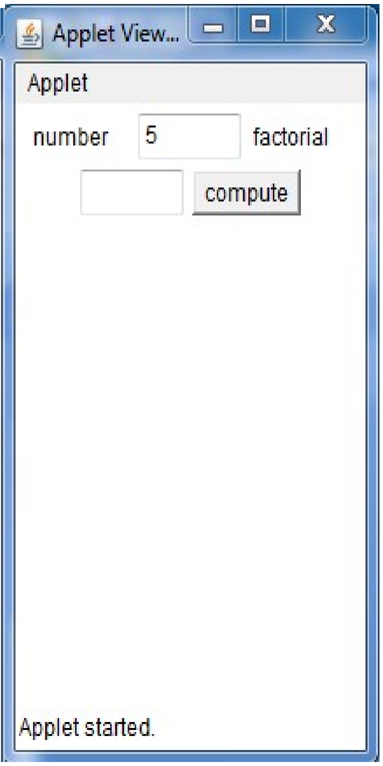
Output:



```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac Fact.java

C:\Users\anusha\Desktop>appletviewer Fact.java
```

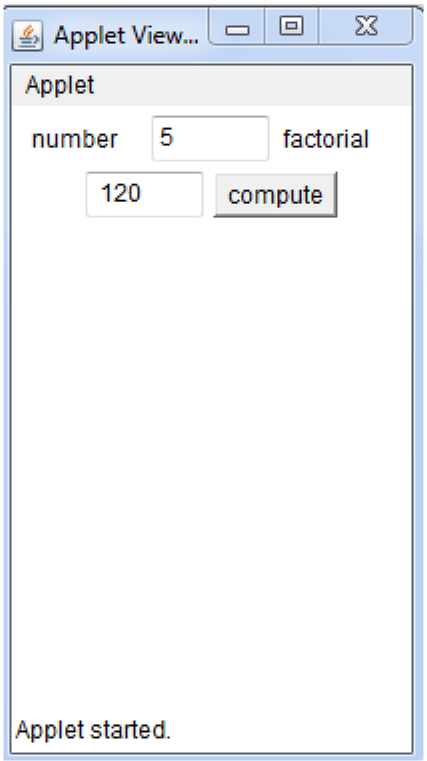


Applet started.



5

Applet started.



5

120

Applet started.

- Write a Java program that works as a simple calculator . Use a grid layout to arrange buttons for the digits and for the +,-,*,% operations. Add a text field to display the result.

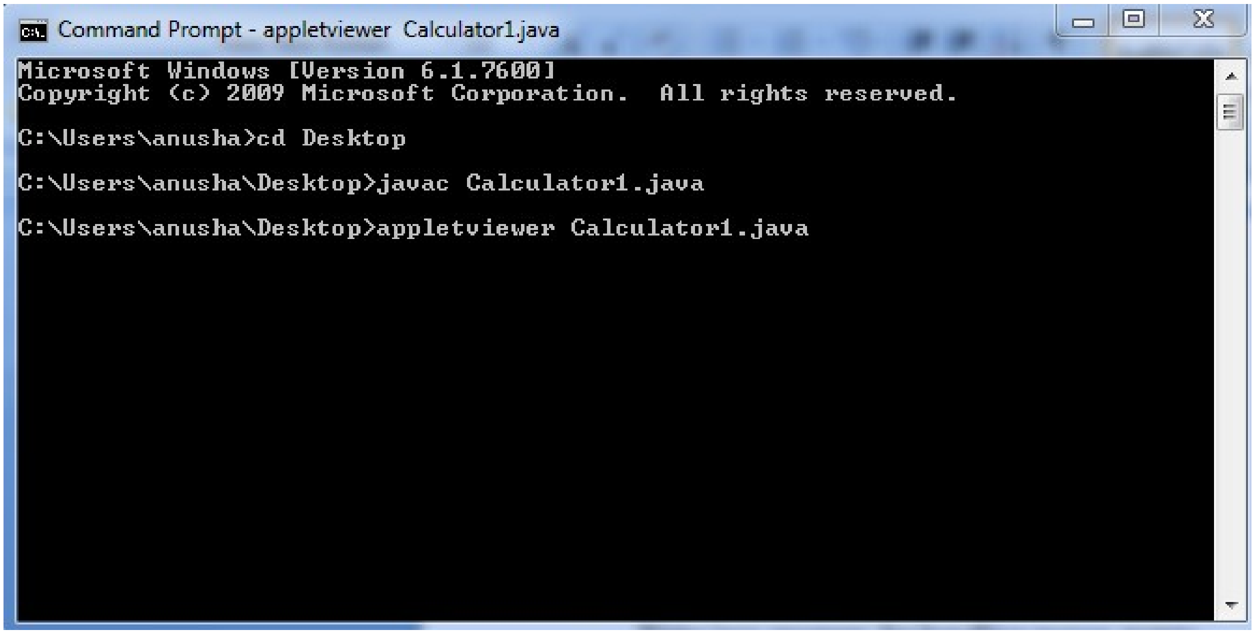
```
  //Program for implementing a Simple Calculator
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*<applet  code="Calculator1" width=300 height=300></applet>*/
public class Calculator1 extends Applet implements ActionListener
{
        TextField t;
        Button b[]=new Button[15];
        Button b1[]=new Button[6];
        String op2[]={"+","-","*","%","=","C"};
        String str1="";
        int p=0,q=0;
        String oper;
        public void init()
        {
                setLayout(new GridLayout(5,4));
                t=new TextField(20);
                setBackground(Color.pink);
                setFont(new Font("Arial",Font.BOLD,20));
                int k=0;
                t.setEditable(false);
                t.setBackground(Color.white);
                t.setText("0");
                for(int i=0;i<10;i++)
                {
                  b[i]=new Button(""+k);
                  add(b[i]);
                  k++;
```

```
          b[i].setBackground(Color.pink);
          b[i].addActionListener(this);
        }

        for(int i=0;i<6;i++)
        {
          b1[i]=new Button(""+op2[i]);
          add(b1[i]);
          b1[i].setBackground(Color.pink);
          b1[i].addActionListener(this);
        }
        add(t);
}
public void actionPerformed(ActionEvent ae)
{

        String  str=ae.getActionCommand();

        if(str.equals("+"))
                {
                        p=Integer.parseInt(t.getText());


             oper=str;
             t.setText(str1="");
                          }
        else if(str.equals("-"))
                {
                        p=Integer.parseInt(t.getText());
             oper=str;
             t.setText(str1="");
         }
```

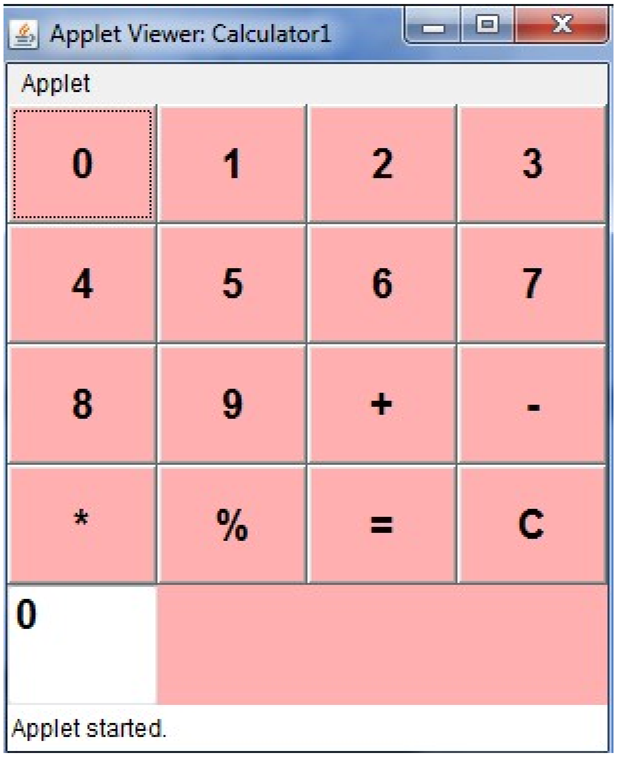
```
else if(str.equals("*"))
        {
                p=Integer.parseInt(t.getText());
     oper=str;
     t.setText(str1="");
}
else if(str.equals("%"))
        {
                p=Integer.parseInt(t.getText());
                   oper=str;


     t.setText(str1="");
 }
else if(str.equals("="))
                {
                 str1="";
                 if(oper.equals("+"))
                  {
          q=Integer.parseInt(t.getText());
          t.setText(String.valueOf((p+q)));
                  }

      else if(oper.equals("-"))
                         {
            q=Integer.parseInt(t.getText());
            t.setText(String.valueOf((p-q)));
                         }

      else  if(oper.equals("*"))
                          {
              q=Integer.parseInt(t.getText());
```

```
                              t.setText(String.valueOf((p*q)));
                                      }

                         else if(oper.equals("%"))
                                    {
                              q=Integer.parseInt(t.getText());
                              t.setText(String.valueOf((p%q)));
                                      }
                                   }

                     else if(str.equals("C"))
                                {
                                 p=0;q=0;
                          t.setText("");
                                    str1="";
                         t.setText("0");
                        }

                          else
                                {  t.setText(str1.concat(str));

                                 str1=t.getText();
                        }

                      }

        }
```

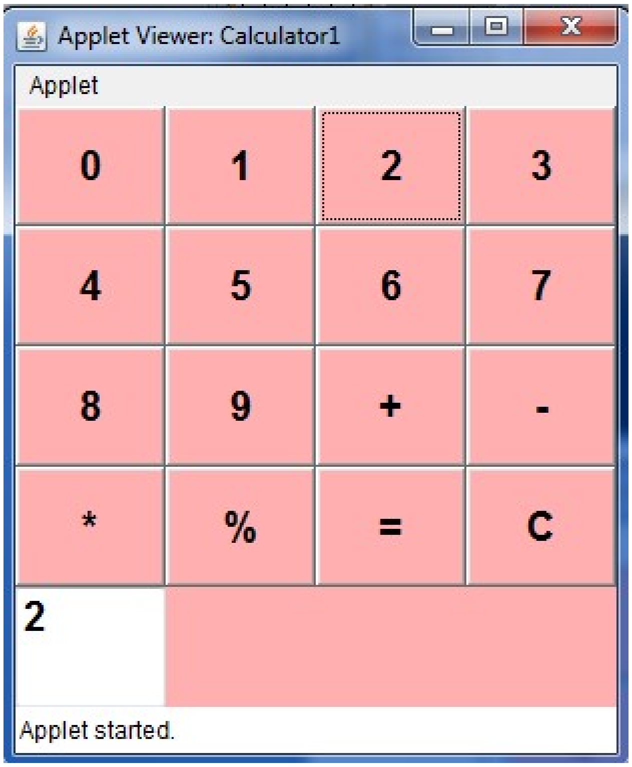
Output:



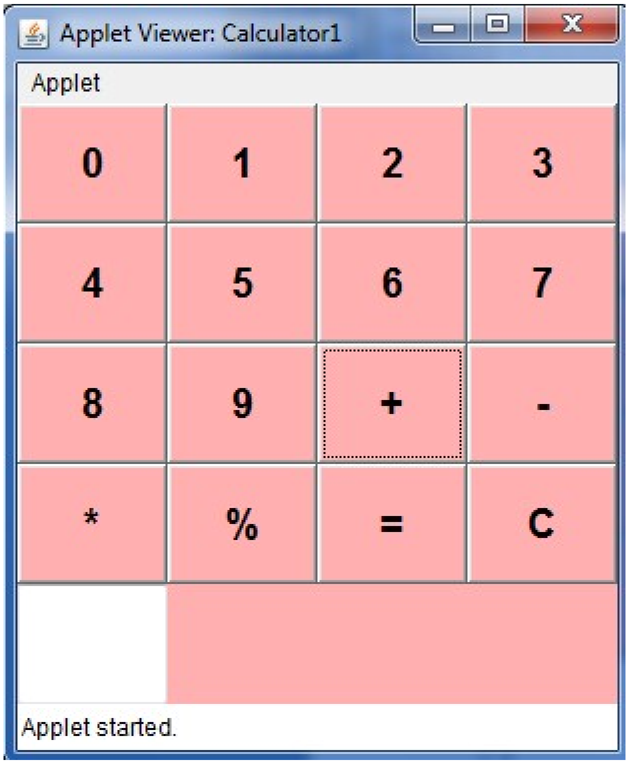
```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac Calculator1.java

C:\Users\anusha\Desktop>appletviewer Calculator1.java
```



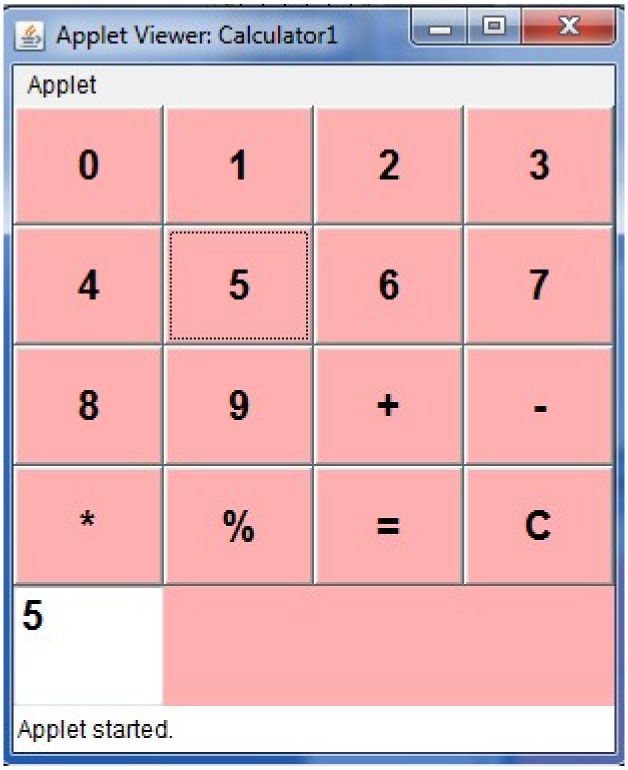
Applet started.



Applet started.

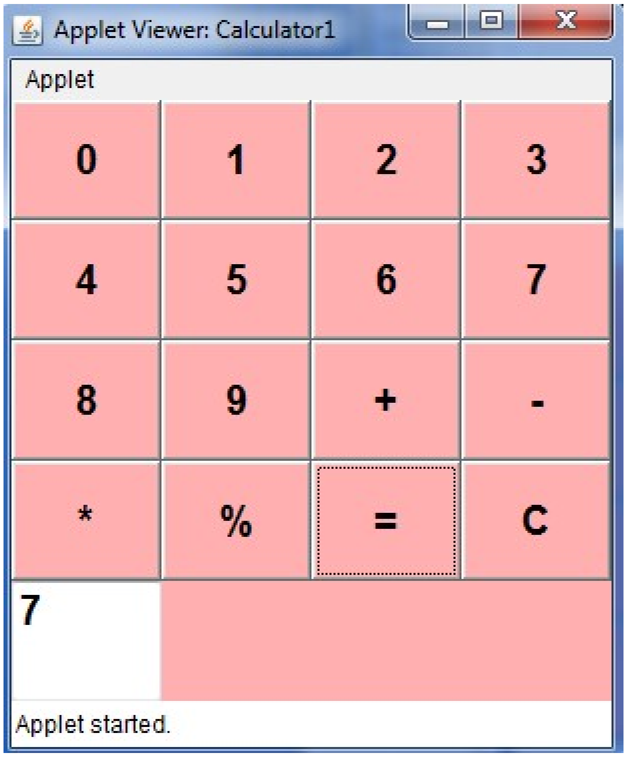


Applet started.



5

Applet started.



7

Applet started.

Week-10

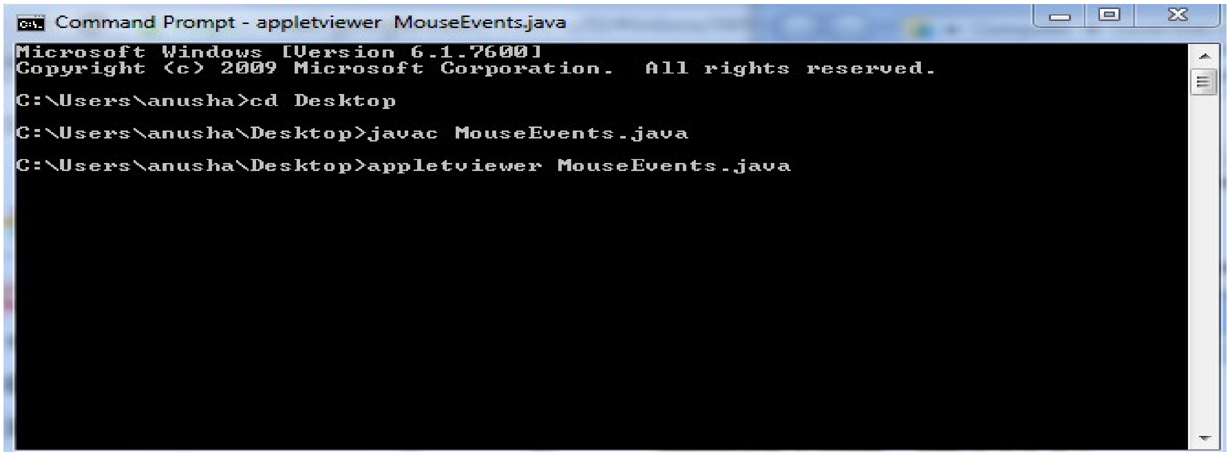
- Write java program for handling mouse events.

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="MouseEvents" width=500 height=500>
</applet>
*/
public class MouseEvents extends Applet implements MouseListener, MouseMotionListener
{
 String msg=" ";
 public void init()
 {
  addMouseListener(this);
  addMouseMotionListener(this);
  }

  public void mouseClicked(MouseEvent me)
  {
     msg="Mouse clicked.";
   repaint();
   }

   public void mouseEntered(MouseEvent me)
   {
     msg="mouse entered.";
   repaint();
   }

   public void mouseExited(MouseEvent me)
```

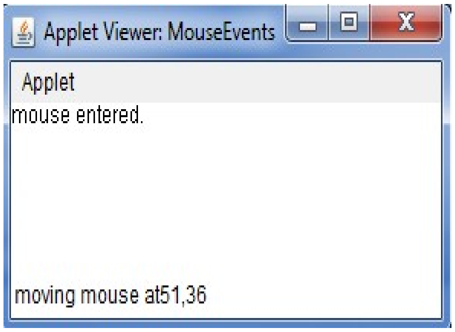
```
{
msg="mouse exited.";
repaint();
}

public void mousePressed(MouseEvent me)
{
  msg="Down";
repaint();
}

public void mouseReleased(MouseEvent me)
{
msg="up";
repaint();
}

public void mouseDragged(MouseEvent me)
{
msg="Dragged";
//showStatus("Dragging mouse at "+mouseX+","+mouseY);
repaint();
}

public void mouseMoved(MouseEvent me)
{
showStatus("moving mouse at”);
}

public void paint(Graphics g)
```

```
    {
    g.drawString(msg,40,50);
    }
}
```

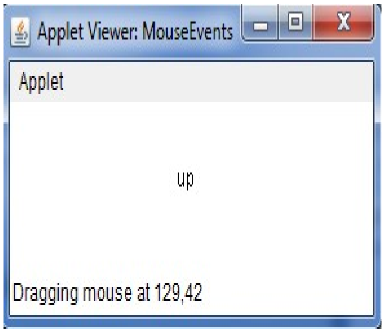
Output:



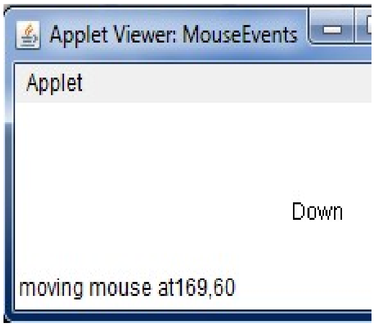
```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac MouseEvents.java

C:\Users\anusha\Desktop>appletviewer MouseEvents.java
```



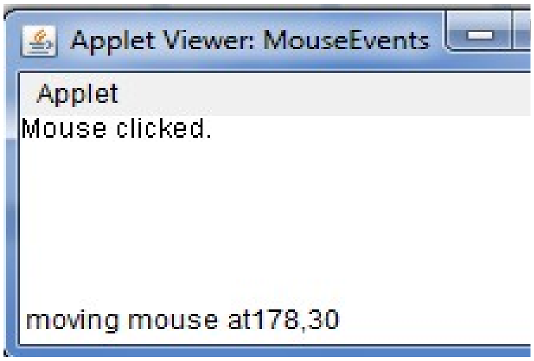
```
mouse entered.

moving mouse at51,36
```



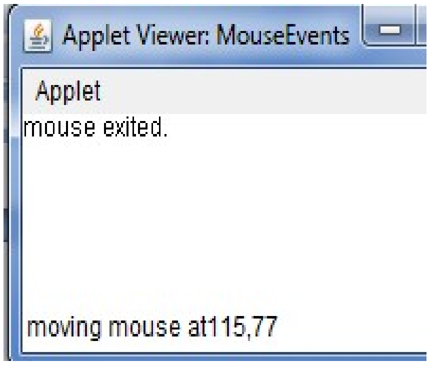
```
up

Dragging mouse at 129,42
```



```
Down

moving mouse at169,60
```



```
Mouse clicked.

moving mouse at178,30
```

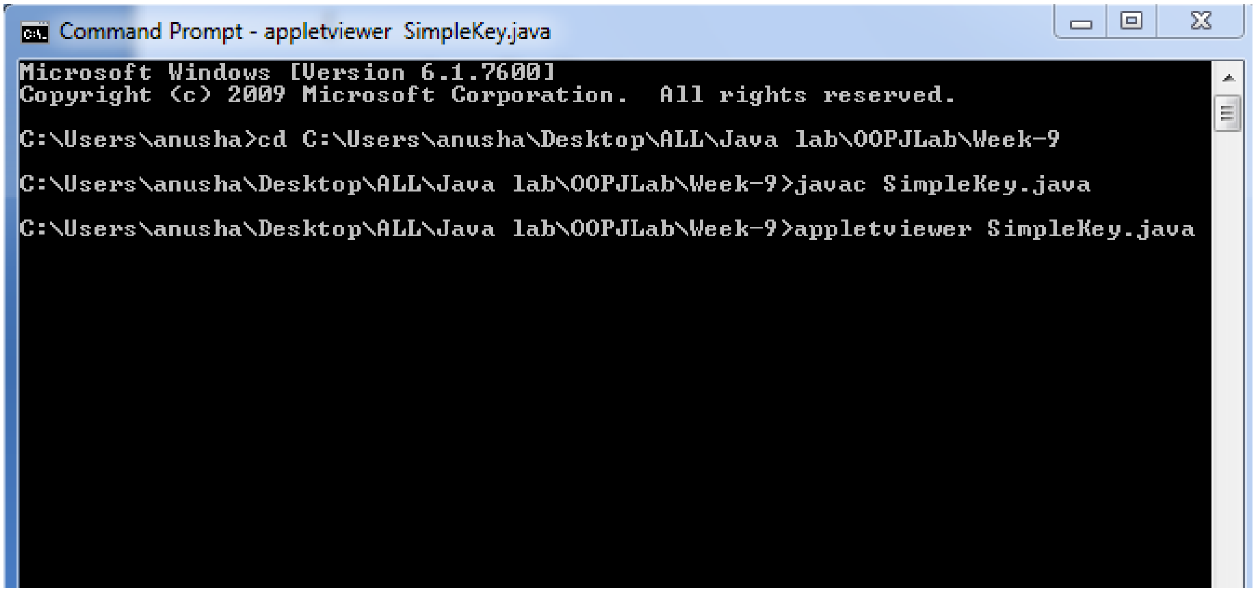


```
mouse exited.

moving mouse at115,77
```

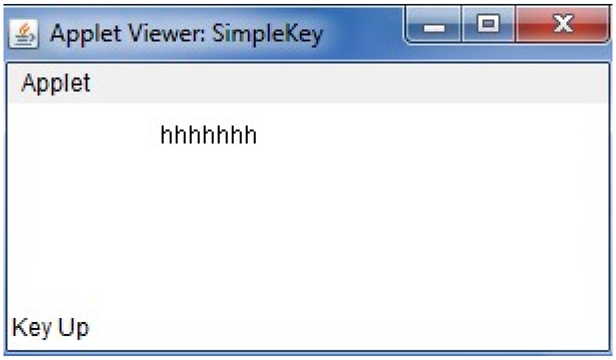
- Write java program for handling key events

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

/* <applet code="SimpleKey" width=300 height=100>
</applet> */

public class SimpleKey extends Applet implements KeyListener
{
public void init()
 {
  addKeyListener(this);
  requestFocus(); // request input focus
 }
 public void keyPressed(KeyEvent ke)
 {
  showStatus("Key Down");
 }
 public void keyReleased(KeyEvent ke)
 {
  showStatus("Key Up");
 }
 public void keyTyped(KeyEvent ke)
 {
  showStatus("Key typed");
}
}
```

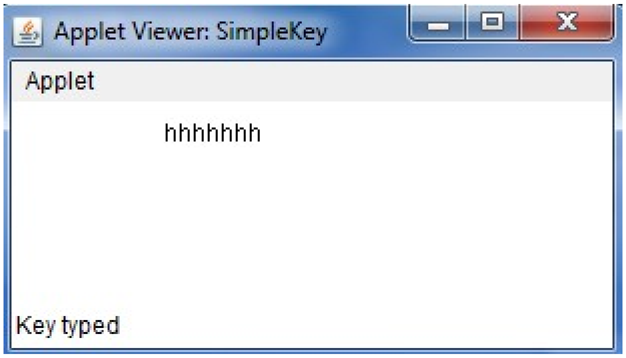
Output:



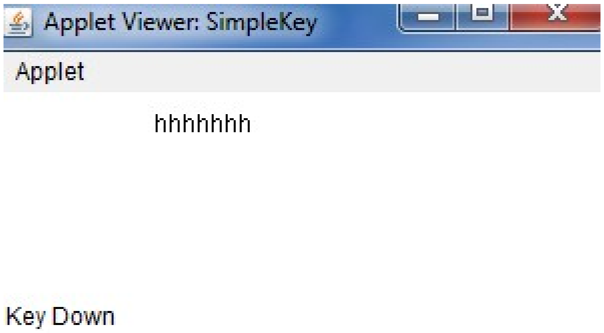
```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd C:\Users\anusha\Desktop\ALL\Java lab\OOPJLab\Week-9

C:\Users\anusha\Desktop\ALL\Java lab\OOPJLab\Week-9>javac SimpleKey.java

C:\Users\anusha\Desktop\ALL\Java lab\OOPJLab\Week-9>appletviewer SimpleKey.java
```



```
hhhhhhh

Key Up
```



```
hhhhhhh

Key typed
```



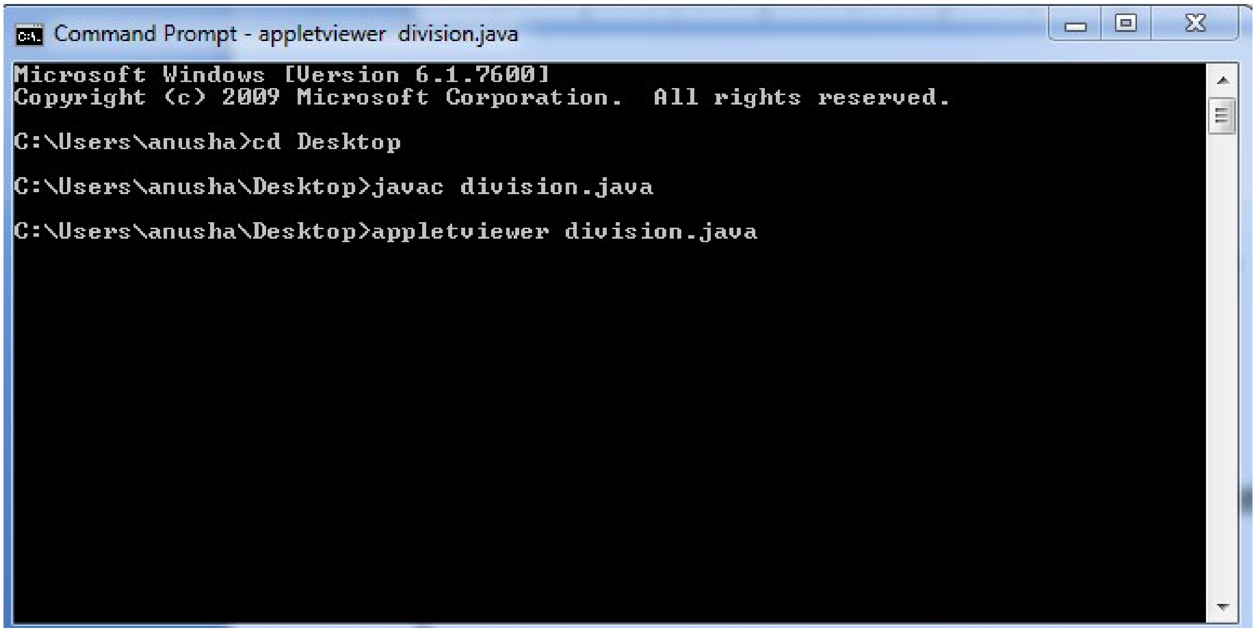
```
hhhhhhh

Key Down
```

Week-11

- Write a Program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields Num1 and Num2.The division of Num1 and Num2 is displayed in the result field when divide button is clicked. If Num1 or Num2 were not an integer , the program would through Number Format Exception. If Num2 were Zero , the program would through an Arithmetic Exception and display the exception in a message dialog box.

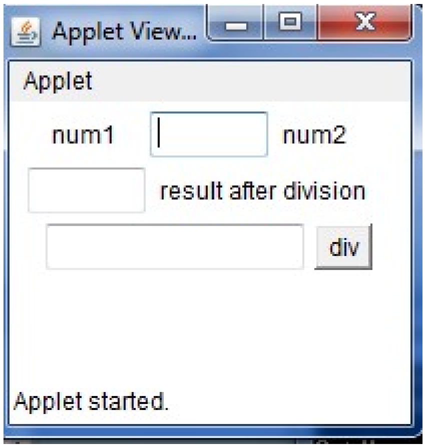
```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
/*<applet code="division.class" width=200 height=300>
</applet>*/
public class division extends Applet implements ActionListener
{
TextField x1,x2,x3;
public void init()
{
Button b=new Button("div");
Label L1=new Label("num1");
Label L2=new Label("num2");
Label L3=new Label("result after division");
x1=new TextField(5);
x2=new TextField(5);
x3=new TextField(15);
add(L1);
add(x1);
add(L2);
add(x2);
add(L3);
add(x3);
add(b);
x1.addActionListener(this);
x2.addActionListener(this);
x3.addActionListener(this);
b.addActionListener(this);
}
public void actionPerformed(ActionEvent j)
{
int m3=0;
String m1=x1.getText();
String m2=x2.getText();
try
{
m3=Integer.parseInt(m1)/Integer.parseInt(m2);
x3.setText(" "+m3);
}
catch(ArithmeticException e)
{
showStatus("divide by zero num2 should not be zero");
}
catch(NumberFormatException e)
{
//showStatus("Number Format Exception num1 n num2 must be integers");
x3.setText("plz enter a num");
}
}
}
```

Output:

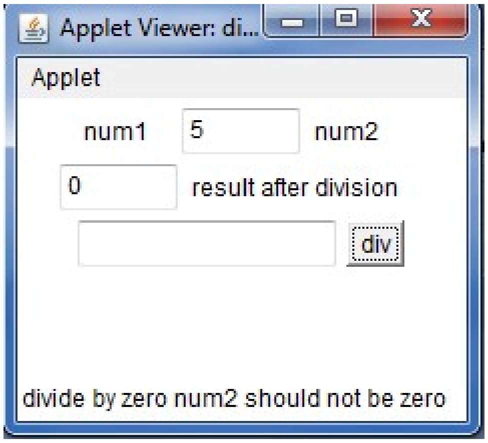


Command Prompt - appletviewer division.java

```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac division.java

C:\Users\anusha\Desktop>appletviewer division.java
```



Applet started.



num1 5 num2

0 result after division

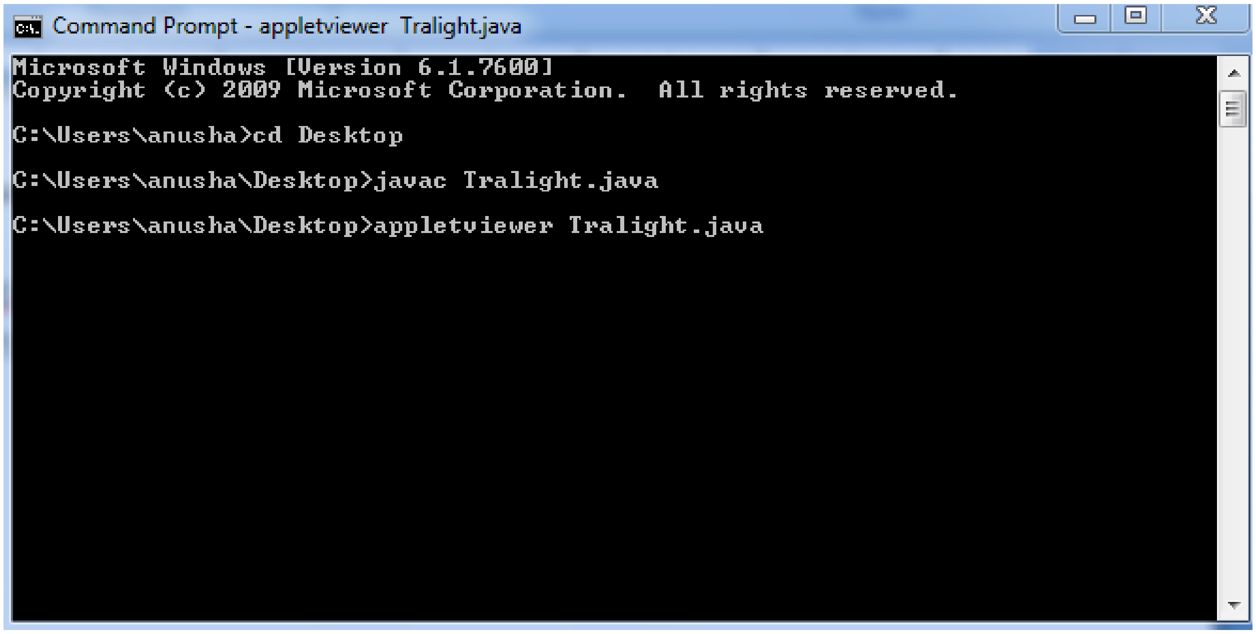
divide by zero num2 should not be zero

Week-12

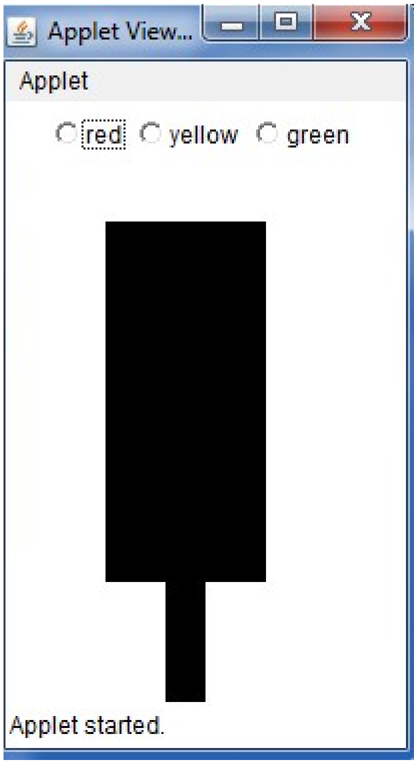
Write a java program that simulates traffic light .The program lets the user select one of three lights : red,yellow or green.When a radio button is selected , the light is turned on , and only one light can be on at a time and No light is ON when program starts.

```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;

/*<applet code="Tralight" width=200 height=300>
</applet>*/

public class Tralight extends Applet implements ItemListener
{
Checkbox c1,c2,c3;
public void init()
{
CheckboxGroup x1=new CheckboxGroup();
c1=new Checkbox("red",x1,false);
c2=new Checkbox("yellow",x1,false);
c3=new Checkbox("green",x1,false);
add(c1);
add(c2);
add(c3);
c1.addItemListener(this);
c2.addItemListener(this);
c3.addItemListener(this);
}
public void itemStateChanged(ItemEvent e)
{
repaint();
}
public void paint(Graphics g)
{
g.fillRect(50,60,80,180);
g.fillRect(80,240,20,130);

if(c1.getState()==true)
{
g.setColor(Color.red);
g.fillOval(75,80,35,35);
}
if(c3.getState()==true)
{
g.setColor(Color.green);
g.fillOval(75,180,35,35);
}
if(c2.getState()==true)
{
g.setColor(Color.yellow);
g.fillOval(75,130,35,35);
}
}
}
```

Output:

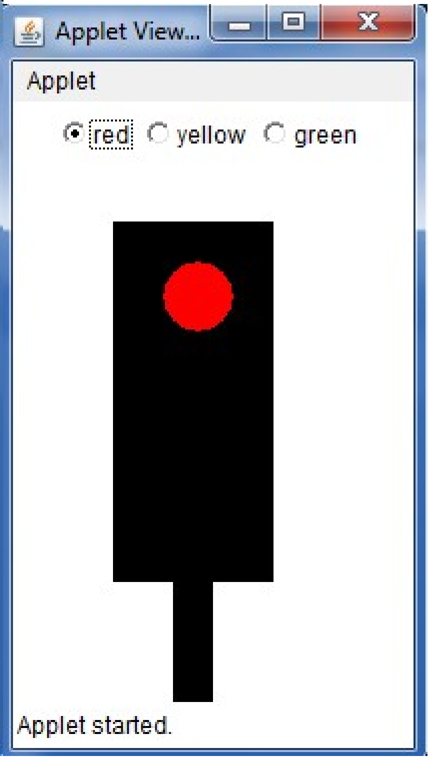


```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac Tralight.java

C:\Users\anusha\Desktop>appletviewer Tralight.java
```



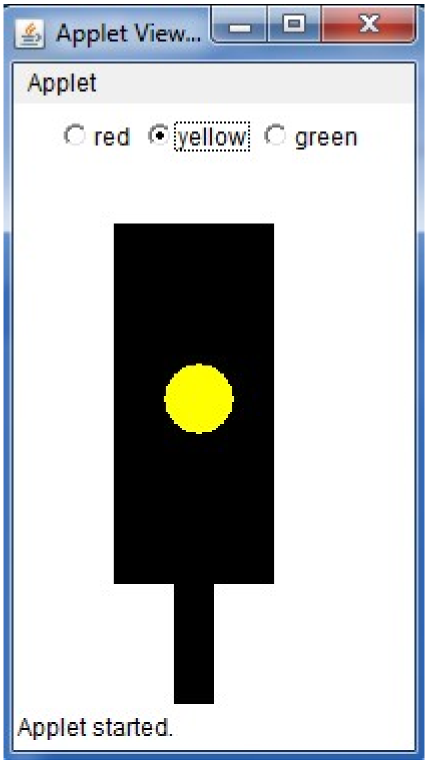
○ red ○ yellow ○ green

Applet started.



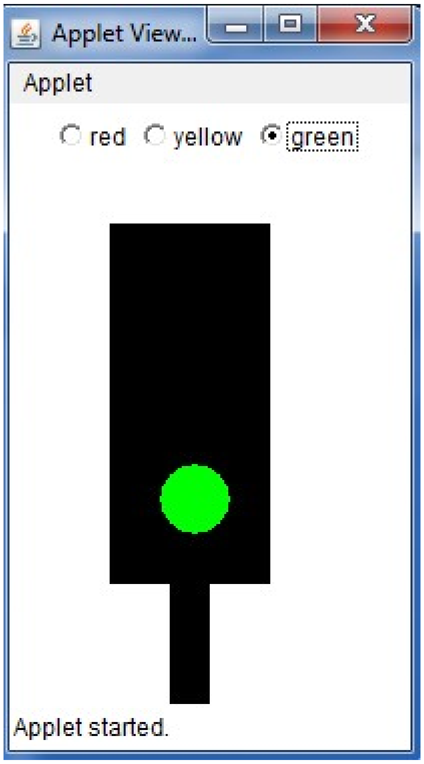
● red ○ yellow ○ green

Applet started.



○ red ● yellow ○ green

Applet started.



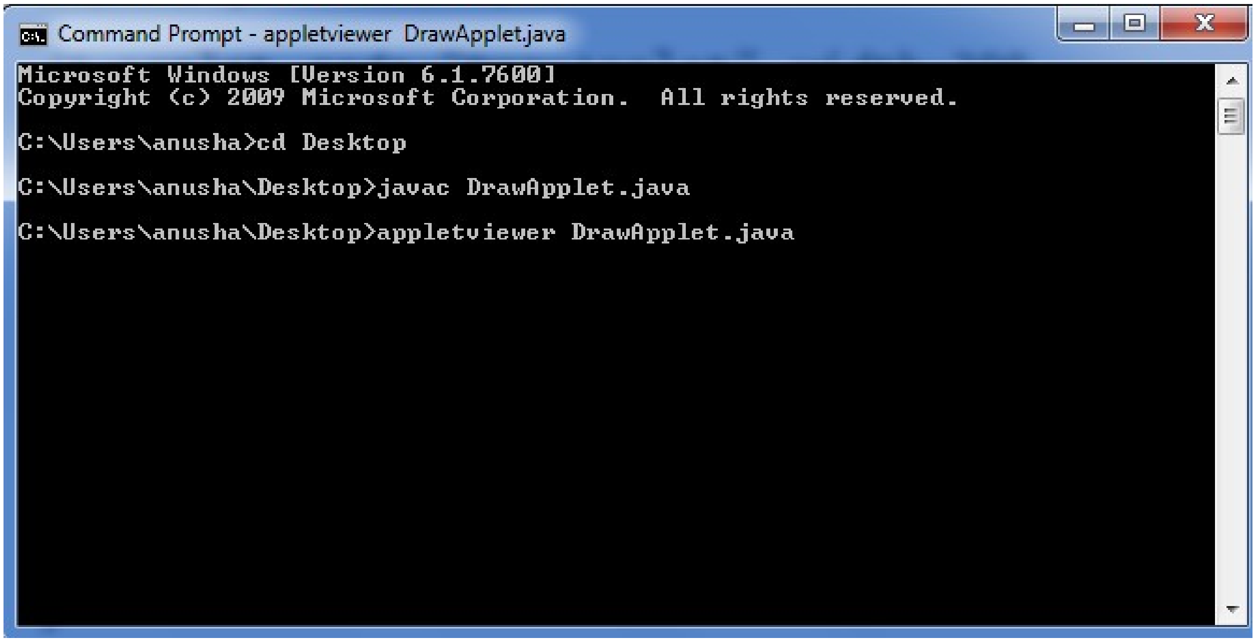
○ red ○ yellow ● green

Applet started.

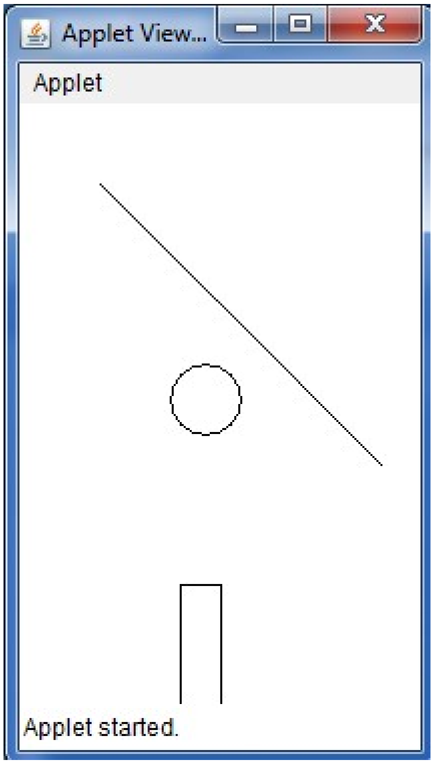
Write a java program that allows the user to draw lines , rectangles and ovals.

```
import java.applet.*;
import java.awt.*;

/*<applet code="DrawApplet" width=200 height=300>
</applet>*/

public class DrawApplet extends Applet
{
 public void paint(Graphics g)
 {
  g.drawLine(40,40,180,180);
  g.drawRect(80,240,20,130);
  g.drawOval(75,130,35,35);
 }
}
```

Output:



```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac DrawApplet.java

C:\Users\anusha\Desktop>appletviewer DrawApplet.java
```



Applet started.

Week-13

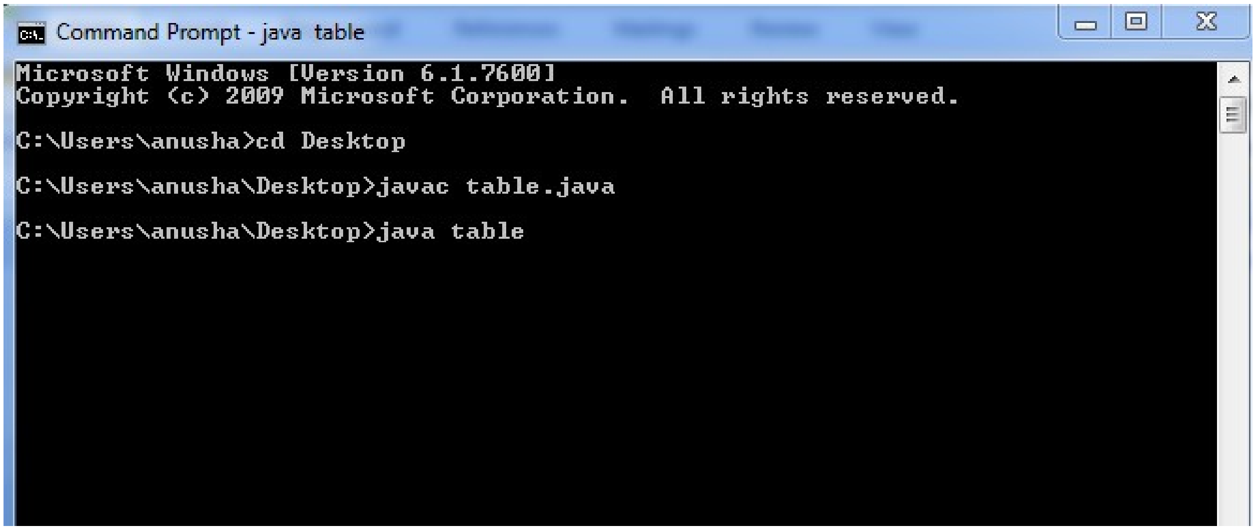
Create a Table.txt file such that the first line in the file is the header , and the remaining lines correspond to rows in the table.The elements are separated by commas.Write a java program to display the table using JTable components.

```
import java.io.*;
import java.awt.*;
import javax.swing.*;
import java.util.*;
class table extends JFrame
{
static int r=0,i=0,j=0,c=0;
static Object d[][];
static Object h[];
table()
{
setVisible(true);
setSize(500,500);
Label l=new Label("GOkaraju Institue of Technology");
setTitle("TABLE DEMO");
setLayout(new FlowLayout(FlowLayout.CENTER));
JTable j=new JTable(d,h);
JScrollPane jsp=new JScrollPane(j);
add(l); add(jsp);
}
public static void main(String[] args) throws IOException
{
FileReader f=new FileReader("file.txt");
BufferedReader b=new BufferedReader(f);
String s=b.readLine();
StringTokenizer g=new StringTokenizer(s,",");
c=g.countTokens();
h=new Object[c];
while(g.hasMoreTokens())
{
h[i]=g.nextToken();
i++;
}
while((s=b.readLine())!=null)
r++;
d=new Object[r][c];
BufferedReader br=new BufferedReader(new FileReader("file.txt"));
String e=br.readLine();
for(i=0;i<r;i++)
{
e=br.readLine();
StringTokenizer st=new StringTokenizer(e,",");
while(st.hasMoreTokens())
{
d[i][j]=st.nextToken();
j++;
}
j=0;
}
new table();
}
}
```

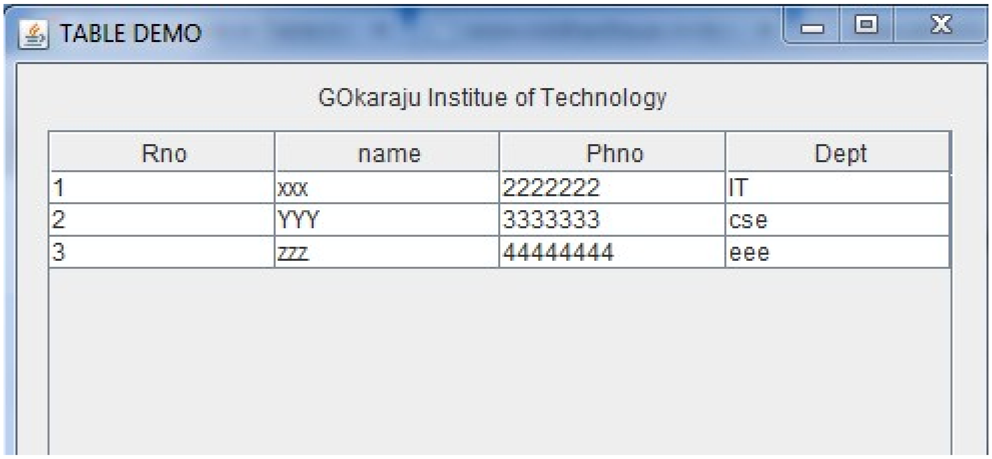
Student.txt:

```
Rno,name,Phno,Dept
1,xxx,2222222,IT
2,YYY,3333333,cse
3,zzz,44444444,eee
```

Output:



```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\anusha>cd Desktop

C:\Users\anusha\Desktop>javac table.java

C:\Users\anusha\Desktop>java table
```



GOkaraju Institue of Technology

Rno	name	Phno	Dept
1	xxx	2222222	IT
2	YYY	3333333	cse
3	zzz	44444444	eee