**EXTRA QUESTIONS JAVA PRACTICAL**

***Q1. Write a program to print Hello World!.***

***Solution:***

**public** **class** MyClass {

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

// **TODO** Auto-generated method stub

System.***out***.print("Hello World!");

}

}

**Output:**

Hello World!

***Q2.Write a program to implement factorial of a given number.***

***Solution:***

**import** java.io.\*;

**public** **class** sum {

**public** **static** **void** main(String[] args)**throws** IOException {

// **TODO** Auto-generated method stub

**int** n,fact =1;

BufferedReader br2= **new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("Enter the value of n");

n=Integer.*parseInt*(br2.readLine());

**for**(**int** i=1;i<=n;i++)

fact=fact\*i;

System.***out***.println("Factorial of a given number is"+fact);

}

}

**Output:**

Enter the value of n

7

Factorial of a given number is5040

***Q3.Write a program to implement the series 1+ 1/2 + 1/3……1/n.***

***solution:***

**import** java.io.\*;

**public** **class** series {

**public** **static** **void** main(String[] args) **throws** IOException {

// **TODO** Auto-generated method stub

**int** n;

BufferedReader br1=**new** BufferedReader(**new** InputStreamReader(System.***in***));

**double** s=0;

System.***out***.println("Enter the value of n");

n=Integer.*parseInt*(br1.readLine());

**for**(**int** i=1;i<=n;i++)

s=s+1.0/i;

System.***out***.println("series is"+s);

}

}

**Output:**

Enter the value of n

6

series is2.4499999999999997

***Q4.Write a program to implement Fibonacci series of a given number.***

***Solution:***

**import** java.io.\*;

**public** **class** Fibonacci {

**public** **static** **void** main(String[] args) **throws** IOException {

// **TODO** Auto-generated method stub

**int** n,a=0,b=1,sum=0;

BufferedReader br2= **new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("Enter the value of n");

n=Integer.*parseInt*(br2.readLine());

**for**(**int** i=1;i<=n;i++)

{

System.***out***.println(a);

sum=a+b;

a=b;

b=sum;

}

}

}

**Output:**

Enter the value of n

7

0

1

1

2

3

5

8

***Q5.Write a program to print odd number up to a given n number.***

***Solution:***

**import** java.io.\*;

**public** **class** oddnumber {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

**int** i,n;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("Enter the value of n");

n=Integer.*parseInt*(br.readLine());

**for**(i=1;i<=n;i++)

{

System.***out***.println("odd number is"+i);

i=i+1;

}

}

}

**Output:**

Enter the value of n

20

odd number is1

odd number is3

odd number is5

odd number is7

odd number is9

odd number is11

odd number is13

odd number is15

odd number is17

odd number is19

***Q6.Write a program to implement maximum number from n given number.***

***Solution:***

import java.io.\*;

**public** **class** maximum {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

**int** n, max=-1;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("Enter the value of n");

n=Integer.*parseInt*(br.readLine());

**int** a[]=**new** **int**[n];

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

{

System.***out***.println("Enter the elements of array");

a[i]=Integer.*parseInt*(br.readLine());

}

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

{

**if**(max<a[i])

{

max=a[i];

}

}

System.***out***.println("Maximum value in the array is"+max);

}

}

**Output:**

Enter the value of n

7

Enter the elements of array

34

Enter the elements of array

78

Enter the elements of array

67

Enter the elements of array

89

Enter the elements of array

60

Enter the elements of array

234

Enter the elements of array

678

Maximum value in the array is678

***Q7.Write a program to implement GCD of a given number.***

***Solution:***

**import** java.io.\*;

**public** **class** GCD {

**public** **static** **void** main(String[] args) **throws** IOException {

// **TODO** Auto-generated method stub

**int** x,y,rem;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("Enter the value of x");

x=Integer.*parseInt*(br.readLine());

System.***out***.println("Enter the value of y");

y=Integer.*parseInt*(br.readLine());

rem=x%y;

**while**(rem!=0)

{

x=y;

y=rem;

rem=x%y;

}

**if**(rem==0)

System.***out***.println("GCD is"+y);

}

}

**Output:**

Enter the value of x

678

Enter the value of y

890

GCD is2

***Q8.Write a program to print an array.***

***Solution:***

**import** java.io.\*;

**public** **class** array {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

**int** a[]=**new** **int** [10];

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("Enter the elements of array ");

**for**(**int** i=0;i<5;i++)

{

a[i]=Integer.*parseInt*(br.readLine());

}

System.***out***.println("Array is");

**for**(**int** i=0;i<5;i++)

{

System.***out***.println(a[i]);

}

}

}

**Output:**

Enter the elements of array

6

7

8

9

67

Array is

6

7

8

9

67

***Q9.Write a program to implement binary search.***

***Solution:***

**import** java.io.\*;

**class** binary{

**private** **int** a[];

**int** size;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

**public** binary(**int** k)

{

size=k;

a=**new** **int**[k];

}

**void** input()**throws** IOException

{

System.***out***.println("Enter the elements of an array");

**for**(**int** i=0;i<**this**.size;i++)

**this**.a[i]=Integer.*parseInt*(br.readLine());

}

**void** search()**throws** IOException

{

**int** mid=0;

**int** low=0;

**int** high=size-1;

**int** x;

System.***out***.println("Enter the element to be search");

x=Integer.*parseInt*(br.readLine());

**while**(low<=high)

{

mid=(low+high)/2;

**if**(a[mid]<x)

low=mid+1;

**else** **if**(a[mid]==x)

{

System.***out***.println("Element is found at position"+(mid+1));

**break**;

}

**else** {

high=mid-1;

}

}

**if**(low>high)

{

System.***out***.println("Element is not found");

}

}

}

**public** **class** Binarysearch {

**public** **static** **void** main(String[] args)**throws** IOException {

// **TODO** Auto-generated method stub

binary o1=**new** binary(6);

o1.input();

o1.search();

}

}

**Output:**

Enter the elements of an array

7

8

9

17

4

90

Enter the element to be search

90

Element is found at position6

***Q10.Write a program to implement linear search.***

***Solution:***

**import** java.io.\*;

**class** linear

{

**int** a[];

**int** size;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

linear(**int** k)

{

size=k;

a=**new** **int**[k];

}

**void** input()**throws** IOException

{

System.***out***.println("Enter the elements of an array");

**for**(**int** i=0;i<**this**.size;i++)

**this**.a[i]=Integer.*parseInt*(br.readLine());

}

**void** search()**throws** IOException{

**int** i,x;

System.***out***.println("Enter the element to be search");

x=Integer.*parseInt*(br.readLine());

**for**(i=0;i<**this**.size;i++)

**if**(**this**.a[i]==x)

{

System.***out***.println("Element found at position"+(i+1));

**break**;

}

**if**(i==size)

System.***out***.println("Element not found");

}

}

**public** **class** Linearsearch {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

linear o1=**new** linear(8);

o1.input();

o1.search();

}

}

**Output:**

Enter the elements of an array

7

8

9

17

34

54

56

78

Enter the element to be search

78

Element found at position8

***Q11.Write a program to remove duplicate elements from an array.***

***Solution:***

**import** java.io.\*;

**public** **class** Duplicates {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("Enter the size of array");

**int** n;

n=Integer.*parseInt*(br.readLine());

**int** a[];

a=**new** **int**[n];

System.***out***.println("Enter the elements of an array");

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

a[i]=Integer.*parseInt*(br.readLine());

**for**(**int** i=0;i<n-1;i++)

**for**(**int** j=i+1;j<n;j++)

**if**(a[i]==a[j])

{

**if**(j==n-1)//j is the last position

n=n-1;

**else** {

**for**(**int** k=j;k<n-1;k++)

a[k]=a[k+1];

n=n-1;

}

}

System.***out***.println("Elements after removing duplicates");

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

System.***out***.println(a[i]);

}

}

**Output:**

Enter the size of array

6

Enter the elements of an array

1

2

3

2

4

2

Elements after removing duplicates

1

2

3

4

***Q12.Write a program to implement all the arithmetic operations.***

***Solution***

**import** java.io.\*;

**public** **class** Arithmeticoperations {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

**int** x,y,z;

**int** x1,y1,z1;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

x=Integer.*parseInt*(br.readLine());

System.***out***.println("Enter the value of x "+x);

y=Integer.*parseInt*(br.readLine());

System.***out***.println("Enter the value of y "+y);

System.***out***.println("value of x "+x);

System.***out***.println("Value of y "+y);

z=x+y;

System.***out***.println("Value of z "+z);

x1=x-y;

System.***out***.println("Value of x1 "+x1);

y1=x\*y;

System.***out***.println("Value of y1 "+y1);

z1=x/y;

System.***out***.println("Value of z1 "+z1);

}

}

**Output:**

45

Enter the value of x 45

67

Enter the value of y 67

Value of x 45

Value of y 67

Value of z 112

Value of x1 -22

Value of y1 3015

Value of z1 0

***Q13.Write a program Create a class fraction by using constructor overloading.***

***Solution***

**import** java.io.\*;

**class** fract

{

**int** n,d;

**public** fract()

{

n=0;

d=0;

}

fract(**int** i,**int** j)

{

n=i;

d=j;

}

fract(fract o1)

{

n=o1.n;

d=o1.d;

}

**void** store(**int** i,**int** j)

{

n=i;

d=j;

}

**void** display()

{

System.***out***.println(n+"/"+d);

}

**public** String toString()

{

**return**(n+"/"+ d);

}

}

**public** **class** Fraction {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

**int** x,y;

fract o1,o2;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("parameterless constructor +o1");

o1=**new** fract();

o1.display();

System.***out***.println("enter the numerator and demoertor");

x=Integer.*parseInt*(br.readLine());

y=Integer.*parseInt*(br.readLine());

o2=**new** fract(x,y);

o2.display();

fract o3;

System.***out***.println("parameter constructor +o3");

o3=**new** fract(6,88);

o3.display();

fract o5=**new** fract(o3);

System.***out***.println("copy constructor +o5");

o5.display();

System.***out***.println("tostring function");

System.***out***.println(o5.toString());

}

}

**Output:**

parameterless constructor +o1

0/0

enter the numerator and demoertor

90

6

90/6

parameter constructor +o3

45/89

copy constructor +o5

45/89

tostring function

45/89

***Q14****.****Write a program to create factorial class.***

***Solution***

**import** java.io.\*;

**class** factorial{

**int** fact(**int** n) {

**if**(n==1)

**return** 1;

**else**

**return** n\*fact(n-1);

}

}

**public** **class** fact {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

**int** n;

System.***out***.println("Enter the value for computing factorial");

n=Integer.*parseInt*(br.readLine());

factorial o1;

o1=**new** factorial();

**int** x=o1.fact(n);

System.***out***.println("The factorial is"+x);

}

}

**Output:**

Enter the value for computing factorial

5

The factorial is120

***Q15.Write a program to create class box.***

***Solution:***

**Import java**.io.\*;

**class** box

{

**private** **double** w;

**double** h;

**double** d;

**public** box()

{

w=10;

h=10;

d=10;

}

box(box o1)

{

w=o1.w;

h=o1.h;

d=o1.d;

}

box(**double** i,**double** j,**double** k)

{

w=i;

h=j;

d=k;

}

**double** volume()

{

**return** w\*h\*d;

}

}

**public** **class** Boxdemo {

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

// **TODO** Auto-generated method stub

box o1,o2,o3;

o1=**new** box();

**double** vol;

vol=o1.volume();

System.***out***.println("Volume of o1 is"+vol);

o2=**new** box(8,9,6);

vol=o2.volume();

System.***out***.println("Volume of o2 is"+vol);

o3=**new** box(o2);

vol=o3.volume();

System.***out***.println("Volume of o3 is"+vol);

}

}

**Output:**

Volume of o1 is1000.0

Volume of o2 is432.0

Volume of o3 is432.0

***Q16.Write a program to implement class stack.***

***Solution:***

**import** java.io.\*;

**class** stacktest

{

**int** a[],size;

**int** tos;

stacktest() {

size =10;

a=**new** **int**[10];

tos=-1;

}

stacktest(**int** k)

{

a=**new** **int**[k];

size=k;

tos=-1;

}

**void** push(**int** x)

{

**if**(tos==9)

System.***out***.println("stack is overflow:");

**else**

a[++tos]=x;

}

**int** pop()

{

**if**(tos<0) {

System.***out***.println("stack is underflow:");

**return** 0;

}

**else**

**return** a[tos--];

}

**void** display()

{

**for**(**int** i=tos;i>=0;i--)

{

System.***out***.println(a[i]);

}

}

}

**public** **class** Stack {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

stacktest o1=**new** stacktest(6);

**int** x;

String ch1;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

**do**

{

System.***out***.println("1.Push operation:");

System.***out***.println("2.Pop operation:");

System.***out***.println("3.Display operation:");

System.***out***.println("Enter your choice:");

**int** ch=Integer.*parseInt*(br.readLine());

**switch**(ch)

{

**case** 1: System.***out***.println("Enter the value to push");

x=Integer.*parseInt*(br.readLine());

o1.push(x);

**break**;

**case** 2: x=o1.pop();

System.***out***.println("popped element is"+x);

**break**;

**case** 3: o1.display();

}

System.***out***.println("Do you want to continue");

ch1=br.readLine();

}**while**(ch1.equals("y"));

}

}

**Output:**

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

1

Enter the value to push

34

Do you want to continue

y

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

1

Enter the value to push

90

Do you want to continue

y

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

1

Enter the value to push

89

Do you want to continue

y

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

2

popped element is89

Do you want to continue

y

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

3

90

34

Do you want to continue

N

***Q17.Write a program to create a matrix class.***

***Solution***

**import** java.io.\*;

**class** matrix1{

**private** **int** a[][];

**private** **int** rows,cols;

matrix1()

{

rows=2;

cols=2;

a=**new** **int**[rows][cols];

}

matrix1(**int** i,**int** j)

{

rows=i;

cols=j;

a=**new** **int**[rows][cols];

}

**int** getrows() {**return** rows;}

**int** getcols() {**return** cols;}

**void** input()**throws** IOException

{

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("enter the rows and cols of the matrix");

rows=Integer.*parseInt*(br.readLine());

cols=Integer.*parseInt*(br.readLine());

a=**new** **int**[rows][cols];

System.***out***.println("enter the elements of matrix");

**for**(**int** i=0;i<rows;i++)

**for**(**int** j=0;j<cols;j++)

a[i][j]=Integer.*parseInt*(br.readLine());

}

**void** display()

{

**for**(**int** i=0;i<rows;i++) {

**for**(**int** j=0;j<cols;j++)

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

System.***out***.println();

}

}

matrix1 add(matrix1 o1)

{

matrix1 temp;

temp=**new** matrix1(o1.rows,o1.cols);

**for**(**int** i=0;i<rows;i++)

**for**(**int** j=0;j<cols;j++)

temp.a[i][j]=a[i][j]+o1.a[i][j];

**return** temp;

}

matrix1 multiplication(matrix1 o1)

{

matrix1 temp;

temp=**new** matrix1(rows,o1.cols);

**for**(**int** i=0;i<rows;i++)

**for**(**int** j=0;j<o1.cols;j++) {

temp.a[i][j]=0;

**for**(**int** k=0;k<o1.rows;k++)

temp.a[i][j]=temp.a[i][j]+a[i][k]\*o1.a[k][j];

}

**return** temp;

}

matrix1 tranpose(matrix1 o1)

{

matrix1 temp;

temp=**new** matrix1(o1.rows,o1.cols);

**for**(**int** i=0;i<rows;i++)

**for**(**int** j=0;j<cols;j++)

temp.a[i][j]=o1.a[j][i];

**return** temp;

}

matrix1 subtraction(matrix1 o1)

{

matrix1 temp;

temp=**new** matrix1(o1.rows,o1.cols);

**for**(**int** i=0;i<rows;i++)

**for**(**int** j=0;j<cols;j++)

temp.a[i][j]=a[i][j]-o1.a[i][j];

**return** temp;

}

}

**public** **class** matrice {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

matrix1 o1=**new** matrix1();

o1.input();

matrix1 o2=**new** matrix1();

o2.input();

matrix1 o3=**new** matrix1();

String ch1;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

**do** {

System.***out***.println("1.Addition:");

System.***out***.println("2.Subtraction:");

System.***out***.println("3.Multiplication:");

System.***out***.println("4.Tranpose:");

System.***out***.println("5.display");

System.***out***.println("enter your choice:");

**int** ch=Integer.*parseInt*(br.readLine());

**switch**(ch) {

**case** 1 : **if**(o1.getrows()==o2.getrows() && o1.getcols()==o2.getcols() )

o3=o1.add(o2);

**else**

System.***out***.println("matrices order not equal and cannot be added");

**break**;

**case** 2:

**if**(o1.getrows()==o2.getrows() && o1.getcols()==o2.getcols() )

o3=o1.subtraction(o2);

**else**

System.***out***.println("matrices order not equal and cannot be subtracted");

**break**;

**case** 3:

**if**(o1.getcols()!=o2.getrows())

System.***out***.println("matrix cannot be multiplied");

**else**

o3=o2.multiplication(o1);

**break**;

**case** 4:

o3=o2.tranpose(o1);

**break**;

**case** 5:System.***out***.println("first matrix:");

o1.display();

System.***out***.println("second matrix:");

o2.display();

System.***out***.println("resulting matrix:");

o3.display();

**break**;

}

System.***out***.println("Do you want to continue");

ch1=br.readLine();

}**while**(ch1.equals("y"));

}

}

**Output:**

enter the rows and cols of the matrix

2

2

enter the elements of matrix

1

2

3

4

enter the rows and cols of the matrix

2

2

enter the elements of matrix

1

2

3

4

1.Addition:

2.Subtraction:

3.Multiplication:

4.Tranpose:

5.display

enter your choice:

1

Do you want to continue

y

1.Addition:

2.Subtraction:

3.Multiplication:

4.Tranpose:

5.display

enter your choice:

5

first matrix:

1 2

3 4

second matrix:

1 2

3 4

resulting matrix:

2 4

6 8

Do you want to continue

y

1.Addition:

2.Subtraction:

3.Multiplication:

4.Tranpose:

5.display

enter your choice:

2

Do you want to continue

y

1.Addition:

2.Subtraction:

3.Multiplication:

4.Tranpose:

5.display

enter your choice:

5

first matrix:

1 2

3 4

second matrix:

1 2

3 4

resulting matrix:

0 0

0 0

Do you want to continue

y

1.Addition:

2.Subtraction:

3.Multiplication:

4.Tranpose:

5.display

enter your choice:

3

Do you want to continue

y

1.Addition:

2.Subtraction:

3.Multiplication:

4.Tranpose:

5.display

enter your choice:

5

first matrix:

1 2

3 4

second matrix:

1 2

3 4

resulting matrix:

7 10

15 22

Do you want to continue

y

1.Addition:

2.Subtraction:

3.Multiplication:

4.Tranpose:

5.display

enter your choice:

4

Do you want to continue

y

1.Addition:

2.Subtraction:

3.Multiplication:

4.Tranpose:

5.display

enter your choice:

5

first matrix:

1 2

3 4

second matrix:

1 2

3 4

resulting matrix:

1 3

2 4

Do you want to continue

n

***Q18.Package and Sub package.***

***Solution:***

**package** pck1.pck2;

**public** **class** box {

**private** **double** length,breadth,depth,vol;

**public** box(**double** x,**double** y,**double** z)

{

length=x;

breadth=y;

depth=z;

}

**public** **void** input(**double** x,**double** y,**double** z)

{

length=x;

breadth=y;

depth=z;

}

**public** **void** volume()

{

vol=length\*breadth\*depth;

System.***out***.println("volume is"+vol);

}

}

**package** pck3;

**import** pck1.pck2.box;

**import** java.io.\*;

**public** **class** boxmain {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

box o1=**new** box(2,4,3);

o1.volume();

**double** x,y,z;

System.***out***.println("enter the value of x,y and z");

x=Double.*parseDouble*(br.readLine());

y=Double.*parseDouble*(br.readLine());

z=Double.*parseDouble*(br.readLine());

o1.input(x,y,z);

o1.volume();

}

}

**Output:**

volume is24.0

enter the value of x,y and z

6

7

9

volume is378.0

***Q19.Stack interface implementation.***

***Solution:***

**package** pck1;

**public** **interface** i1 {

**void** push(**int** x);

**int** pop();

}

**package** pck2;

**import** pck1.i1;

**public** **class** stackclass **implements** i1 {

**private** **int** a[];

**private** **int** tos;

**public** stackclass(**int** p)

{

a=**new** **int** [p];

tos=-1;

}

**public** **void** push(**int** x)

{

**if**(tos==a.length-1)

{

System.***out***.println("stack overflow");

**return**;

}

**else**

a[++tos]=x;

}

**public** **int** pop() {

**if**(tos<0)

{

System.***out***.println("stack underflow");

**return** 0;

}

**else**

**return** a[tos--];

}

**public** **void** display()

{

System.***out***.println("stack is");

**for**(**int** x=tos;x>=0;x--)

System.***out***.println(a[x]);

}

**import** pck2.stackclass;

**import** java.io.\*;

**public** **class** stackmain {

**public** **static** **void** main(String[] args)**throws** IOException {

// **TODO** Auto-generated method stub

stackclass o1=**new** stackclass(6);

**int** x;

String ch1;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

**do**

{

System.***out***.println("1.Push operation:");

System.***out***.println("2.Pop operation:");

System.***out***.println("3.Display operation:");

System.***out***.println("Enter your choice:");

**int** ch=Integer.*parseInt*(br.readLine());

**switch**(ch)

{

**case** 1: System.***out***.println("Enter the value to push");

x=Integer.*parseInt*(br.readLine());

o1.push(x);

**break**;

**case** 2: x=o1.pop();

System.***out***.println("popped element is"+x);

**break**;

**case** 3: o1.display();

}

System.***out***.println("Do you want to continue");

ch1=br.readLine();

}**while**(ch1.equals("y"));

}

}

}

**Output:**

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

1

Enter the value to push

78

Do you want to continue

y

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

1

Enter the value to push

90

Do you want to continue

y

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

2

popped element is90

Do you want to continue

y

1.Push operation:

2.Pop operation:

3.Display operation:

Enter your choice:

3

stack is

78

45

Do you want to continue

n

***Q20. Interface extends another interface.***

***solution***

**public** **interface** i1 {

**void** callme(**int** p);

}

**interface** i2 **extends** i1

{

**void** display();

}

**class** interface1 **implements** i2{

**public** **void** callme(**int** p)

{

System.***out***.println(p);

}

**public** **void** display()

{

System.***out***.println("interface display function");

}

}

**public** **class** interfacemain {

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

// **TODO** Auto-generated method stub

interface1 o1=**new** interface1();

o1.callme(67);

o1.display();

}

}

**Output:**

67

interface display function

***Q21.Box,boxweight and boxcolor classes in different packages.***

***Solution:***

**package** p1;

**import** p.i1;

**public** **class** box **implements** i1 {

**protected** **double** length,breadth,depth,vol;

**public** box()

{

length=6;

breadth=6;

depth=6;

}

**public** box(**double** x,**double** y,**double** z)

{

length=x;

breadth=y;

depth=z;

}

**public** **void** volume()

{

vol=length\*breadth\*depth;

}

**public** **void** display()

{

System.***out***.println("volume is"+vol);

}

}

**package** p2;

**import** p.i1;

**import** p1.\*;

**public** **class** boxweight **extends** box **implements** i1{

**private** **double** weight;

**public** boxweight()

{

weight=89.8;

}

**public** boxweight(**double** l,**double** b,**double** d,**double** w)

{

**super**(l,b,d);

weight=w;

}

**public** **void** display()

{

System.***out***.println("weight is"+weight);

}

}

**package** p3;

**import** p.i1;

**import** p1.box;

**public** **class** boxcolor **extends** box **implements** i1 {

**protected** String color;

**public** boxcolor()

{

color="blue";

}

**public** boxcolor(**double** x,**double** y,**double** z,String k)

{

**super**(x,y,z);

color=k;

}

**public** **void** display()

{

System.***out***.println("color is"+color);

}

}

**package** p;

**public** **interface** i1 {

**void** display();

}

**package** p;

**import** p1.box;

**import** p2.boxweight;

**import** p3.boxcolor;

**import** java.io.\*;

**public** **class** boxmain {

**public** **static** **void** main(String[] args)**throws** IOException {

// **TODO** Auto-generated method stub

box o1=**new** box(1,2,3);

boxweight o2=**new** boxweight(4,5,6,7);

boxcolor o3=**new** boxcolor(8,10,11,"pink");

String ch;

**int** x;

**do**

{

System.***out***.println("1.box");

System.***out***.println("2.boxweight");

System.***out***.println("3.boxcolor");

System.***out***.println("enter your choice");

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

x=Integer.*parseInt*(br.readLine());

**switch**(x)

{

**case** 1: System.***out***.println("boxclass");

o1.volume();

o1.display();

**break**;

**case** 2: System.***out***.println("boxweight");

o2.display();

**break**;

**case** 3: System.***out***.println("boxcolor");

o3.display();

**break**;

}

System.***out***.println("Do you want to continue");

ch=br.readLine();

}**while**(ch.equals("y"));

}

}

**Output:**

1.box

2.boxweight

3.boxcolor

enter your choice

1

boxclass

volume is6.0

Do you want to continue

y

1.box

2.boxweight

3.boxcolor

enter your choice

2

boxweight

weight is7.0

Do you want to continue

y

1.box

2.boxweight

3.boxcolor

enter your choice

3

boxcolor

color is pink

Do you want to continue

n

***Q22. Create an exception subclass negative number ,which print the sum of numbers and throw negative numbers.***

***Solution:***

**import** java.io.\*;

**class** Negexcep **extends** Exception

{

String str;

Negexcep(String str)

{

**this**.str=str;

}

**public** String toString()

{

**return** str;

}

}

**public** **class** Negativenumberclass {

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

// **TODO** Auto-generated method stub

**int** a[];

**int** sum=0;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("enter the size of array");

**try** {

**int** n=Integer.*parseInt*(br.readLine());

a=**new** **int**[n];

System.***out***.println("Enter the elements");

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

a[i]=Integer.*parseInt*(br.readLine());

**try** {

**if**(a[i]<0)

**throw** **new** Negexcep("Negative number found");

sum=sum+a[i];

}

**catch**(Negexcep o1)

{

System.***out***.println(o1);

}

}

System.***out***.println("The sum is"+sum);

}

**catch**(IOException o)

{

System.***out***.println(o);

}

}

}

**Output:**

enter the size of array

6

Enter the elements

2

-9

Negative number found

6

8

-5

Negative number found

12

The sum is28

***Q23. Write a program to print the content of a file.***

***Solution*:**

**import** java .io.\*;

**public** **class** filecopy {

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

// **TODO** Auto-generated method stub

FileInputStream f1=**null**;

**int** i;

**for**(**int** i1=0;i1<args.length;i1++)

System.***out***.println("args["+i1+"]"+args[i1]);

**if**(args.length!=1)

{

System.***out***.println("File name out mentioned");

**return**;

}

**try**

{

f1=**new** FileInputStream(args[0]);

//f2=new FileOutputStream(args[1]);

**do**

{

i=f1.read();

**if**(i!=-1)

System.***out***.print((**char**)i);

}**while**(i!=-1);

}

**catch**(FileNotFoundException o1)

{

System.***out***.println(o1);

}

**catch**(IOException o1)

{

System.***out***.println(o1);

}

}

}

**Output:**

args[0]C:\Users\user\eclipse-workspace\filehandling\src\n.txt

//A smile is the

//light in the

heart

//window of your heart

when the soul smile!!.

***Q24.write a program to compute series of 13+23+……….***

***Solution:***

**import** java.io.\*;

**public** **class** series {

**public** **static** **void** main(String[] args) **throws** IOException {

// **TODO** Auto-generated method stub

**int** n;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("enter the number of terms:");

n=Integer.*parseInt*(br.readLine());

**if**(n>0)

{

System.***out***.println("series:");

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

{

System.***out***.println((**int**)(Math.*pow*(i, 3)+1));

}

}

**else**

System.***out***.println("you entered unvalid number");

}

}

**Output:**

enter the number of terms:

6

series:

1

2

9

28

65

126

***Q25.Write a program to compute a Christmas tree pattern.***

***Solution:* import** java.io.\*;

**public** **class** merry {

//public static final int h=6;

**public** **static** **void** main(String[] args) **throws** IOException {

// **TODO** Auto-generated method stub

**int** w;

**int** h;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

w=Integer.*parseInt*(br.readLine());

h=Integer.*parseInt*(br.readLine());

**int** space=w\*3;

**int** x=1;

System.***out***.println("merry christmas");

**for**(**int** i=1;i<=h;i++)

{

**for**(**int** j=x;j<=w;j++)

{

**for**(**int** k=space;k>=j;k--)

{

System.***out***.print(" ");

}

**for**(**int** l=1;l<=j;l++)

{

System.***out***.print("\* ");

}

System.***out***.println();

}

x=x+2;

w=w+2;

}

**for**(**int** i=1;i<=h/2;i++)

{

**for**(**int** k=space-3;k>=1;k--)

{

System.***out***.print(" ");

}

**for**(**int** l=1;l<=w/2-4;l++)

{

System.***out***.print("\* ");

}

System.***out***.println();

}

}

}

***Output:***

5

5

merry Christmas

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

\* \* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \*

\* \* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \* \* \* \* \*

\* \* \*

\* \* \*

***Q26.Write a program to compute a mirror image.***

***Solution:***

**public** **class** mirror {

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

// **TODO** Auto-generated method stub

**int** rows=7;

**for**(**int** i=1;i<=rows;i++)

{

**for**(**int** j=1;j<i;j++)

{

System.***out***.print(" ");

}

**for**(**int** k=i;k<=rows;k++)

{

System.***out***.print(k+" ");

}

System.***out***.println();

}

**for**(**int** i=rows-1;i>=1;i--)

{

**for**(**int** j=1;j<i;j++)

{

System.***out***.print(" ");

}

**for**(**int** k=i;k<=rows;k++)

{

System.***out***.print(k+" ");

}

System.***out***.println();

}

System.***out***.println(" the mirror pattern");

}

}

***Output:***

1 2 3 4 5 6 7

2 3 4 5 6 7

3 4 5 6 7

4 5 6 7

5 6 7

6 7

7

6 7

5 6 7

4 5 6 7

3 4 5 6 7

2 3 4 5 6 7

1 2 3 4 5 6 7

The mirror pattern

***Q27.Write a to make a smiley shape in swimgs.***

***Solution:***

**import** java.awt.\*;

**import** javax.swing.\*;

**public** **class** Smiley **extends** JPanel{

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

// **TODO** Auto-generated method stub

Smiley s=**new** Smiley();

JFrame ap=**new** JFrame("Smiley");

ap.add(s);

ap.setSize(300,300);

ap.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

ap.setVisible(**true**);

}

**public** **void** paint(Graphics g)

{

g.setColor(Color.***yellow***);

//Draw and fill the face

g.drawArc(100,100,250,250,0,360);

g.fillArc(100,100,250,250,0,360);

//Change color to black

g.setColor(Color.***black***);

//Draw the left eye

g.drawArc(170,185,25,25,0,360);

g.fillArc(170,185,25,25,0,360);

//Draw the right eye

g.drawArc(255,185,25,25,0,360);

g.fillArc(255,185,25,25,0,360);

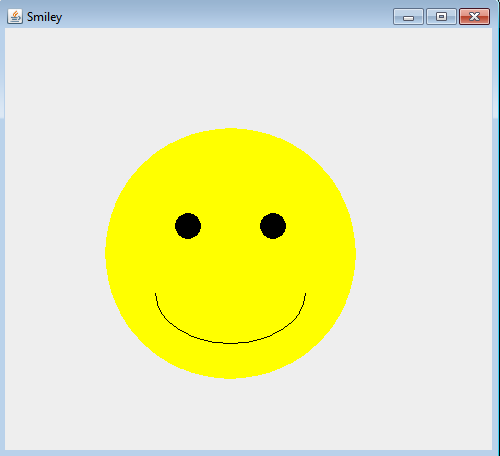
//Draw the smile

g.drawArc(150,215,150,100,0,-180);

}

}

**Ouput:**

******

***Q28.Write a program to change the face of smiley to sad to happy.***

***Solution:***

**import** javax.swing.\*;

**import** java.awt.\*;

**import** java.awt.event.\*;

**public** **class** s **extends** JFrame **implements** ActionListener {

// initializes JButton and JPanel

JButton change = **new** JButton("Turn that frown upside down!");

JPanel panel = **new** JPanel();

**int** i = 0;

// constructor

**public** s() {

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

add(panel); // adds JPanel

panel.setLayout(**new** FlowLayout());

// sets color to yellow

panel.setBackground(Color.***YELLOW***);

panel.add(change); // adds button

change.addActionListener(**this**); // allows button to receive user input

}

**public** **void** paint(Graphics g) {

**super**.paint(g);

//Graphics2D draw = (Graphics2D) draw1;

g.drawOval(50, 170, 70, 70); // face

g.fillOval(73, 185, 3, 5); // left eye

g.fillOval(93, 185, 3, 5); // right eye

// if button count is on even number, shows frown

**if** (i % 2 == 0) {

g.drawArc(48, 200, 75, 75, 30, 120); // frown

}

// if button count is on odd number, shows smile

**else** {

g.drawArc(48, 150, 75, 75, 210, 120); // smile

}

}

// when button is pressed

**public** **void** actionPerformed(ActionEvent arg0) {

i++; // increments i

repaint(); // calls paint method again

}

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

// creates window

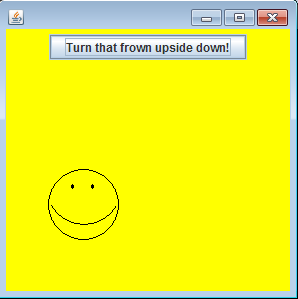
s window = **new** s();

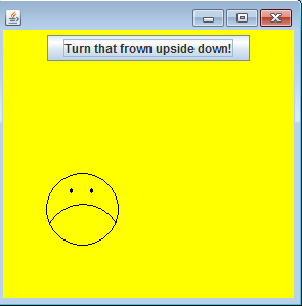
window.setVisible(**true**);

window.setSize(300, 300);

}

}

**Output:**

******

***Q29.Write a program to form a registration form.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** frameref **extends** Frame **implements** ActionListener

{

Label l1,l2,l3;

TextField t1,t2,t3;

Button b,b1;

String msg1=" ";

String msg2=" ";

String msg3=" ";

frameref()

{

**super**("registration form");

l1=**new** Label("name");

l2=**new** Label("age");

l3=**new** Label("address");

t1=**new** TextField(20);

t2=**new** TextField(20);

t3=**new** TextField(20);

b=**new** Button("save");

b1=**new** Button("view");

setLayout(**new** FlowLayout());

add(l1);

add(t1);

add(l2);

add(t2);

add(l3);

add(t3);

add(b);

add(b1);

b.addActionListener(**this**);

b1.addActionListener(**this**);

addWindowListener(**new** WindowAdapter(){

**public** **void** windowClosing(WindowEvent we)

{

System.*exit*(0);

}

});

}

**public** **void** actionPerformed(ActionEvent ae)

{

String s=ae.getActionCommand();

**if**(s.equals("view")) {

msg1="Name-"+t1.getText();

msg2="age-"+t2.getText();

msg3="address-"+t3.getText();

repaint();

}

**else** **if**(s.equals("save"))

{

msg1=" submitted";

msg2=" ";

msg3=" ";

repaint();

}

}

**public** **void** paint(Graphics g)

{

g.drawString(msg1, 100, 150);

g.drawString(msg2, 100, 200);

g.drawString(msg3, 100, 250);

}

}

**public** **class** registration {

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

// **TODO** Auto-generated method stub

frameref o1=**new** frameref();

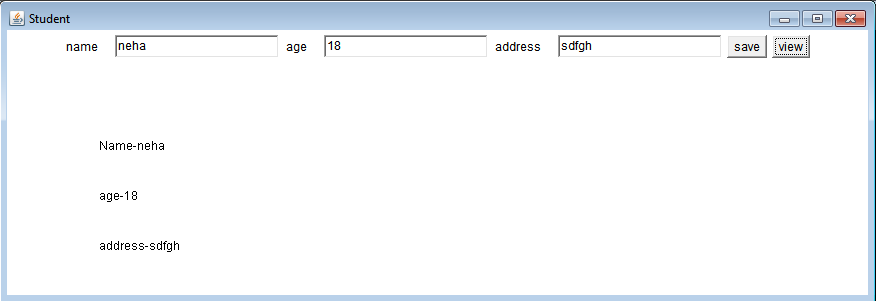
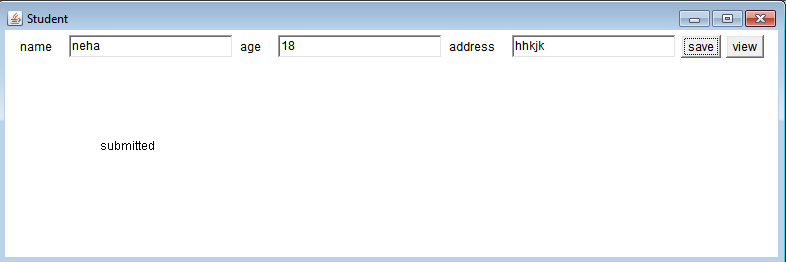
o1.setTitle("Student");

o1.setSize(**new** Dimension(300,300));

o1.setVisible(**true**);

}

}

***Output:***

***Q30.Using awt to type only capital letter in first line and in second line type only lower letters using keyadapter.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** Myframe2 **extends** Frame

{

String msg="Typed characters are";

String msg1=" ";

String msg2=" ";

Myframe2(String str)

{

**super**(str);

setBackground(Color.***pink***);

setForeground(Color.***blue***);

addKeyListener(**new** MyKeyAdapterk(**this**));

addWindowListener(**new** MyWindowAdapterw(**this**));

}

**public** **void** paint(Graphics g)

{

g.drawString(msg,100,100);

g.drawString(msg1,100,120);

g.drawString(msg2,100,140);

}

}

**class** MyKeyAdapterk **extends** KeyAdapter

{

Myframe2 o1;

**int** counter=0;

MyKeyAdapterk(Myframe2 o2)

{

o1=o2;

}

**public** **void** keyTyped(KeyEvent ke)

{

counter++;

**char** c1,c;

**if**(counter<=20) {

c1=Character.*toUpperCase*(ke.getKeyChar());

o1.msg+= c1;

}

**else** **if**(counter<=40) {

c1=Character.*toLowerCase*(ke.getKeyChar());

o1.msg1+=c1;

}

**else** {

c1=Character.*toUpperCase*(ke.getKeyChar());

o1.msg2+=c1;

}

o1.repaint();

}

}

**class** MyWindowAdapterw **extends** WindowAdapter

{

Myframe2 o1;

MyWindowAdapterw(Myframe2 o2)

{

o1=o2;

}

**public** **void** windowClosing(WindowEvent we)

{

o1.setVisible(**false**);

}

}

**public** **class** upperchase {

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

// **TODO** Auto-generated method stub

Myframe2 o1=**new** Myframe2("KeyTyped");

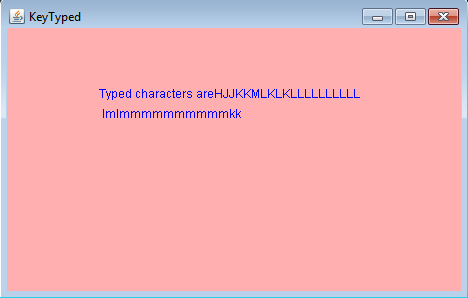
o1.setSize(**new** Dimension(400,400));

o1.setVisible(**true**);

}

}

**Output:**

**

***Q31.write a program to compute factorial when mouse clicked in swings.***

***Solution***

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** java.io.\*;

**class** mouse1 **extends** Frame **implements** MouseListener {

String msg = " ";

**int** mouseX , mouseY ; // coordinates of mouse

**int** i=0;

mouse1() {

addWindowListener(**new** MyWindowAdapterm());

addMouseListener(**new** MouseAdapter()

{

**public** **void** mouseClicked(MouseEvent me)

{

**int** y;

i++;

mouseX=me.getX();

mouseY=me.getY();

y=fact(i);

msg+=y;

System.***out***.println(y);

repaint();

}

**int** fact(**int** k)

{

**int** pro=1;

**for**(**int** j=1;j<=k;j++)

pro=pro\*j;

**return** pro;

}

});

}

**public** **void** paint(Graphics g) {

**super**.paint(g);

g.drawString(msg, mouseX, mouseY);

msg=" ";

}

**class** MyWindowAdapterm **extends** WindowAdapter{

**public** **void** windowClosing(WindowEvent we) {

System.*exit*(0);

}

}

}

**public** **class** factorialanywhere {

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

// **TODO** Auto-generated method stub

mouse1 o1 = **new** mouse1();

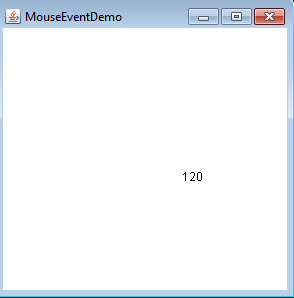
o1.setSize(**new** Dimension (300,300));

o1.setTitle("MouseEventDemo");

o1.setVisible(**true**);

}

}

******

***Output:***

***Q32.Write a program to form a login form using swings.***

***Solution:***

**package** swingslistquestions;

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**class** framep **extends** JFrame **implements** ActionListener{

JTextField t1;

//JTextField t2;

JPasswordField t;

JLabel l1,l2,l3,l4;

JButton b1;

//String msg=" ";

framep()

{

**super**("password");

setSize(400,400);

l1=**new** JLabel("user name");

l2=**new** JLabel(" password");

l3=**new** JLabel(" ");

l4=**new** JLabel(" ");

t1=**new** JTextField(30);

//t2=new JTextField(30);

t=**new** JPasswordField(30);

b1=**new** JButton("submit");

setLayout(**new** FlowLayout());

//t2.setText("\*");

//t2.setEchoChar('\*');

add(l1);

add(t1);

add(l2);

add(t);

add(b1);

add(l3);

add(l4);

b1.addActionListener(**this**);

addWindowListener(**new** WindowAdapter()

{

**public** **void** windowClosing(WindowEvent we)

{

setVisible(**false**);

}

});

setVisible(**true**);

}

**public** **void** actionPerformed(ActionEvent ae)

{

String s1=t1.getText();

String s2=t.~~getText~~();

//System.out.println("s1="+s1+"s2="+s2);

**if**(s1.equals("neha") && s2.equals("kumari"))

{

l4.setText("user="+t1.getText() +"password="+t.~~getText~~());

l3.setText("login successsfully");

//l4.setText(s);

}

**else** {

l3.setText("invalid username/password");

l4.setText("enter again");

}

}

}

**public** **class** password {

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

// **TODO** Auto-generated method stub

SwingUtilities.*invokeLater*(**new** Runnable()

{

**public** **void** run()

{

**new** framep();

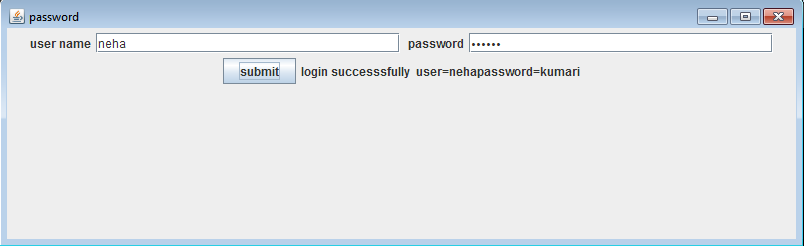
}

});

}

}

***Output:***

******

***Q33.write a program in awt to copy the content of file in label and text field.***

***Solution:***

**package** swingslistquestions;

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**import** java.io.\*;

**class** frame1 **extends** JFrame **implements** ActionListener

{

JButton b1,b2;

JTextField t1;

FileInputStream f1;

JLabel l;

frame1()

{

**super**("filesave");

setSize(600,600);

b1=**new** JButton("view");

//b1.setActionCommand("submit");

t1=**new** JTextField(20);

l=**new** JLabel();

setLayout(**new** FlowLayout());

add(b1);

add(t1);

add(l);

b1.addActionListener(**this**);

getContentPane().setBackground(Color.***red***);

setVisible(**true**);

}

**public** **void** actionPerformed(ActionEvent ae)

{

String s1=t1.getText();

**int** x;

String m=" ";

**try** {

f1= **new** FileInputStream("C:\\Users\\user\\eclipse-workspace\\swingslistquestions\\src\\swingslistquestions\\a1.txt");

**do**

{

x=f1.read();

**if**(x!=-1)

{

m+=(**char**)x;

}

}**while**(x!=-1);

t1.setText(m);

l.setText(m);

}

**catch**(FileNotFoundException e)

{

System.***out***.println(e);

}

**catch**(IOException e)

{

System.***out***.println(e);

}

}

**public** **class** filelabelcopy {

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

// **TODO** Auto-generated method stub

SwingUtilities.*invokeLater*(**new** Runnable()

{

**public** **void** run()

{

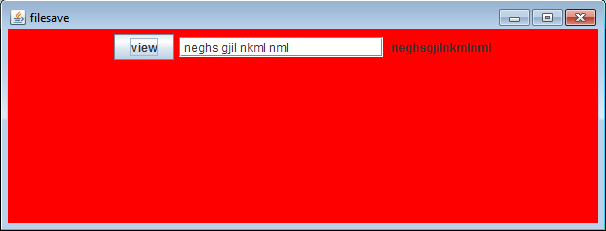
**new** frame1();

}

});

}

}

***Output:***

***Q34.write a program using swings to use inset.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**import** java.util.\*;

**class** panel **extends** JPanel

{

Insets i1;

Random r;

panel()

{

setBorder(BorderFactory.*createLineBorder*(Color.***BLUE***,8));

r=**new** Random();

}

**protected** **void** paintComponent(Graphics g)

{

**super**.paintComponent(g);

**int** x1,y1,x2,y2;

**int** height=getHeight();

**int** width=getWidth();

i1=getInsets();

**for**(**int** i=0;i<10;i++)

{

x1=r.nextInt(width-i1.left);

y1=r.nextInt(height-i1.bottom);

x2=r.nextInt(width-i1.left);

y2=r.nextInt(height-i1.left);

g.drawLine(x1, y1, x2, y2);

}

}

}

**class** paint{

panel p;

paint()

{

JFrame f=**new** JFrame("paint demo");

f.setSize(400,400);

f.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

p=**new** panel();

f.add(p);

f.setVisible(**true**);

}

}

**public** **class** insect {

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

// **TODO** Auto-generated method stub

SwingUtilities.*invokeLater*(**new** Runnable()

{

**public** **void** run()

{

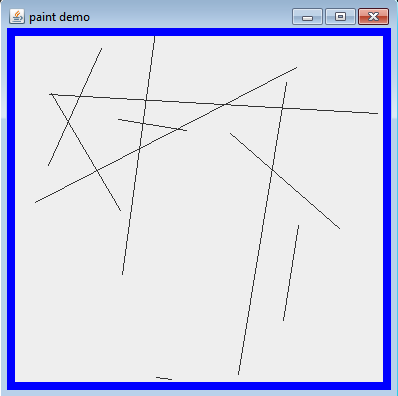
**new** paint();

}

});

}

}

***Output:***

***Q35.write a program using awt to get display area (inset).***

***Solution:***

**package** insect;

**import** java.awt.\*;

**import** java.awt.event.\*;

**public** **class** resizeme **extends** Frame {

**int** max=500;

**int** min=300;

Dimension d;

**final** **int** inc =25;

resizeme()

{

addMouseListener(**new** MouseAdapter(){

**public** **void** mouseReleased(MouseEvent me)

{

**int** w=(d.width+inc)>max?min:(d.width+inc);

**int** h=(d.height+inc)>max?min:(d.height+inc);

setSize(**new** Dimension(w,h));

}

});

addWindowListener(**new** WindowAdapter() {

**public** **void** windowClosing(WindowEvent we)

{

System.*exit*(0);

}

});

}

**public** **void** paint(Graphics g)

{

Insets i=getInsets();

d=getSize();

**int** wr=100,hr=100,offset=80;

**int** x[]= {150,150,350,350,250};

**int** y[]= {250,350,350,250,150};

g.drawLine(i.left, i.top, d.width-i.right, d.height-i.bottom);

g.drawLine(i.left, d.height-i.bottom, d.width-i.right, i.top);

System.***out***.println(d.width-i.right);

**if**(wr<=d.width-i.right && hr<=d.height-i.bottom)

g.drawRect(i.top+offset,i.left+offset,wr,hr);

g.drawPolygon(x, y, 5);

}

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

// **TODO** Auto-generated method stub

resizeme r=**new** resizeme();

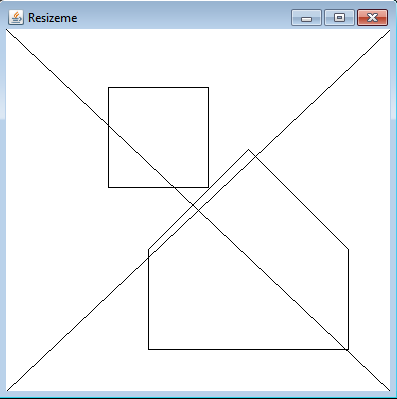
r.setTitle("Resizeme");

r.setSize(400,400);

r.setVisible(**true**);

}

}



**Output:**

***Q36.Write a program using Awt to make a calculator.***

***Solution:***

**package** calculator;

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** Myframe **extends** Frame **implements** ActionListener

{

TextField t1,t2;

String msg=" ";

Label l1,l2;

Button b1,b2,b3,b4,b5;

Myframe()

{

t1=**new** TextField(40);

t2=**new** TextField(40);

l1=**new** Label("enter the number:num1");

l2=**new** Label("enter the number:num2");

b1=**new** Button("+");

b2=**new** Button("-");

b3=**new** Button("/");

b4=**new** Button("\*");

b5=**new** Button("clear");

setLayout(**new** FlowLayout());

add(l1);

add(t1);

add(l2);

add(t2);

add(b1);

add(b2);

add(b3);

add(b4);

add(b5);

b1.addActionListener(**this**);

b2.addActionListener(**this**);

b3.addActionListener(**this**);

b4.addActionListener(**this**);

b5.addActionListener(**this**);

addWindowListener(**new** WindowAdapter() {

**public** **void** windowClosing(WindowEvent we)

{

System.*exit*(0);

}

});

}

**public** **void** actionPerformed(ActionEvent ae)

{

**int** x=0;

String s2=ae.getActionCommand();

String s=t1.getText();

String s1=t2.getText();

**if**(s.equals("") || s1.equals(" "))

{

msg="insufficient inputs";

repaint();

}

**else**

{

**if**(s2.equals("+"))

{

x=Integer.*parseInt*(s)+Integer.*parseInt*(s1);

msg="result"+x;

repaint();

}

**else** **if**(s2.equals("-"))

{

x=Integer.*parseInt*(s)-Integer.*parseInt*(s1);

msg="result"+x;

repaint();

}

**else** **if**(s2.equals("\*"))

{

x=Integer.*parseInt*(s)\*Integer.*parseInt*(s1);

msg="result"+x;

repaint();

}

**else** **if**(s2.equals("/"))

{

x=Integer.*parseInt*(s)/Integer.*parseInt*(s1);

msg="result"+x;

repaint();

}

**else** **if**(s2.equals("clear"))

{

t1.setText("0");

t2.setText("0");

}

}

}

**public** **void** paint(Graphics g)

{

g.drawString(msg,100, 130);

}

}

**public** **class** Calculator {

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

// **TODO** Auto-generated method stub

Myframe o1=**new** Myframe();

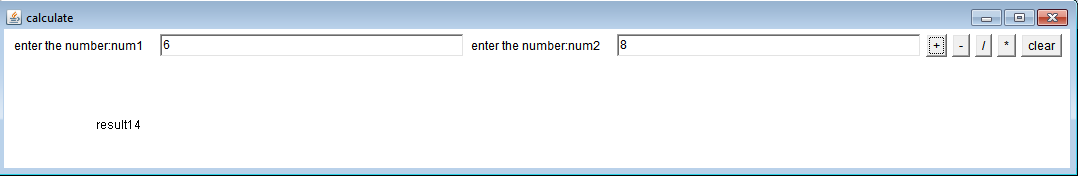
o1.setSize(600,600);

o1.setTitle("calculate");

o1.setVisible(**true**);

}

}

**Output:**

***Q37.write a program Simple Mouse Handling Event.***

**Solution:**

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** MyFrame **extends** Frame **implements** MouseListener,WindowListener{

String msg ="Hello";

MyFrame(String s)

{

**super**(s);

addMouseListener(**this**);

addWindowListener(**this**);

}

**public** **void** windowClosed(WindowEvent ae)

{

}

**public** **void** windowAcitvated(WindowEvent ae)

{

}

**public** **void** windowClosing(WindowEvent ae)

{

setVisible(**false**);

}

**public** **void** windowDeactivated(WindowEvent ae)

{

}

**public** **void** windowDeiconified(WindowEvent ae)

{

}

**public** **void** windowIconfied(WindowEvent ae)

{

}

**public** **void** windowOpened(WindowEvent ae)

{

}

**public** **void** mouseClicked(MouseEvent ae1)

{

msg="Mouse Clicked”;

repaint();

}

**public** **void** mousePressed(MouseEvent ae1)

{

msg="Mouse Pressed";

repaint();

}

**public** **void** mouseReleased(MouseEvent ae1)

{

msg="Mouse Released";

repaint();

}

**public** **void** mouseEntered(MouseEvent ae1)

{

msg="Mouse Entered";

repaint();

}

**public** **void** mouseExited(MouseEvent ae1)

{

msg="Mouse Exited";

repaint();

}

**public** **void** paint(Graphics g)

{

g.drawString(msg,50,100);

}

}

**public** **class** mouseevenhandling {

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

// **TODO** Auto-generated method stub

MyFrame o1=**new** MyFrame("MouseEventhandling");

o1.setSize(300,300);

o1.setVisible(**true**);

}

}

**Output:**



***Q38.write a program using awt to draw shapes using buttons.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** myframe **extends** Frame **implements** ActionListener

{

Button b1,b2,b3,b4;

//Label l1;

//TextField t1;

**int** flag=0;

String msg=" ";

String msg1=" ";

myframe()

{

b1=**new** Button("Rectangle");

b2=**new** Button("line");

b3=**new** Button("red");

b4=**new** Button("blue");

setLayout(**new** FlowLayout());

add(b1);

add(b2);

add(b3);

add(b4);

addWindowListener(**new** WindowAdapter() {

**public** **void** windowClosing(WindowEvent we)

{

System.*exit*(0);

}

});

b1.addActionListener((ae)->{

msg=ae.getActionCommand();//lambda EXPRESSION

flag=1;

repaint();

});

b2.addActionListener((ae)->{

msg="u pressed"+ae.getActionCommand();//lambda EXPRESSION

flag=2;

repaint();

});

b3.addActionListener((ae)->{

msg="u pressed"+ae.getActionCommand();//lambda EXPRESSION

setForeground(Color.***red***);

repaint();

});

b4.addActionListener((ae)->{

msg="u pressed"+ae.getActionCommand();

setForeground(Color.***blue***);

repaint();

});

}

**public** **void** paint(Graphics g)

{

**if**(flag==1)

{

g.drawRect(50, 100, 100, 100);

**int** area=0,l,b;

l=100;

b=100;

area=l\*b;

msg1="area"+area;

g.drawString(msg1,150,100);

}

**else** **if**(flag==2){

g.drawLine(50, 100, 100, 100);

}

}

}

**public** **class** buttongraphy {

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

// **TODO** Auto-generated method stub

myframe o1=**new** myframe();

o1.setSize(600,600);

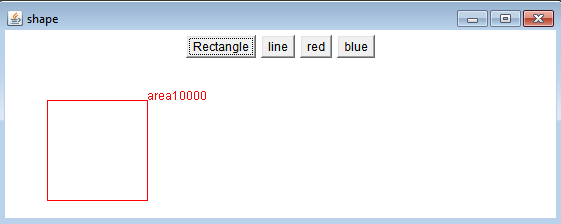
o1.setTitle("shape");

o1.setVisible(**true**);

}

}

**Output:**

******

***Q39.Write a program using awt to draw the shape by reading the text from the text field.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** myframe **extends** Frame **implements** ActionListener{

Button b1,b2,b3;

Label l1;

TextField t1;

**int** flag=0;

myframe()

{

b1=**new** Button("draw figure");

b2=**new** Button("red");

b3=**new** Button("blue");

t1=**new** TextField(30);

l1=**new** Label("enter the label name");

setLayout(**new** FlowLayout());

add(l1);

add(t1);

add(b1);

add(b2);

add(b3);

b1.addActionListener(**this**);

b2.addActionListener(**this**);

b3.addActionListener(**this**);

addWindowListener(**new** WindowAdapter() {

**public** **void** windowClosing(WindowEvent we)

{

System.*exit*(0);

}

});

}

**public** **void** actionPerformed(ActionEvent ae)

{

String s1=t1.getText();

String s=ae.getActionCommand();

**if**(s.equals("draw figure"))

{

**if**(s1.equals("line"))

flag=1;

**else** **if**(s1.equals("rectangle"))

flag=2;

**else** **if**(s1.equals("polygon"))

flag=3;

}

**if**(s.equals("red"))

setForeground(Color.***red***);

**else** **if**(s.equals("blue"))

setBackground(Color.***blue***);

repaint();

}

**public** **void** paint(Graphics g)

{

**int** x[]= {50,50,250,250,150};

**int** y[]= {150,350,350,150,50};

**int** num=5;

**if**(flag==1)

g.drawLine(50,100,100,100);

**else** **if**(flag==2)

g.drawRect(50,100,100,100);

**else** **if**(flag==3)

g.drawPolygon(x,y,5);

}

}

**public** **class** buttontextfield {

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

// **TODO** Auto-generated method stub

myframe o1=**new** myframe();

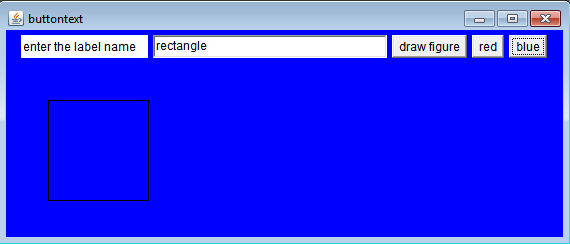
o1.setTitle("buttontext");

o1.setVisible(**true**);

o1.setSize(600,600);

}

}

***Output:***

***Q40.Write a program to create a login form using awt.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** java.io.\*;

**class** framep **extends** Frame **implements** ActionListener{

TextField t1,t2;

Label l1,l2,l3,l4;

Button b1;

//String msg=" ";

framep()

{

**super**("password");

setSize(400,400);

l1=**new** Label("user name");

l2=**new** Label("login password");

l3=**new** Label(" ");

l4=**new** Label(" ");

t1=**new** TextField(30);

t2=**new** TextField(30);

b1=**new** Button("submit");

setLayout(**new** FlowLayout());

t2.setEchoChar('\*');

add(l1);

add(t1);

add(l2);

add(t2);

add(b1);

add(l3);

add(l4);

b1.addActionListener(**this**);

addWindowListener(**new** WindowAdapter()

{

**public** **void** windowClosing(WindowEvent we)

{

setVisible(**false**);

}

});

setVisible(**true**);

}

**public** **void** actionPerformed(ActionEvent ae)

{

String s1=t1.getText();

String s2=t2.getText();

//System.out.println("s1="+s1+"s2="+s2);

**if**(s1.equals("neha") && s2.equals("kumari"))

{

String s="username" +t1.getText();

s+="password"+**new** String(t2.getText());

l3.setText("login successsfully");

l4.setText(s);

}

**else** {

l3.setText("invalid username/password");

l4.setText("enter again");

}

}

}

**public** **class** password {

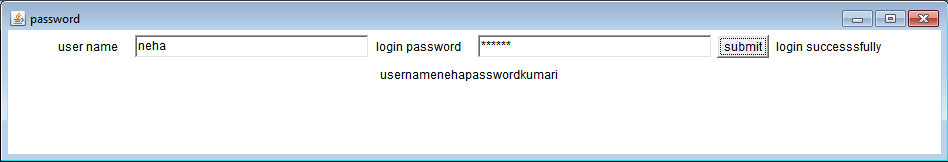
**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

framep o1=**new** framep();

}

}

***Output:***

***Q41.Write a program to form a registration form using swings.***

***Solution:***

**package** swingslistquestions;

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**class** frameref **extends** JFrame **implements** ActionListener

{

JLabel l1,l2,l3,l4;

JTextField t1,t2,t3;

JButton b,b1;

frameref()

{

**super**("registration form");

setSize(**new** Dimension(300,300));

setVisible(**true**);

l1=**new** JLabel("name");

l1.setForeground(Color.***pink***);

l2=**new** JLabel("age");

l3=**new** JLabel("address");

l4=**new** JLabel();

t1=**new** JTextField(20);

t1.setForeground(Color.***blue***);

t1.setBackground(Color.***white***);

t2=**new** JTextField(20);

t3=**new** JTextField(20);

b=**new** JButton("save");

b.setBackground(Color.***cyan***);

b1=**new** JButton("view");

b1.setBackground(Color.***orange***);

//p=new panelab();

//setBackground(Color.blue);

getContentPane().setBackground(Color.***white***);

setLayout(**new** FlowLayout());

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

add(l1);

add(t1);

add(l2);

add(t2);

add(l3);

add(t3);

add(b);

add(b1);

add(l4);

//add(p);

b.addActionListener(**this**);

b1.addActionListener(**this**);

}

**public** **void** actionPerformed(ActionEvent ae)

{

String msg1=" ";

l4.setText(" ");

String s=ae.getActionCommand();

**if**(s.equals("view")) {

msg1="Name-"+t1.getText();

msg1+="age-"+t2.getText();

msg1+="address-"+t3.getText();

l4.setText(msg1);

}

**else** **if**(s.equals("save"))

{

msg1= "submitted";

l4.setText(msg1);

}

}

}

**public** **class** registration {

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

// **TODO** Auto-generated method stub

SwingUtilities.*invokeLater*(**new** Runnable()

{

**public** **void** run()

{

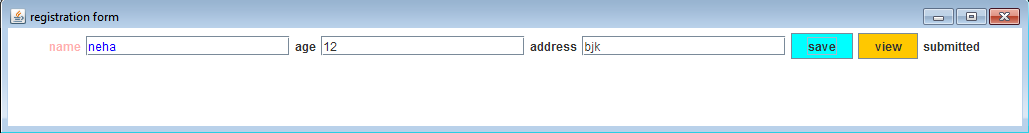
**new** frameref();

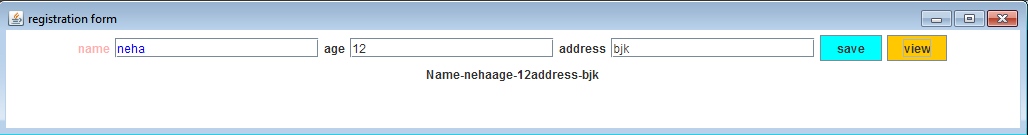
}

});

}

}

***Output:***



***Q42.write a program to copy the content of textfield in file.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**import** java.io.\*;

**class** frame **extends** JFrame **implements** ActionListener

{

JButton b1,b2;

JTextField t1;

FileOutputStream f1;

JLabel l;

frame()

{

**super**("filesave");

setSize(600,600);

b1=**new** JButton("submit");

b1.setActionCommand("submit");

t1=**new** JTextField(20);

l=**new** JLabel("hello");

setLayout(**new** FlowLayout());

add(b1);

add(t1);

add(l);

b1.addActionListener(**this**);

getContentPane().setBackground(Color.***red***);

setVisible(**true**);

}

**public** **void** actionPerformed(ActionEvent ae)

{

String s1=" ";

s1=t1.getText();

System.***out***.println("s="+s1+"actioncommand"+ae.getActionCommand());

**try** {

f1=**new** FileOutputStream("A2.txt",**true**);

**for**(**int** i=0;i<s1.length();i++)

{

System.***out***.print("char="+s1.charAt(i));

f1.write(s1.charAt(i));

}

l.setText("file copied");

t1.setText(" ");

}

**catch**(FileNotFoundException o1)

{

System.***out***.println(o1);

}

**catch**(IOException o1)

{

System.***out***.println(o1);

}

}

**public** **void** actionPerformed(ActionEvent ae)

{

//String s1=t1.getText();

String s1=" ";

//s1=l.getText();

s1=t1.getText();

System.***out***.println("s="+s1+"actioncommand"+ae.getActionCommand());

**try** {

f1=**new** FileOutputStream("A2.txt",**true**);

**for**(**int** i=0;i<s1.length();i++)

{

System.***out***.print("char="+s1.charAt(i));

f1.write(s1.charAt(i));

}

l.setText("file copied");

t1.setText(" ");

}

**catch**(FileNotFoundException o1)

{

System.***out***.println(o1);

}

**catch**(IOException o1)

{

System.***out***.println(o1);

}

}

}

**public** **class** file {

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

// **TODO** Auto-generated method stub

SwingUtilities.*invokeLater*(**new** Runnable()

{

**public** **void** run()

{

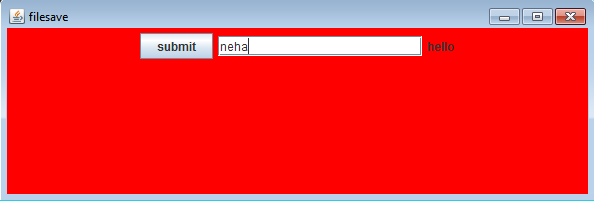
**new** frame();

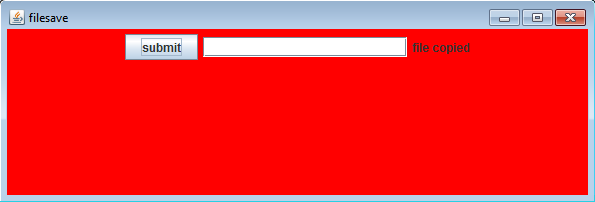
}

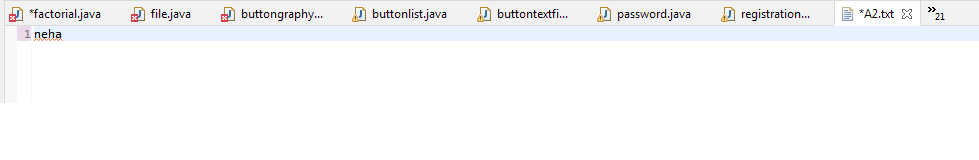
});

}

}

***Output:***





**Q43.*Using swings,write a program to create two buttons labeled ‘A’ and ‘B’.When button ‘A’ is pressed,it displays your personal information(Name,empid,age ) and when button ‘B’ is pressed,it display your salary in previous year.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**class** Empinfo

{

String Name;

**int** empid,age;

Double salary;

Empinfo(String name,**int** Empid,**int** a,**double** salary)

{

Name=name;

empid=Empid;

age=a;

**this**.salary=salary;

}

}

**class** panelE **extends** JPanel

{

String msg1=" ";

String msg2=" ";

String msg3=" ";

String msg4=" ";

**protected** **void** paintComponent(Graphics g)

{

**super**.paintComponent(g);

g.drawString(msg1, 150, 200);

g.drawString(msg2, 150, 220);

g.drawString(msg3, 150, 240);

g.drawString(msg4, 150, 260);

}

}

**class** MyframeE1 **extends** JFrame **implements** ActionListener

{

Empinfo E=**new** Empinfo("Neha",2020331,89,9009);

JButton b1,b2;

panelab p1;

//JLabel l;

MyframeE1()

{

setTitle("button a and b");

setSize(400,400);

p1=**new** panelab();

b1=**new** JButton("A");

b2=**new** JButton("B");

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

p1.add(b1);

p1.add(b2);

add(p1);

b1.addActionListener(**this**);

b2.addActionListener(**this**);

setVisible(**true**);

}

**public** **void** actionPerformed(ActionEvent ae)

{

String s=ae.getActionCommand();

**if**(s.equals("A"))

{

p1.msg1="NAME-"+E.Name;

p1.msg2="EMPID-"+E.empid;

p1.msg3="AGE-"+E.age;

p1.repaint();

}

**else** **if**(s.equals("B")) {

p1.msg1="Salary-"+E.salary;

p1.msg2=" ";

p1.msg3=" ";

p1.msg4="";

p1.repaint();

}

}

}

**public** **class** Empolyee {

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

// **TODO** Auto-generated method stub

SwingUtilities.*invokeLater*(**new** Runnable()

{

**public** **void** run()

{

**new** MyframeE1();

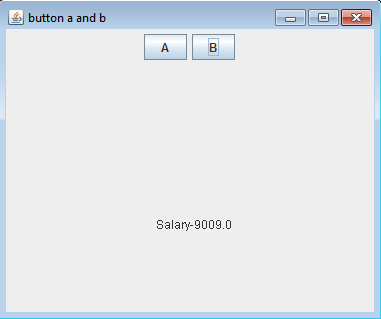
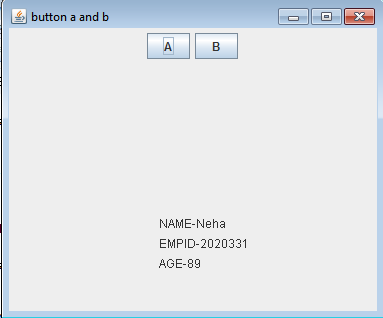
}

});

}

}

**Output:**

****

**Q44.*Using AWT,write a program to create two buttons labeled ‘A’ and ‘B’.When button ‘A’ is pressed,it displays your personal information(Name,emip,age) and when button ‘B’ is pressed,it display your salary in previous year.***

***Solution:***

**import** java.awt.\*;

**import** java.awt.event.\*;

**class** Empinfo

{

String Name;

**int** empid,age;

Double salary;

Empinfo(String name,**int** Empid,**int** a,**double** salary)

{

Name=name;

empid=Empid;

age=a;

**this**.salary=salary;

}

}

**class** MyframeE **extends** Frame **implements** ActionListener

{

Empinfo s1=**new** Empinfo("Neha",2020331,89,9009);

Button b1,b2;

Label l1,l2,l3,l4;

MyframeE()

{

setLayout(**new** FlowLayout());

b1=**new** Button("A");

b2=**new** Button("B");

l1=**new** Label();

l2=**new** Label();

l3=**new** Label();

l4=**new** Label();

add(b1);

add(b2);

add(l1);

add(l2);

add(l3);

add(l4);

b1.addActionListener(**this**);

b2.addActionListener(**this**);

addWindowListener(**new** WindowAdapter(){

**public** **void** windowClosing(WindowEvent we)

{

System.*exit*(0);

}

});

}

**public** **void** actionPerformed(ActionEvent ae)

{

String s=ae.getActionCommand();

l1.setText(" ");

l2.setText(" ");

l3.setText(" ");

l4.setText(" ");

**if**(s.equals("A"))

{

l1.setText("Name-"+s1.Name);

l2.setText("empid-"+s1.empid);

l3.setText("age-"+s1.age);

}

**else** **if**(s.equals("B"))

l1.setText("salary-"+s1.salary);

}

}

**public** **class** empolydetails {

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

// **TODO** Auto-generated method stub

MyframeE o1=**new** MyframeE();

o1.setTitle("EMPLOYEE");

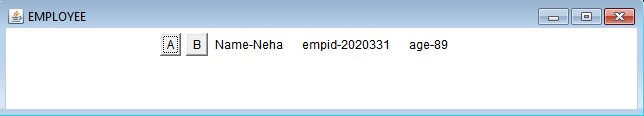
o1.setSize(**new** Dimension(300,300));

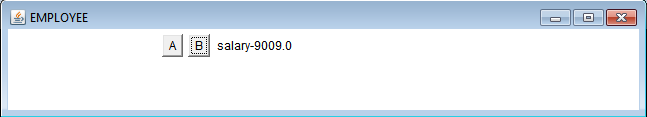
o1.setVisible(**true**);

}

}

***Output:***

****



***Q45. Pattern :***

**Solution:**

**public** **class** p

{

**public** **static** **void** main(String args[])

{

**int** row=7;

**for** (**int** i= row; i>= 1; i--)

{

**for** (**int** j=row; j>i;j--)

{

System.***out***.print(" ");

}

**for** (**int** k=1;k<=i;k++)

{

System.***out***.print("\*");

}

System.***out***.println("");

}

}

}

**Output:**

\*\*\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

***Q46.write a program to print n odd numbers.***

***Solution:***

**import** java.io.\*;

**public** **class** oddnumber {

**public** **static** **void** main(String[] args) **throws** IOException{

// **TODO** Auto-generated method stub

**int** i,n;

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.***in***));

System.***out***.println("enter the value of n");

n=Integer.*parseInt*(br.readLine());

**for**(i=1;i<=n;i++)

{

System.***out***.println("odd number is"+i);

i=i+1;

}

}

}

**Output:**

enter the value of n

8

odd number is1

odd number is3

odd number is5

odd number is7

***Q47.write a program to reverse a string.***

***Solution:* import** java.util.\*;

**public** **class** reverse {

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

// **TODO** Auto-generated method stub

String original, reverse = "";

Scanner in = **new** Scanner(System.***in***);

System.***out***.println("Enter a string to reverse");

original = in.nextLine();

**int** length = original.length();

**for** (**int** i = length - 1 ; i >= 0 ; i--)

reverse = reverse + original.charAt(i);

System.***out***.println("Reverse of the string: " + reverse);

}

}

***Output:***

Enter a string to reverse

neha

Reverse of the string: ahen

***Q48.write a program to swap two numbers.***

***Solution:***

**import** java.util.\*;

**public** **class** swap {

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

// **TODO** Auto-generated method stub

**int** x, y, t;// x and y are to swap

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter the value of X and Y");

x = sc.nextInt();

y = sc.nextInt();

System.***out***.println("before swapping numbers: "+x +" "+ y);

/\*swapping \*/

t = x;

x = y;

y = t;

System.***out***.println("After swapping: "+x +" " + y);

System.***out***.println( );

}

}

**Output:**

Enter the value of X and Y

6

8

before swapping numbers: 6 8

After swapping: 8 6

***Q49.write a program to check whether the number is print or not.***

***Solution:***

**public** **class** prime {

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

// **TODO** Auto-generated method stub

**int** i,m=0,flag=0;

**int** n=89;//it is the number to be checked

m=n/2;

**if**(n==0||n==1){

System.***out***.println(n+" is not prime number");

}**else**{

**for**(i=2;i<=m;i++){

**if**(n%i==0){

System.***out***.println(n+" is not prime number");

flag=1;

**break**;

}

}

**if**(flag==0) { System.***out***.println(n+" is prime number"); }

}//end of else

}

}

**Output:** 89 is prime number

***Q50.write to make a make a robot face.***

***Solution:* public** **class** print {

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

// **TODO** Auto-generated method stub

System.***out***.println(" +\"\"\"\"\"+ ");

System.***out***.println("[| o o |]");

System.***out***.println(" | ^ | ");

System.***out***.println(" | '-' | ");

System.***out***.println(" +-----+ ");

}

}

***Output:***

+"""""+

[| o o |]

| ^ |

| '-' |

+-----+ **Name:neha kumari**

**Roll no:2020331**

**Semester:IInd**