|  |
| --- |
| **Program 1 Date: 14-12-2022** |
| **Write a program to print your name** |

**Code:**

class Name

{

public static void main(String[] args)

{

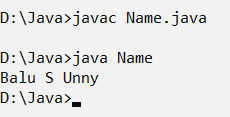
String Name="Balu";

System.out.print(Name+" S Unny");

}

}

**Output:**



|  |
| --- |
| **Program 2 Date:14-12-2022** |
| **Write a program to display two numbers received as command line argument, and print its product.** |

**Code:**

class Prod

{

public static void main(String[] args){

int p=1;

int i;

int a=Integer.parseInt(args[0]);

for(i=1;i<=a;i++)

{

p=p\*i;

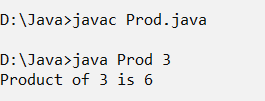
}

System.out.println("Product of "+a+" is " +p);

}

}

**Output:**



|  |
| --- |
| **Program 3 Date:14-12-2022** |
| **Write a program to display two strings received as command line arguments** |

**Code:**

class Str

{

public static void main(String[] args)

{

System.out.println("Strings are: ");

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

{

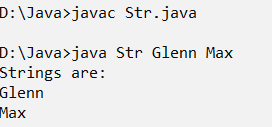
System.out.println(args[i]);

}

}

}

**Output:**

****

|  |
| --- |
| **Program 4 Date: 14-12-2022** |
| **Write a program to read two numbers and display the output in the form of ‘Sum of 2 and 3 is 5** |

**Code:**

class Sum

{

public static void main(String[] args)

{

int x=Integer.parseInt(args[0]);

int y=Integer.parseInt(args[1]);

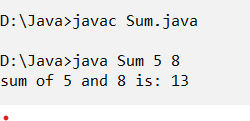
int sum=x+y;

System.out.println("sum of "+x+" and "+y+" is: " +sum);

}

}

**Output**

****

|  |
| --- |
| **Program 5 Date: 14-12-2022** |
| **Write a program to accept two numbers from the keyboard and swap them.** |

**Code:**

import java.util.Scanner;

class Swap

{

public static void main(String[] args)

{

int n,m,t;

Scanner num=new Scanner(System.in);

System.out.println("Enter the first number");

n=num.nextInt();

System.out.println("Enter the second number");

m=num.nextInt();

t=n;

n=m;

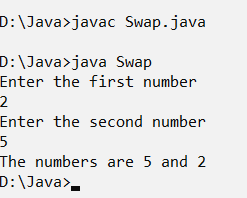
m=t;

System.out.print("The numbers are "+n+" and "+m+" ");

}

}

**Output:**

****

|  |
| --- |
| **Program 6 Date: 16-12-2022** |
| **WAP to read three numbers and the find maximum** |

**Code:**

import java.util.Scanner;

public class Maximum {

static Scanner sc = new Scanner(System.in);

public static void main(String[] args) {

int a,b,c;

System.out.println("Enter 3 Numbers");

a = sc.nextInt();

b = sc.nextInt();

c = sc.nextInt();

int max;

if(a > b && a > c){

max = a;

}else if(b > c){

max = b;

}else{

max = c;

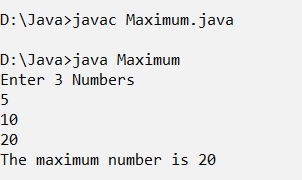
}

System.out.println("The maximum number is "+max);

}

}

**Output:**

****

|  |
| --- |
| **Program 7 Date: 16-12-2022** |
| **Find the minimum of three numbers using a single statement** |

**Code:**

import java.util.Scanner;

public class Min

{

public static void main(String[] args)

{

Scanner no=new Scanner(System.in);

int a=no.nextInt();

int b=no.nextInt();

int c=no.nextInt();

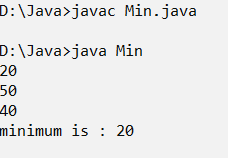
int min=(a<b?a<c?a:c:b<c?b:c);

System.out.println("minimum is : " +min);

}

}

**Output:**

****

|  |
| --- |
| **Program 8 Date: 16-12-2022** |
| **WAP to search for a given element in an array.** |

**Code:**

import java.util.Scanner;

class Search

{

public static void main(String[] args)

{

int n,i,key;

Scanner scn=new Scanner(System.in);

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

n=scn.nextInt();

int a[]=new int[n];

System.out.println("enter the numbers: " );

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

a[i]=scn.nextInt();

}

System.out.print("numbers are: " );

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

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

}

System.out.print("enter the number to find: " );

key=scn.nextInt();

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

if(a[i]==key) {

System.out.println("number found");

break;

}

}

if(i==n) {

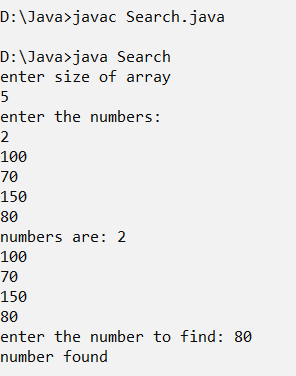
System.out.println("not found " );

}

}

}

**Output:**

****

|  |
| --- |
| **Program 9 Date:16-12-2022** |
| **WAP to sort elements in an array in ascending order.** |

**Code:**

import java.util.Scanner;

class Sort

{

public static void main(String[] args)

{

int n,i,j,t;

System.out.println("enter array size");

Scanner scn=new Scanner(System.in);

n=scn.nextInt();

int a[]=new int[n];

System.out.print("enter elements");

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

a[i]=scn.nextInt();

}

System.out.println("the elements are:");

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

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

}

System.out.println();

System.out.println("sorted elements are:");

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

for(j=i+1;j<n;j++){

if(a[i]>a[j]){

t=a[i];

a[i]=a[j];

a[j]=t;

}

}

}

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

{

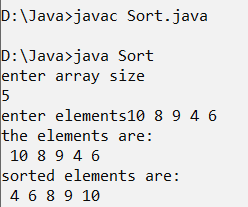
System.out.print(" "+a[i]);

}

}

}

**Output:**

****

|  |
| --- |
| **Program 10 Date:16-12-2022** |
| **Write a program to print the row wise and column wise sum of a 2D array.** |

**Code:**

import java.util.Scanner;

class Sum2d

{

public static void main(String[] args)

{

int r,c;

int i,j;

System.out.println("Enter the row size:");

Scanner scn=new Scanner(System.in);

r=scn.nextInt();

System.out.println("Enter the column size:");

c=scn.nextInt();

int a[][]=new int[r][c];

System.out.print("Element at a[i][j] : " );

for(i=0;i<r;i++) {

for(j=0;j<c;j++) {

a[i][j]=scn.nextInt();

}

}

System.out.println("the array elements are:");

for(i=0;i<r;i++) {

for(j=0;j<c;j++) {

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

}

System.out.println();

}

int sum=0;

for(i=0;i<r;i++) {

int rsum=0;

int csum=0;

for(j=0;j<c;j++) {

rsum += a[i][j];

csum += a[j][i];

}

System.out.println("Row sum " +(i)+ "=" +rsum);

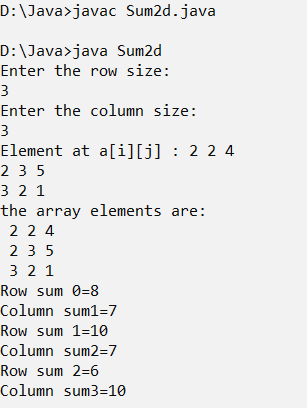
System.out.println("Column sum" +(i+1)+ "=" +csum);

}

}

}

**Output:**

****

|  |
| --- |
| **Program 11 Date: 21-12-2022** |
| **WAP with two functions to check for an integer palindrome.** |

**Code:**

import java.util.Scanner;

public class Palin {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a Number: ");

int num = sc.nextInt();

int rev = 0, n = num;

while (n > 0) {

rev = (rev \* 10) + (n % 10);

n = n / 10;

}

if(num == rev)

{

System.out.println("The number is a palindrome");

}

else{

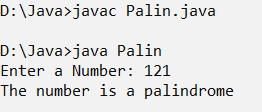
System.out.println("The number is not a palindrome");

}

}

}

**Output:**

****

|  |
| --- |
| **Program 12 Date: 21-12-2022** |
| **WAP to display numbers from m to n using a single while loop. (eg: m=2, n=8 - randomly given numbers)** |

**Code:**

import java.util.Scanner;

public class Randomn{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int m,n;

System.out.println("Enter the first number: ");

m = sc.nextInt();

System.out.println("Enter the last number ");

n = sc.nextInt();

while(m != n){

System.out.print(m+"\t");

if(m > n){

m--;

}else{

m++;

}

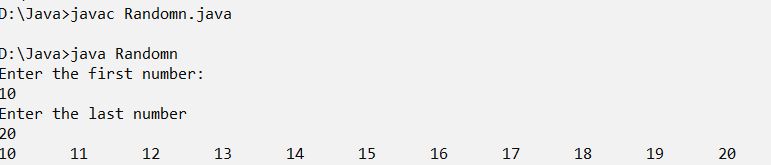
}

System.out.print(n);

}

}

**Output:**

****

|  |
| --- |
| **Program 13 Date: 21-12-2022** |
| **WAP to find the sum of the series 1+(1+2)+(1+2+3)+(1+2+3+…+n) using a single while loop.** |

**Code:**

import java.util.Scanner;

public class Sumseries{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int n,sum = 0,i = 1;

System.out.print("Enter the value of n: ");

n = sc.nextInt();

while(i <= n+1){

sum += i;

System.out.print(sum+"\t");

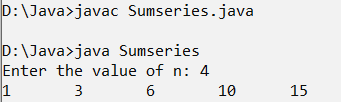
i++;

}

}

}

**Output:**

****

|  |
| --- |
| **Program 14 Date: 21-12-2022** |
| **WAP to find the sum of 1+2/2!+3/3!+4/4!++n/n! using a single for loop.** |

**Code:**

import java.util.\*;

public class Sfact

{

static int fact(float x){

int fact=1;

for(int i=1;i<=x;i++)

{

fact=fact\*i;

}

return fact;

}

public static void main(String[]args)

{

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number: ");

int n= sc.nextInt();

float sum=0;

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

{

sum=sum+i/fact(i);

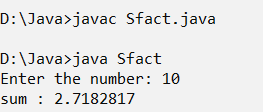
}

System.out.println("sum : " +sum);

}

}

**Output:**

****

|  |
| --- |
| **Program 15 Date: 21-12-2022** |
| **WAP to calculate the area of a circle (method with no argument and no return type.** |

**Code:**

import java.util.Scanner;

class Area

{

double r,pi;

Area(double t) {

r=t;

pi=3.14;

}

void area() {

double area=0;

area=pi\*r\*r;

System.out.println("area is: " +area);

}

public static void main(String[] args)

{

Scanner scn=new Scanner(System.in);

System.out.print("enter value of radius: ");

double r=scn.nextDouble();

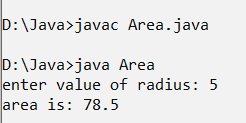
Area a=new Area(r);

a.area();

}

}

**Output:**

****

|  |
| --- |
| **Program 16 Date: 21-12-2022** |
| **WAP to calculate sum of n even numbers (method with no argument and return type.)** |

**Code:**

import java.util.Scanner;

class Sum\_even

{

int n;

Sum\_even(int e)

{

n=e;

}

void sum() {

int s=0;

int i;

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

if(i%2==0) {

s=s+i;

}

}

System.out.println("Sum is: " +s);

}

public static void main(String[] args)

{

int n;

System.out.println("Enter the limit: ");

Scanner scn=new Scanner(System.in);

n=scn.nextInt();

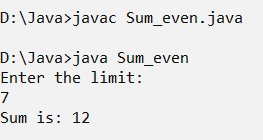
Sum\_even obj=new Sum\_even(n);

obj.sum();

}

}

**Output:**

****

|  |
| --- |
| **Program 17 Date: 21-12-2022** |
| **WAP to reverse a number (method with argument and no return type.)** |

**Code:**

import java.util.Scanner;

class Reverse

{

void rev(int n) {

int t,d,rem;

t=n;

rem=0;

while(t!=0) {

d=t%10;

rem=(rem\*10)+d;

t=t/10;

}

System.out.println("Reverse of a number is : "+rem);

}

public static void main(String[] args)

{

int n;

System.out.println("enter the number");

Scanner scn=new Scanner(System.in);

n=scn.nextInt();

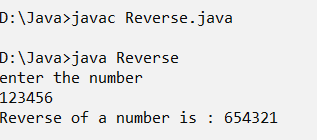
Reverse obj=new Reverse();

obj.rev(n);

}

}

**Output:**

****

|  |
| --- |
| **Program 18 Date: 21-12-2022** |
| **WAP to calculate the sum of digits of a number (method with argument and return type.)** |

**Code:**

import java.util.\*;

class Sums

{

int sum(int n)

{

int t=n;

int rev=0,d;

while(t!=0)

{

d=t%10;

rev=rev+d;

t=t/10;

}

return rev;

}

public static void main(String[] args)

{

int n;

System.out.println("Enter the number");

Scanner scn=new Scanner(System.in);

n=scn.nextInt();

Sums obj=new Sums();

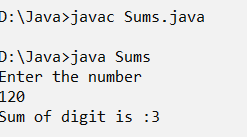
int s=obj.sum(n);

System.out.println("Sum of digit is :"+s);

}

}

**Output:**

****

|  |
| --- |
| **Program 19 Date: 04-01-2023** |
| **A function takes 2 arguments and returns the maximum. Use this function for finding max of 3 numbers. (use both the concepts of method overloading and reusability)** |

**Code:**

import java.util.Scanner;

class Maxnum

{

int max(int a,int b)

{

if(a > b)

{

return a;

}

else

{

return b;

}

}

int max(int a,int b,int c)

{

if( a >=b )

{

if( a >= c)

{

return a;

}

else

{

return c;

}

}

else

{

if( b >= c)

{

return b;

}

else

{

return c;

}

}

}

public static void main(String[] args)

{

Scanner in=new Scanner(System.in);

Maxnum obj=new Maxnum();

System.out.println("Enter two numbers");

int a=in.nextInt();

int b=in.nextInt();

int c=0;

c=obj.max(a,b);

System.out.println("The maximum of "+a+" and "+b+" is "+c);

System.out.println("Enter three numbers");

a=in.nextInt();

b=in.nextInt();

c=in.nextInt();

int d=0;

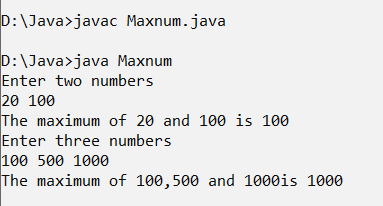
d=obj.max(a,b,c);

System.out.println("The maximum of "+a+","+b+" and "+c+"is " +d);

}

}

**Output:**

****

|  |
| --- |
| **Program 20 Date: 04-01-2023** |
| **WAP to find the factorial of n, using recursion.** |

**Code:**

import java.util.Scanner;

class Fact

{

int fact(int n)

{

if(n!=0) {

return n\*fact(n-1);

}

return 1;

}

public static void main(String[] args)

{

int n,s;

System.out.println("Enter a number");

Scanner scn=new Scanner(System.in);

s=scn.nextInt();

Fact obj=new Fact();

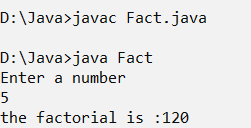
n=obj.fact(s);

System.out.println("the factorial is :" +n);

}

}

**Output:**

****

|  |
| --- |
| **Program 21 Date: 04-01-2023** |
| **WAP to display numbers from n to 1 and vice versa, using recursion.** |

**Code:**

import java.util.Scanner;

class Recurto1

{

int oneton(int m,int n) {

if(n<=m) {

System.out.print(n+ " ");

return (oneton(m,++n));

}

return 1;

}

int ntone(int i)

{

if(i!=0) {

System.out.print(i+ " ");

ntone (i-1);

}

return 1;

}

public static void main(String[] args)

{

int n;

System.out.println("enter the limit");

Scanner scn=new Scanner(System.in);

n=scn.nextInt();

Recurto1 rc=new Recurto1();

System.out.println();

rc.oneton(n,1);

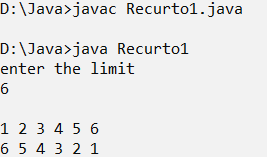
System.out.println();

rc.ntone(n);

}

}

**Output:**

****

|  |
| --- |
| **Program 22 Date: 04-01-2023** |
| **Create a class complex having a real and imaginary part. Provide functions for read, display ,add and multiplying two complex numbers** |

**Code:**

import java.util.\*;

class Complex

{

double r,i;

void read()

{

System.out.println("enter the real part");

Scanner scn=new Scanner(System.in);

r=scn.nextDouble();

System.out.println("Enter the imaginary part");

i=scn.nextDouble();

}

void disp()

{

System.out.println("The real part is: " +r);

System.out.println("The imaginary part is: " +i+"i");

}

void add(Complex c,Complex c1)

{

r=c.r+c1.r;

i=c.i+c1.i;

System.out.println("sum is:" +r+ "+" +i+"i");

}

void mult(Complex c,Complex c1)

{

r=c.r\*c1.r-c.i\*c1.i;

i=c.r\*c1.i+c.i\*c1.r;

System.out.println("Multiplication value is: " +r+ "+" +i+"i");

}

public static void main(String[] args)

{

Complex c=new Complex();

Complex c1=new Complex();

c.read();

c.disp();

c1.read();

c1.disp();

Complex c3=new Complex();

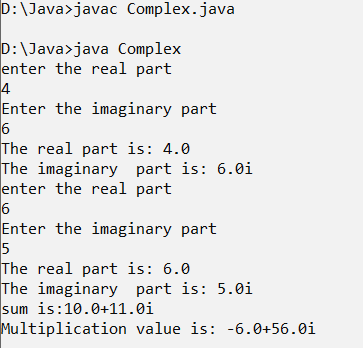
c3.add(c,c1);

c3.mult(c,c1);

}

}

**Output:**

****

|  |
| --- |
| **Program 23 Date: 04-01-2023** |
| **Program to explain static keyword with different usage including function** |

**Code:**

class St

{

static int a=10;

static void disp() {

System.out.println("It's a static function");

}

public static void main(String[] args)

{

St o1=new St();

St.disp();

System.out.println(St.a);

System.out.println(o1.a);

o1.a=15;

System.out.println(St.a);

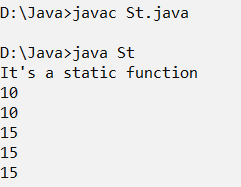
System.out.println(o1.a);

System.out.println(a);

}

}

**Output:**

****

|  |
| --- |
| **Program 24 Date: 04-01-2023** |
| **WAP to display even numbers upto ‘n’ using a static function** |

**Code:**

import java.util.\*;

class Eventon

{

static void st(int n) {

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

if(i%2==0)

{

System.out.println(" " +i);

}

}

}

static Scanner scn=new Scanner(System.in);

public static void main(String[] args)

{

System.out.println("enter the number ");

int n=scn.nextInt();

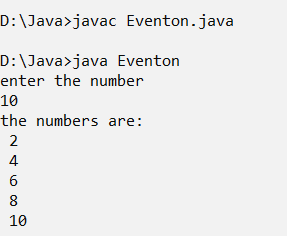
System.out.println("the numbers are:");

Eventon.st(n);

}

}

**Output:**

****

|  |
| --- |
| **Program 25 Date: 06-01-2023** |
| **WAP (menu driven) to demonstrate method overriding in java, by displaying details of a student, and a teacher** |

**Code:**

import java.util.\*;

class Teacher

{

String name,dept;

void read()

{

System.out.println("Enter the teacher name");

Scanner scn=new Scanner(System.in);

name=scn.nextLine();

System.out.println("Enter the dept name");

dept=scn.nextLine();

}

void disp()

{

System.out.println("Name of the teacher: " +name);

System.out.println("Name of department: " +dept);

}

}

class Student extends Teacher

{

void read()

{

System.out.println("Enter the student name");

Scanner scn=new Scanner(System.in);

name=scn.nextLine();

System.out.println("Enter the dept name of the Student");

dept=scn.nextLine();

}

void disp()

{

System.out.println("Name of the student: " +name);

System.out.println("Name of department: " +dept);

}

}

class Overriding

{

public static void main(String[] args)

{

int ch=0;

Scanner sc=new Scanner(System.in);

Teacher obj=new Teacher();

Teacher ob=new Student();

while(ch!=6)

{

System.out.println("1.Enter the teacher details:");

System.out.println("2.Enter the student details:");

System.out.println("3.Display the teacher details:");

System.out.println("4.Display the student details:");

System.out.println("5.Exit");

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

ch=sc.nextInt();

switch(ch) {

case 1:

obj.read();

break;

case 2:

ob.read();

break;

case 3:

obj.disp();

break;

case 4:

ob.disp();

break;

case 5:

break;

default:

System.out.println("Wrong Choice");

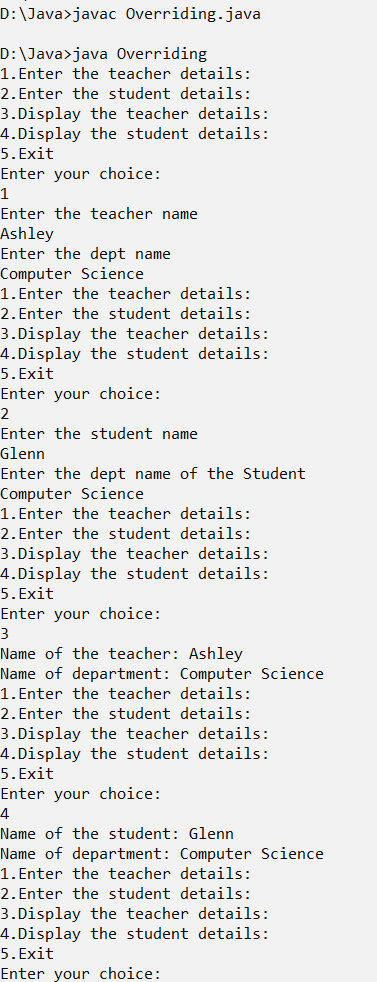
}

}

}

}

**Output:**



|  |
| --- |
| **Program 26 Date: 06-01-2023** |
| **Create a class for employees having eno,ename and esal as data members. Provide functions for reading and displaying employee details. (Accept information of n employees in the main function, display the same and search for an emp (using eno)).** |

**Code:**

import java.util.\*;

class Employee

{

int eno;

int esal;

String ename;

void read(String name,int no,int sal)

{

this.eno=no;

this.esal=sal;

this.ename=name;

}

void disp()

{

System.out.println("employee number is: "+eno);

System.out.println("Name of employee is: "+ename);

System.out.println("Salary of employee: "+esal);

}

public static void main(String[] args)

{

int n,number;

System.out.println("Enter number of employees: ");

Scanner scn=new Scanner(System.in);

n=scn.nextInt();

Employee[] obj=new Employee[n];

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

{

obj[i]=new Employee();

Scanner sc=new Scanner(System.in);

System.out.println("Enter the employee name:");

String name=sc.nextLine();

System.out.println("Enter the employee number:");

int no=sc.nextInt();

System.out.println("Enter the employee salary:");

int sal=sc.nextInt();

obj[i].read(name,no,sal);

obj[i].disp();

}

int ch=0;

while(ch!=3)

{

System.out.println("1.Display all Employees");

System.out.println("2.search Employees");

System.out.println("3.Exit");

System.out.println("Enter your option");

ch=scn.nextInt();

switch(ch)

{

case 1:for(int i=0;i<n;i++)

{

obj[i].disp();

}

break;

case 2: System.out.println("Enter the eno number");

number=scn.nextInt();

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

{

if(obj[i].eno==number)

{

obj[i].disp();

}

}

break;

case 3:break;

default:System.out.println("Wrong option");

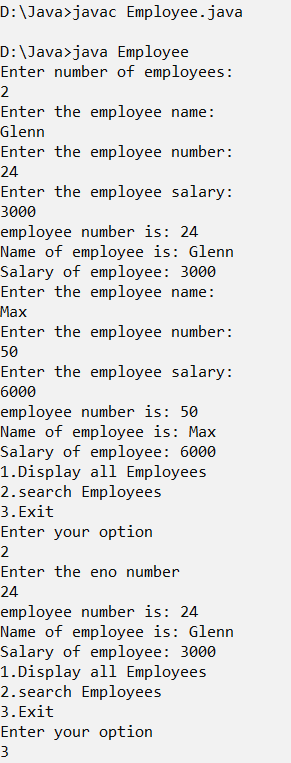
}

}

}

}

**Output:**

****

|  |
| --- |
| **Program 27 Date: 11-01-2023** |
| **Program to implement ISA and HASA relationship.** |

**Code:**

import java.util.Scanner;

class Address

{

String city, state, country;

int pinCode;

public Address(String city, String state, String country, int pinCode)

{

this.city = city;

this.state = state;

this.country = country;

this.pinCode = pinCode;

}

}

class Student

{

String name;

int rollNo;

Address address;

public Student(String name, int rollNo, Address address)

{

this.rollNo = rollNo;

this.name = name;

this.address=address;

}

void display()

{

System.out.println("Name: " +name);

System.out.println("Roll no: " +rollNo);

System.out.println("Address:");

System.out.println(address.city+" "+address.state+" "+address.country+ " " +address.pinCode);

System.out.println("\n");

}

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the name of city");

String city=sc.nextLine();

System.out.println("Enter the name of state");

String state=sc.nextLine();

System.out.println("Enter the name of country");

String country=sc.nextLine();

System.out.println("Enter the pincode");

int pincode=sc.nextInt();

System.out.println("Enter the name ");

String name=sc.next();

System.out.println("Enter the rollno");

int rollNo=sc.nextInt();

Address addres = new Address(city,state,country,pincode);

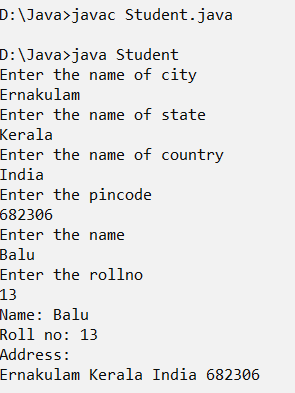
Student st = new Student(name,rollNo,addres);

st.display();

}

}

**Output:**

****

|  |
| --- |
| **Program 28 Date: 11-01-2023** |
| **Program to overcome function overriding in java** |

**Code:**

class Animal

{

Animal()

{

System.out.println("Animal cons");

}

final void eat()

{

System.out.println("Animal eats");

}

}

class Cat extends Animal

{

Cat()

{

System.out.println("Cat cons");

}

void eat1()

{

System.out.println("Cat eats");

}

}

class Funcov

{

public static void main(String args[])

{

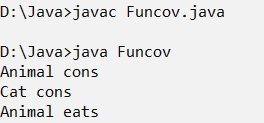
Cat c=new Cat();

c.eat();

}

}

**Output:**

****

|  |
| --- |
| **Program 29 Date: 11-01-2023** |
| **Program to implement run time polymorphism in Java using Interface, wrt calculating area of a triangle.** |

**Code:**

import java.util.Scanner;

interface Shape{

void cal(int b,int h);

}

class Triangle implements Shape

{

public void cal(int b,int h)

{

float ans=(h\*b)/2;

System.out.println("The area of the triangle:"+ans);

}

}

class Area

{

public static void main(String[] args)

{

Scanner in=new Scanner(System.in);

Shape obj=new Triangle();

System.out.println("Enter b");

int b=in.nextInt();

System.out.println("Enter h");

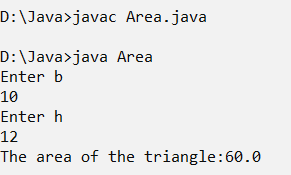
int h=in.nextInt();

obj.cal(b,h);

}

}

**Output:**



|  |
| --- |
| **Program 30 Date: 11-01-2023** |
| **Create an interface Shape having two prototypes disp() and calc(), to display the shape and calculate area respectively. Create two classes: circle and rectangle which implements the above interface. In the main function create a reference of Shape depending on the user-choice.** |

**Code:**

import java.util.Scanner;

interface Shape

{

public void disp();

public void calc();

}

class Circle implements Shape

{

int radius;

Circle(int radius)

{

this.radius = radius;

}

public void disp()

{

System.out.println("Shape is Circle");

}

public void calc()

{

double area = 3.14\*radius\*radius;

System.out.println("The area of the Circle with radius " + radius + " is " + area);

}

}

class Rectangle implements Shape

{

int l, b;

Rectangle(int l, int b) {

this.l = l;

this.b = b;

}

public void disp()

{

System.out.println("Shape is Rectangle");

}

public void calc()

{

double area = l \* b;

System.out.println("The area of the rectangle with l=" + l + " and b=" + b + " is " + area);

}

}

class Area

{

public static void main(String[] args)

{

Scanner sc = new Scanner(System.in);

Shape sh=null;

int ch=0;

while(ch!=3)

{

System.out.println("Select your choice ");

System.out.println("1. Circle");

System.out.println("2. Rectangle");

System.out.println("3. Exit");

System.out.print("Enter your Choice: ");

ch = sc.nextInt();

switch (ch)

{

case 1:

System.out.print("Enter the radius of the Circle: ");

sh = new Circle(sc.nextInt());

break;

case 2:

System.out.print("Enter the length and breath of the Rectangle: ");

sh = new Rectangle(sc.nextInt(),sc.nextInt());

case 3: break;

default:System.out.println("Enter the choice");

}

if(sh != null)

{

sh.calc();

}

else

{

System.out.println("Invalid");

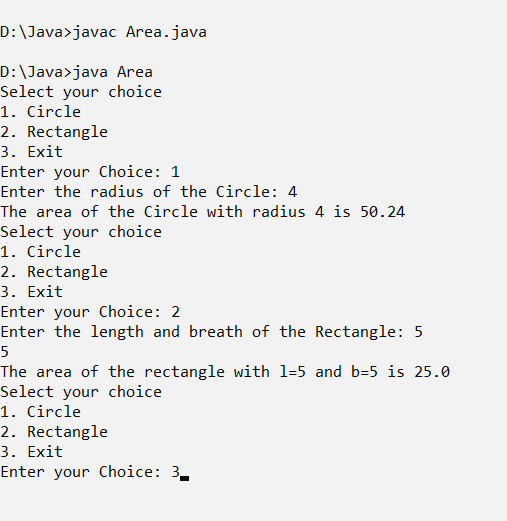
}

}

}

}

**Output:**

****

|  |
| --- |
| **Program 31 Date: 11-01-2023** |
| **WAP to implement a function using call by value to swap two float numbers.** |

**Code:**

import java.util.Scanner;

class num

{

void test(float a,float b)

{

float c=a;

a=b;

b=c;

System.out.println("After Swapping :: a = "+a+" b = "+b);

}

}

class CBVSwap

{

public static void main(String args[])

{

num n=new num();

float a,b;

Scanner in=new Scanner(System.in);

System.out.println("Enter the values for a and b:");

a=in.nextFloat();

b=in.nextFloat();

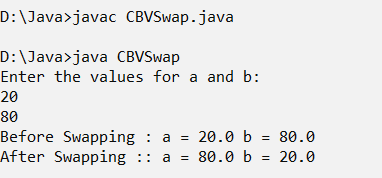
System.out.println("Before Swapping : a = "+a+" b = "+b);

n.test(a,b);

}

}

**Output:**

****

|  |
| --- |
| **Program 32 Date: 11-01-2023** |
| **WAP to implement a function using call by reference to find the square root of a given number.** |

**Code:**

import java.util.Scanner;

import java.lang.Math;

class Sqt

{

double x;

double sqr(Sqt obj)

{

double s=Math.sqrt(x);

return s;

}

public static void main(String args[])

{

Scanner sc=new Scanner(System.in);

Sqt obj=new Sqt();

System.out.println("Enter a number:");

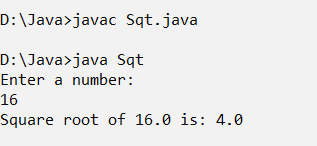
obj.x=sc.nextInt();

System.out.println("Square root of "+obj.x+" is: "+ obj.sqr(obj));

}

}

**Output:**

****

|  |
| --- |
| **Program 33 Date: 13-01-2023** |
| **Create a class for Cstring having a string data member and provide functions for read, display, compare (return Boolean value), add and concatenate.** |

**Code:**

import java.util.\*;

class Cstring

{

String name;

void read(String name)

{

this.name=name;

}

void display()

{

System.out.println("Name is :" +name);

}

void compare(String name1)

{

System.out.println(name.equals(name1));

}

void concatenate(String n)

{

System.out.println(name + n);

}

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter first name");

String name=sc.nextLine();

Cstring obj=new Cstring();

obj.read(name);

obj.display();

System.out.println("Enter second name");

String name1=sc.nextLine();

obj.compare(name1);

System.out.println("Enter the name to add");

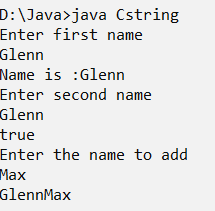
String n=sc.nextLine();

obj.concatenate(n);

}

}

**Output:**

****

|  |
| --- |
| **Program 34 Date: 13-01-2023** |
| **Write a program to implement object cloning for the class Distance which has inch and feet as data members.** |

**Code:**

import java.util.\*;

class distance

{

int inch;

int ft;

}

class Clone

{

public static void main(String[] args)

{

distance dis=new distance();

System.out.println("Enter Measure(inch)");

Scanner scn=new Scanner(System.in);

dis.inch=scn.nextInt();

System.out.println("Enter Measure(ft)");

dis.ft=scn.nextInt();

distance dis1=new distance();

dis1=dis;

System.out.println("1st object");

System.out.println(dis.inch+ " and" +dis.ft);

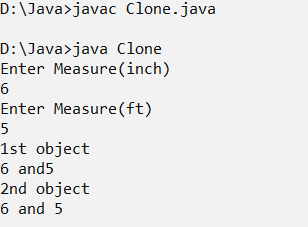
System.out.println("2nd object");

System.out.println(dis1.inch+ " and " +dis1.ft);

}

}

**Output:**

****

|  |
| --- |
| **Program 35 Date: 13-01-2023** |
| **Write a menu driven program for performing the following operations.**  **a. Length of a given string**  **b. Compare for equality**  **c. Extract a substring from a string.**  **d. Convert to uppercase and lowercase** |

**Code:**

import java.util.Scanner;

class Strings\_7\_3

{

String value;

Scanner in=new Scanner(System.in);

Strings\_7\_3(String v1)

{

value=v1;

}

void length()

{

System.out.println("The length is "+value.length());

}

void compare(String s1)

{

System.out.println(value.equals(s1));

}

void subsring()

{

int start,end;

System.out.println("Enter the strating index");

start=in.nextInt();

System.out.println("The substrings is "+value.substring(start));

System.out.println("Enter the strating and ending index");

start=in.nextInt();

end=in.nextInt();

System.out.println("The substrings starting from "+start+" to "+end+" is "+value.substring(start,end));

}

void convert()

{

System.out.println("The upper case is "+value.toUpperCase());

System.err.println("The lower case is "+value.toLowerCase());

}

}

class Mainstring

{

public static void main(String[] arsg)

{

Scanner sc=new Scanner(System.in);

System.out.println("Enter the string");

String s1=sc.next();

Strings\_7\_3 obj=new Strings\_7\_3(s1);

int ch=0;

while(ch!=5)

{

System.out.println("1.Length of the String");

System.err.println("2.Compare");

System.err.println("3.Extract a substring");

System.err.println("4.converting case");

System.err.println("5.Exit");

System.out.println("Enter your option");

ch=sc.nextInt();

switch(ch)

{

case 1: obj.length();

break;

case 2: System.out.println("Enter a String to compare");

String s2=sc.next();

obj.compare(s2);

break;

case 3: obj.subsring();

break;

case 4: obj.convert();

break;

case 5: break;

default:System.out.println("Wrong option");

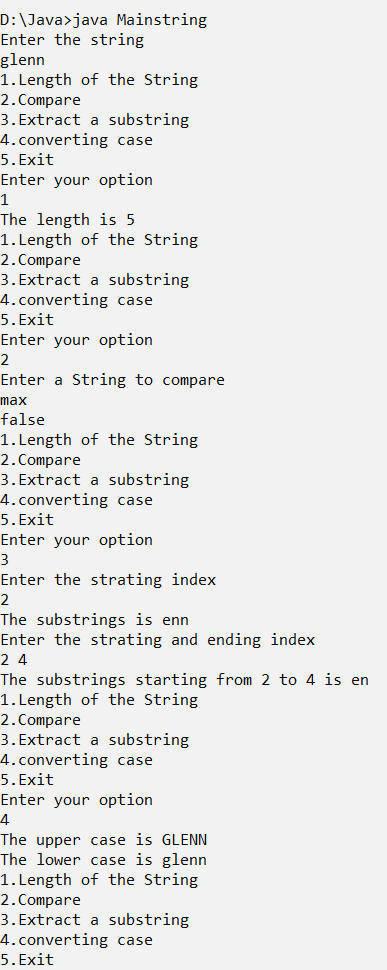
}

}

}

}

**Output:**

****

|  |
| --- |
| **Program 36 Date: 13-01-2023** |
| **Write a program to reverse a string** |

**Code:**

import java.util.Scanner;

class Stringss

{

String s1;

Stringss(String value)

{

s1=value;

}

void display()

{

System.out.println("The string is "+s1);

}

void revdisplay()

{

String revstr="";

for(int i=0;i<s1.length();i++)

{

char ch=s1.charAt(i);

revstr=ch+revstr;

}

System.out.println("The reverse "+revstr);

}

}

class Reverse\_7\_4

{

public static void main(String[] args)

{

Scanner in=new Scanner(System.in);

System.out.print("Enter the string: ");

String str=in.next();

Stringss obj=new Stringss(str);

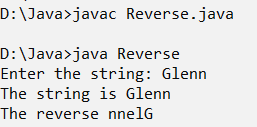
obj.display();

obj.revdisplay();

}

}

**Output:**

****

|  |
| --- |
| **Program 37 Date: 13-01-2023** |
| **Write a program to calculate the prime factors of a given number, using packages.** |

**Code:**

import java.util.\*;

public class PrimeFactors {

public static List<Integer> primeFactors(int number) {

List<Integer> factors = new ArrayList<Integer>();

for (int i = 2; i <= number; i++) {

while (number % i == 0) {

factors.add(i);

number /= i;

}

}

return factors;

}

}

import java.util.\*;

public classMain {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int number = sc.nextInt();

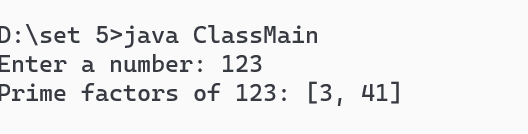
List<Integer> factors = PrimeFactors.primeFactors(number);

System.out.println("Prime factors of " + number + ": " + factors);

}

}

**Output:**

****

|  |
| --- |
| **Program 38 Date: 18-01-2023** |
| **Read numbers into an array. Perform validations using multiple catch statements / predefined Exceptions.** |

**Code:**

import java.util.Scanner;

class TCarr

{

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

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

int x=sc.nextInt();

int array[]=new int[x];

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

for(int i=0;i<x;i++)

{

try

{

array[i]=sc.nextInt();

}catch(NumberFormatException n)

{

System.out.println("Enter a valid number");

}catch(Exception e)

{

System.out.println("Enter a valid number");

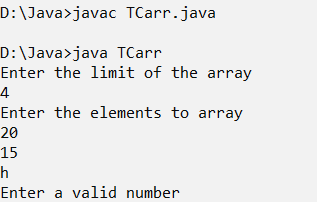
}

}

}

}

**Output:**

****

|  |
| --- |
| **Program 39 Date: 18-01-2023** |
| **Write a program to implement a user defined Exception, which will throw an Exception when a given number is prime.** |

**Code:**

import java.io.\*;

import java.util.Scanner;

class PrimeNumberException extends Exception{

public PrimeNumberException(String message) {

super(message);

}

}

class Prime {

static Scanner sc = new Scanner(System.in);

static void checkPrime(int n) throws PrimeNumberException {

if (n == 1 || n == 3 || n == 2) {

throw new PrimeNumberException("The Number is a Prime Number");

} else if (((n \* n) - 1) % 24 == 0) {

throw new PrimeNumberException("The Number is a Prime Number");

} else {

System.out.println("The Number is not a Prime Number");

}

}

public static void main(String[] args) {

int n, ch;

while (true) {

System.out.println("1. Enter a Number");

System.out.println("2. Exit");

ch = sc.nextInt();

if (ch == 2) {

break;

}

try {

n = sc.nextInt();

checkPrime(n);

} catch (PrimeNumberException p) {

System.out.println("Error: " + p.getMessage());

} catch (Exception e) {

System.out.println("An Error has Occurred! Try Again!");

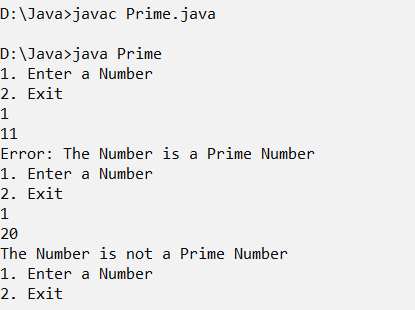
}

}

}

}

**Output:**



|  |
| --- |
| **Program 40 Date: 18-01-2023** |
| **Write a program to implement throw and finally.** |

**Code:**

import java.util.\*;

class Tf

{

public static void main(String[] args)

{

Scanner scn=new Scanner(System.in);

System.out.println("Enter a number");

try

{

int n=scn.nextInt();

}catch(Exception e)

{

System.out.println("Enter a valid number");

}finally

{

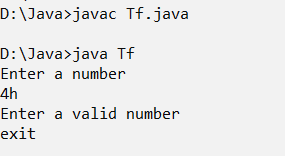
System.out.println("exit");

}

}

}

**Output:**

****

|  |
| --- |
| **Program 41 Date: 20-01-2023** |
| **Write a program to create multiple threads by extending the Thread class.** |

**Code:**

class Multi extends Thread

{

int n;

Multi(int n)

{

this.n = n;

}

public void run()

{

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

{

System.out.println(i);

try

{

Thread.sleep(1000);

} catch (InterruptedException e)

{

System.out.println("Something went Wrong "+e.getMessage());

}

}

}

public static void main(String[] args)

{

Multi t1 = new Multi(5);

Multi t2 = new Multi(15);

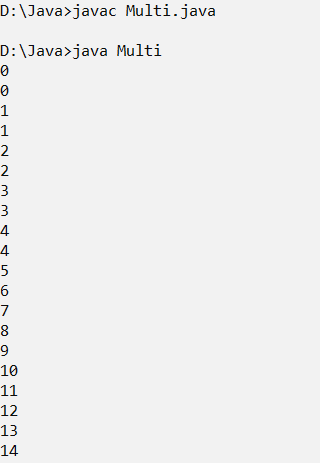
new Thread(t1).start();

new Thread(t2).start();

}

}

**Output:**

****

|  |
| --- |
| **Program 42 Date: 20-01-2023** |
| **Write a program to implement threads by implementing the Runnable interface.** |

**Code:**

class Implementthreads implements Runnable

{

int n;

Implementthreads(int n)

{

this.n = n;

}

public void run()

{

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

{

System.out.println(i);

try

{

Thread.sleep(1000);

} catch (InterruptedException e)

{

System.out.println("Something went Wrong " + e.getMessage());

}

}

}

public static void main(String[] args) {

Implementthreads s1 = new Implementthreads(4);

Implementthreads s2 = new Implementthreads(10);

new Thread(s1).start();

new Thread(s2).start();

}

}

**Output:**

****

|  |
| --- |
| **Program 43 Date: 20-01-2023** |
| **Write a program to implement Synchronization using inter-thread communication.** |

**Code:**

class Stud

{

void print(int n)

{

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

{

System.out.println(i);

}

}

}

class Thread1 extends Thread

{

Stud s;

Thread1(Stud s)

{

this.s=s;

}

public void run()

{

s.print(5);

}

}

class Thread2 extends Thread

{

Stud s;

Thread2(Stud s)

{

this.s=s;

}

public void run()

{

s.print(8);

}

}

class Thread3 extends Thread

{

Stud s;

Thread3(Stud s)

{

this.s=s;

}

public void run()

{

s.print(6);

}

}

public class Exam2

{

public static void main(String[] args)

{

Stud obj=new Stud();

Thread1 t=new Thread1(obj);

Thread2 t1=new Thread2(obj);

Thread3 t2=new Thread3(obj);

t.start();

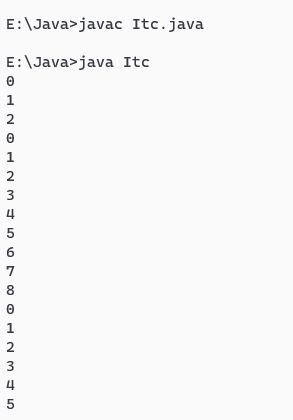
t1.start();

t2.start();

}

}

**Output:**

****

|  |
| --- |
| **Program 44 Date: 20-01-2023** |
| **Implement the Producer- Consumer Problem, using Threads.** |

**Code:**

public class ProducerConsumerTest {

public static void main(String[] args) {

Container c = new Container();

Producer p1 = new Producer(c, 1);

Consumer c1 = new Consumer(c, 1);

p1.start();

c1.start();

}

}

class Container {

private int contents;

private boolean available = false;

public synchronized int get() {

while (available == false) {

try {

wait();

} catch (InterruptedException e) {}

}

available = false;

notifyAll();

return contents;

}

public synchronized void put(int value) {

while (available == true) {

try {

wait();

} catch (InterruptedException e) { }

}

contents = value;

available = true;

notifyAll();

}

}

class Consumer extends Thread {

private Container con;

private int number;

public Consumer(Container c, int number) {

con = c;

this.number = number;

}

public void run() {

int value = 0;

for (int i = 1; i < 10; i++) {

value = con.get();

System.out.println("Consumer #" + this.number + " got: " + value);

}

}

}

class Producer extends Thread {

private Container con;

private int number;

public Producer(Container c, int number) {

con = c;

this.number = number;

}

public void run() {

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

con.put(i);

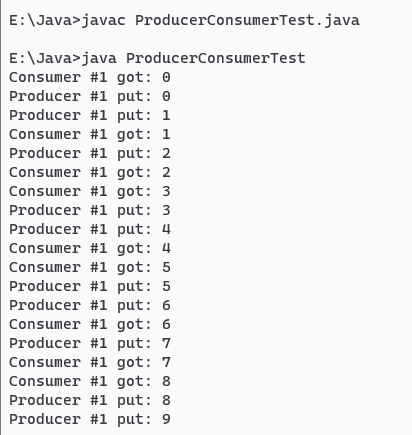
System.out.println("Producer #" + this.number + " put: " + i);

}

}

}

**Output:**

****

|  |
| --- |
| **Program 45 Date: 25-01-2023** |
| **Write a program to display the contents of a directory by displaying the subdirectory’s name first, then the file names.** |

**Code:**

import java.io.\*;

class Contents

{

public void set(File[] a,int i,int lvl)

{

if(i == a.length)

{

return;

}

for(int j=0;j<lvl;j++)

{

System.out.println("\t");

}

if(a[i].isFile())

{

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

}

else if(a[i].isDirectory())

{

System.out.println("[" +a[i].getName() + "]");

set(a[i].listFiles(),0,lvl+1);

}

set(a,i+1,0);

}

public static void main(String[] args)

{

String st = "P:\\CAE 2";

File fi = new File(st);

Contents cs = new Contents();

if(fi.exists() && fi.isDirectory())

{

File a[] = fi.listFiles();

System.out.println("Files are:" +fi);

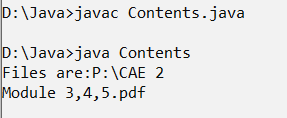
cs.set(a,0,0);

}

}

}

**Output:**

****

|  |
| --- |
| **Program 46 Date: 25-01-2023** |
| **Write a program to search for a given file name in a directory** |

**Code:**

import java.io.\*;

import java.util.Scanner;

class Search implements FilenameFilter

{

String name;

public Search(String name)

{

this.name = name;

}

public boolean accept(File dir, String name1)

{

return name1.startsWith(name);

}

}

class Sear {

public static void main(String[] args)

{

Scanner in=new Scanner(System.in);

System.out.println("Enter the path");

File directory = new File(in.nextLine());

System.out.println("Enter the name of file with extension");

Search filter= new Search(in.nextLine());

String[] flist = directory.list(filter);

if (flist == null)

{

System.out.println("Empty directory or directory does not exists.");

}

else

{

for (int i = 0; i < flist.length; i++)

{

System.out.println(flist[i]+" found");

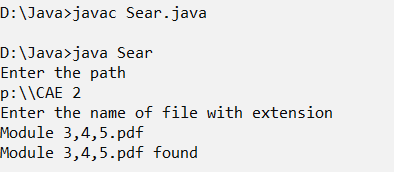
}

}

}

}

**Output:**



|  |
| --- |
| **Program 47 Date: 25-01-2023** |
| **Write a program to search for a given string in a file.** |

**Code:**

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.IOException;

import java.util.Scanner;

public class Searchst {

static Scanner sc = new Scanner(System.in);

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

File f = new File("test.txt");

if(!f.exists()){

FileOutputStream fos = new FileOutputStream("test.txt");

String message = "Computer Science";

fos.write(message.getBytes());

fos.close();

}

FileInputStream fis = new FileInputStream("test.txt");

String content = new String(fis.readAllBytes());

if(content.contains("is")){

System.out.println("Word is found");

}else{

System.out.println("Word not found");

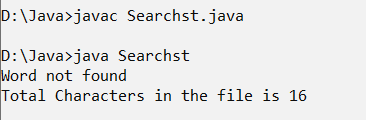
}

System.out.println("Total Characters in the file is "+f.length());

}

}

**Output:**

****

|  |
| --- |
| **Program 48 Date: 25-01-2023** |
| **Write a program to find the number of characters, number of words and number of lines in a given file** |

**Code:**

import java.io.File;

import java.io.FileInputStream;

import java.io.FileOutputStream;

import java.io.IOException;

public class CFile{

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

File f1 = new File("sample.txt");

File f2 = new File("final.txt");

if(!f2.exists()){

f2.createNewFile();

}

FileInputStream fis = new FileInputStream(f1);

FileOutputStream fos = new FileOutputStream(f2);

byte[] f1\_data = fis.readAllBytes();

fos.write(f1\_data);

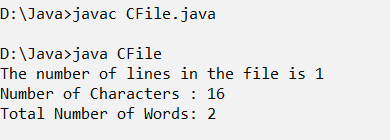
fis.close();

fos.close();

}

}

**Output:**

****

|  |
| --- |
| **Program 49 Date: 25-01-2023** |
| **Write a program to accept two filenames, copy the content from the first file to the second file** |

**Code:**

import java.io.\*;

import java.util.\*;

public class Copy

{

public static void copyContent(File a, File b)

throws Exception

{

FileInputStream in = new FileInputStream(a);

FileOutputStream out = new FileOutputStream(b);

try {

int n;

// read() function to read the

// byte of data

while ((n = in.read()) != -1) {

// write() function to write

// the byte of data

out.write(n);

}

}

finally {

if (in != null) {

// close() function to close the

// stream

in.close();

}

// close() function to close

// the stream

if (out != null) {

out.close();

}

}

System.out.println("File Copied");

}

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

{

Scanner sc = new Scanner(System.in);

System.out.println("Enter the source filename from where you have to read/copy :");

String a = sc.nextLine();

// source file

File x = new File(a);

// get the destination file name

System.out.println("Enter the destination filename where you have to write/paste :");

String b = sc.nextLine();

// destination file

File y = new File(b);

// method called to copy the

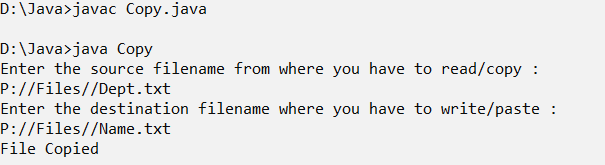
// contents from x to y

copyContent(x, y);

}

}

**Output:**

****

|  |
| --- |
| **Program 50 Date: 27-01-2023** |
| **Write a menu driven program to demonstrate Random Access File handling, with options for creating, deleting, writing, appending and reading the file.** |

**Code:**

import java.io.File;

import java.io.FileNotFoundException;

import java.io.RandomAccessFile;

import java.util.Scanner;

class Randomfile

{

static Scanner sc = new Scanner(System.in);

static void menu()

{

System.out.println("1. Create a File");

System.out.println("2. Delete a File");

System.out.println("3. Write a File");

System.out.println("4. Append to File");

System.out.println("5. Read File");

System.out.println("6. Exit");

System.out.print("Enter your Choice: ");

}

RandomAccessFile rf;

String fileName;

Randomfile(String fileName){

try{

this.fileName = fileName;

rf = new RandomAccessFile(this.fileName,"rw");

}catch (FileNotFoundException fnf){

this.create();

}

catch(Exception e){

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

System.exit(0);

}

}

void create(){

try{

System.out.println("Cresting file "+this.fileName);

File f = new File(this.fileName);

if(f.exists()){

System.out.println("File with the Name Already Exists!");

}else{

if(f.createNewFile()){

System.out.println("File Created Successfully!");

}else{

throw new Exception("Error Creating File!");

}

}

}catch(Exception e){

System.out.println("Error Creating File...");

System.out.println("Error: "+e.getMessage());

}

}

void delete(){

try{

File f = new File(this.fileName);

rf.close();

if(f.delete()){

System.out.println("File Deleted Successfully!");

}

}catch (Exception e){

System.out.println("Something went Wrong");

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

}

}

void write(){

System.out.println("Enter the Content to write:");

sc.nextLine();

String content = sc.nextLine();

try{

this.rf.write(content.getBytes());

System.out.println("Successfully wrote to File!");

}catch(Exception e){

System.out.println("Error Writing file");

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

}

}

void append(){

System.out.println("Enter the Contents to Append into File: ");

sc.nextLine();

String message = sc.nextLine();

try{

long length = rf.length();

rf.seek(length);

rf.write(message.getBytes());

System.out.println("Append Successful!");

}catch(Exception e){

System.out.println("Error Appending File!");

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

}

}

void read(){

try{

String message = rf.readLine();

System.out.println("Contents of the file are: ");

System.out.println(message);

}catch(Exception e){

System.out.println("Error Reading File!");

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

}

}

public static void main(String[] args) {

System.out.print("Enter the Name of the file: ");

String fileName;

fileName = sc.nextLine();

Randomfile obj = new Randomfile(fileName);

int ch;

boolean isRunning = true;

while(isRunning){

menu();

ch = sc.nextInt();

switch(ch){

case 1:obj.create();break;

case 2:obj.delete();break;

case 3:obj.write();break;

case 4:obj.append();break;

case 5:obj.read();break;

case 6:isRunning = false; break;

default:System.out.println("Enter a Valid Choice");

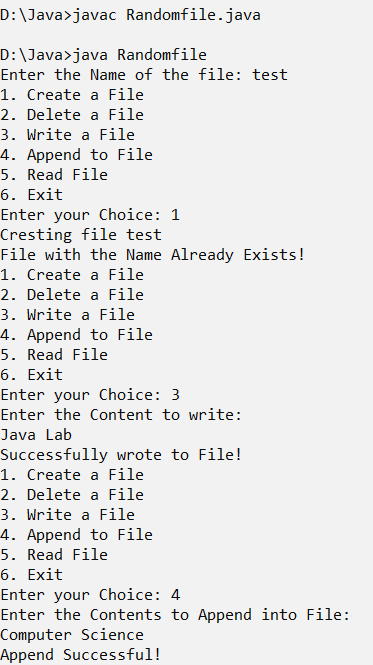
}

}

}

}

**Output:**

****

|  |
| --- |
| **Program 51 Date: 27-01-2023** |
| **Write a program to implement a Generic method, which can display the elements of various arrays of different data types, and find the length of each array.** |

**Code:**

class Myclass

{

public static <T> void display(T[] array)

{

for(T t : array)

{

System.out.print(t+ " ");

}

System.out.println();

System.out.println("The length of array is "+array.length);

}

}

class Gen{

public static void main(String[] args)

{

Integer[ ] arr1 = {10,20,30};

System.out.println("Integers: ");

Myclass.display(arr1);

Double[ ] arr2 = {1.1, 2.2, 3.3};

System.out.println("Double: ");

Myclass.display(arr2);

String[ ] arr3 = {"Hello","How are you"};

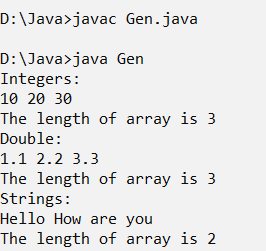
System.out.println("Strings: ");

Myclass.display(arr3);

}

}

**Output:**

****

|  |
| --- |
| **Program 52 Date: 27-01-2023** |
| **Write a program to implement a Generic class, and display the types of various parameters passed** |

**Code:**

class GenericClass<T, U>

{

T ob1;

U ob2;

GenericClass(T ob1, U ob2)

{

this.ob1 = ob1;

this.ob2 = ob2;

}

public void displayTypes()

{

System.out.println("Type of T is " + ob1.getClass().getName());

System.out.println("Type of U is " + ob2.getClass().getName());

}

}

class Param

{

public static void main(String[] args)

{

GenericClass<Integer, String> ob

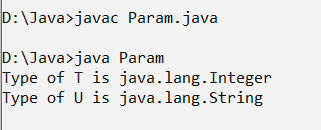
= new GenericClass<Integer, String>(15, "hello");

ob.displayTypes();

}

}

**Output:**

****

|  |
| --- |
| **Program 53 Date: 03-02-2023** |
| **Program to implement Serialization and DeSerialization, for an object of Student Class** |

**Code:**

import java.io.\*;

import java.util.\*;

class Student implements Serializable

{

String name;

String course;

int age;

Student(String name,String course)

{

this.name=name;

this.course=course;

}

}

class Student1 extends Student{

int age;

public Student1(String name,String course,int age)

{

super(name,course);

this.age=age;

}

}

class Sedes

{

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

{

Student1 st=new Student1("Balu","Msc",13);

FileOutputStream fos=new FileOutputStream("D:\\Student.ser");

ObjectOutputStream oos=new ObjectOutputStream(fos);

oos.writeObject(st);

oos.close();

fos.close();

FileInputStream fis=new FileInputStream("D:\\Student.ser");

ObjectInputStream ois=new ObjectInputStream(fis);

Student1 st1=(Student1)ois.readObject();

ois.close();

fis.close();

System.out.println("Student name is:" +st1.name);

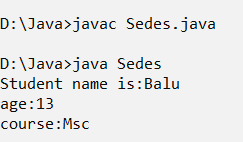
System.out.println("age:" +st1.age);

System.out.println("course:" +st1.course);

}

}

**Output:**

****

|  |
| --- |
| **Program 54 Date: 03-02-2023** |
| **Program to implement IS A Serialization and DeSerialization, for a Maruti Car inherited from Vehicle** |

**Code:**

import java.io.\*;

import java.util.\*;

class Vehicle implements Serializable

{

String name;

Vehicle(String s)

{

name=s;

}

}

class Car extends Vehicle

{

String model;

int number;

Car(String name,String model,int number)

{

super(name);

this.model=model;

this.number=number;

}

}

class Isedes

{

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

{

Car c=new Car("Maruthi","VXI", 2314);

System.out.println("name is:" +c.name);

System.out.println("model is:" +c.model);

System.out.println("number is:" +c.number);

FileOutputStream fos=new FileOutputStream("files.ser");

ObjectOutputStream oos=new ObjectOutputStream(fos);

oos.writeObject(c);

oos.close();

fos.close();

System.out.println("Serialized");

FileInputStream fis=new FileInputStream("files.ser");

ObjectInputStream ois=new ObjectInputStream(fis);

Car c1=(Car)ois.readObject();

ois.close();

fis.close();

System.out.println("Deserialized");

System.out.println("name is:" +c1.name);

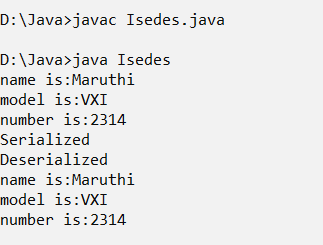
System.out.println("model is:" +c1.model);

System.out.println("number is:" +c1.number);

}

}

**Output:**

****

|  |
| --- |
| **Program 55 Date: 03-02-2023** |
| **Write a program to implement HAS-A Serialization and De- Serialization for the Engine of a Vehicle.** |

**Code:**

import java.io.\*;

class Engine implements Serializable{

String model;

int capacity;

boolean isPetrol;

public Engine(String model, int capacity, boolean isPetrol) {

this.model = model;

this.capacity = capacity;

this.isPetrol = isPetrol;

}

}

class Vehicle\_New implements Serializable {

String name;

Engine eng;

public Vehicle\_New(String name, Engine eng) {

this.name = name;

this.eng = eng;

}

void getInfo(){

System.out.println(String.format("Engine %s and name is %s",eng.model,name));

}

}

public class Hsedes{

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

Engine en = new Engine("XL",30,true);

Vehicle\_New car = new Vehicle\_New("Benz Car",en);

car.getInfo();

System.out.println("Serializing...");

FileOutputStream fos = new FileOutputStream("alto.ser");

ObjectOutputStream oos = new ObjectOutputStream(fos);

oos.writeObject(car);

System.out.println("Serialization Successful!");

oos.close();

fos.close();

System.out.println("DeSerializing...");

FileInputStream fis = new FileInputStream("alto.ser");

ObjectInputStream ois = new ObjectInputStream(fis);

Vehicle\_New new\_car = (Vehicle\_New) ois.readObject();

System.out.println("DeSerialization Successful!");

new\_car.getInfo();

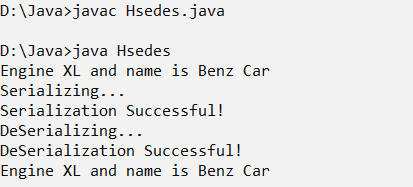
ois.close();

fis.close();

}

}

**Output:**

****

|  |
| --- |
| **Program 56 Date: 03-02-2023** |
| **Write a program to Serialize/De-Serialize selected attributes of an Employee.** |

**Code:**

import java.io.\*;

import java.util.\*;

class Student implements Serializable

{

String name;

String dept;

int age;

}

class Sd

{

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

{

Student s = new Student();

Scanner scn=new Scanner(System.in);

s.name=scn.nextLine();

s.dept=scn.nextLine();

s.age=scn.nextInt();

System.out.println("Name is:" +s.name);

System.out.println("Age is:" +s.age);

System.out.println("Dept is:" +s.dept);

FileOutputStream fos=new FileOutputStream("file5.ser");

ObjectOutputStream oos=new ObjectOutputStream(fos);

oos.writeObject(s);

oos.close();

fos.close();

System.out.println("Serialized");

FileInputStream fis=new FileInputStream("file5.ser");

ObjectInputStream ois=new ObjectInputStream(fis);

Student s1=(Student)ois.readObject();

ois.close();

fis.close();

System.out.println("Deserialized");

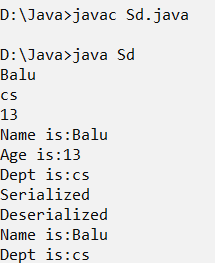
System.out.println("Name is:" +s1.name);

System.out.println("Dept is:" +s1.dept);

}

}

**Output:**



|  |
| --- |
| **Program 57 Date: 08-02-2023** |
| **Write a program to implement various methods of a StringBuffer class.** |

**Code:**

import java.util.\*;

class Pgm\_13\_1

{

public static void main(String args[])

{

int ch=0;

Scanner in=new Scanner(System.in);

System.out.println("Enter the value:");

String value=in.next();

StringBuffer sb=new StringBuffer(value);

while(ch!=8)

{

System.out.println("1.Show");

System.out.println("2.Append");

System.out.println("3.Insert");

System.out.println("4.Replace");

System.out.println("5.Delete");

System.out.println("6.Reverse");

System.out.println("7.Capacity ");

System.out.println("8.Exit ");

System.out.println("Enter your option:");

ch=in.nextInt();

switch(ch)

{

case 1:

System.out.println(sb);

break;

case 2:

System.out.println("Enter the value to append:");

sb.append(in.next());

System.out.println(sb);

break;

case 3: System.out.println("Enter the value to insert:");

String v1=in.next();

System.out.println("Enter the position:");

int position=in.nextInt();

sb.insert(position,v1);

System.out.println(sb);

break;

case 4: System.out.print("Ente the value to replace:");

String v2=in.next();

System.out.println("Enter the beginning position and ending position:");

int position1=in.nextInt();

int position2=in.nextInt();

sb.replace(position1, position2, v2);

System.out.println(sb);

break;

case 5: System.out.println("Enter the beginning position and ending position to delete:");

position1=in.nextInt();

position2=in.nextInt();

sb.delete(position1, position2);

System.out.println(sb);

break;

case 6: System.out.println(sb.reverse());

break;

case 7: System.out.println(sb.capacity());

break;

case 8: break;

default: System.out.println("Worong option");

}

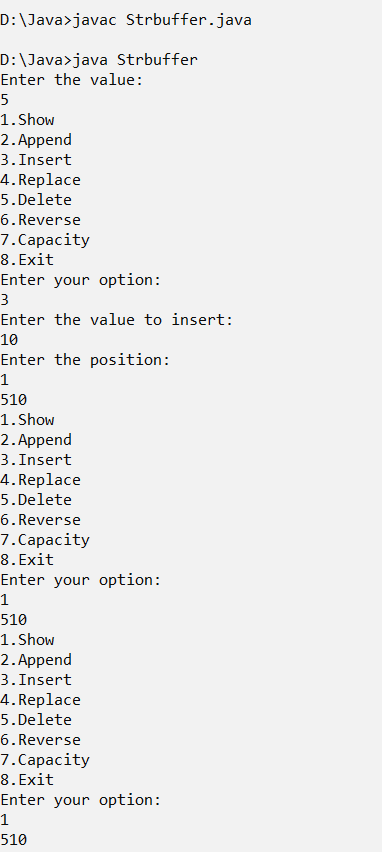
}

in.close();

}

}

**Output:**



|  |
| --- |
| **Program 58 Date: 08-02-2023** |
| **Write a program to implement communication between a client and server via Socket Programming** |

**Code:**

import java.io.\*;

import java.net.\*;

class Myclient

{

public static void main(String[] args)

{

try

{

Socket s=new Socket("localhost",6666);

DataOutputStream dout=new DataOutputStream(s.getOutputStream());//write to an output source

dout.writeUTF("fine");

DataInputStream dis=new DataInputStream(s.getInputStream());

String str=(String)dis.readUTF();

System.out.println("message: "+str);

dout.flush();

dout.close();

s.close();

}

catch(Exception e)

{

System.out.println(e);

}

}

}

import java.io.\*;

import java.net.\*;

public class Server {

public static void main(String[] args){

try{

ServerSocket ss=new ServerSocket(6666);

Socket s=ss.accept();//establishes connection

DataInputStream dis=new DataInputStream(s.getInputStream()); //to read data from client

DataOutputStream dout=new DataOutputStream(s.getOutputStream());

String str=(String)dis.readUTF();

if(str !=null)

{

System.out.println("message: "+str);

dout.writeUTF("How are you");

}

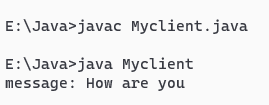
ss.close();

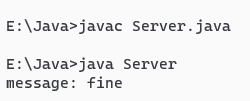
}catch(Exception e){System.out.println(e);}

}

}

**Output:**





|  |
| --- |
| **Program 59 Date: 15-02-2023** |
| **Write a program to implement one-one chatting using the TCP protocol** |

**Code:**

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class Soccli

{

public static void main(String[] args)

{

try

{

Socket s=new Socket("localhost",4444);

Scanner dis=new Scanner(System.in); //read from keyboard

DataInputStream di=new DataInputStream(s.getInputStream());

DataOutputStream dos=new DataOutputStream(s.getOutputStream());

String str;

while(true)

{

dos.writeUTF("hai");

str=di.readUTF();

if(str.equals("bye"))

break;

System.out.println("Server says " + str);

System.out.println("Enter data for server");

str= dis.nextLine();

dos.writeUTF(str);

if(str.equals("bye"))

break;

}

s.close();

}catch(Exception e)

{

System.out.println(e);

}

}

}

import java.io.\*;

import java.net.\*;

import java.util.\*;

public class Socser

{

public static void main(String[] args)

{

try

{

ServerSocket ss=new ServerSocket(4444);

Socket s=ss.accept();

Scanner din=new Scanner(System.in); //read from keyboard

DataInputStream dins=new DataInputStream(s.getInputStream());

DataOutputStream dout=new DataOutputStream(s.getOutputStream());

String str;

while(true)

{

str=dins.readUTF();

System.out.println("Client says " +str);

if(str.equals("bye"))

break;

System.out.println("Enter msg for client");

str=din.nextLine();

dout.writeUTF(str);

if(str.equals("bye"))

break;

}

ss.close();

}catch(Exception e)

{

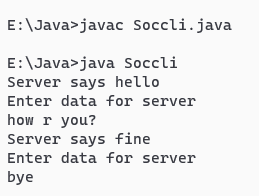
System.out.println(e);

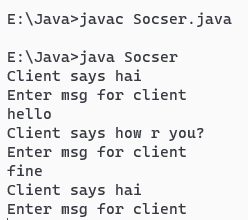
}

}

}

**Output:**

****

****

|  |
| --- |
| **Program 60 Date: 15-02-2023** |
| **Write a program to implement public chatting.** |

**Code:**

import java.net.\*;

import java.io.\*;

import java.util.\*;

public class GroupChat

{

private static final String TERMINATE = "Exit";

static String name;

static volatile boolean finished = false;

public static void main(String[] args)

{

if (args.length != 2)

System.out.println("Two arguments required: <multicast-host> <port-number>");

else

{

try

{

InetAddress group = InetAddress.getByName(args[0]);

int port = Integer.parseInt(args[1]);

Scanner sc = new Scanner(System.in);

System.out.print("Enter your name: ");

name = sc.nextLine();

MulticastSocket socket = new MulticastSocket(port);

// Since we are deploying

socket.setTimeToLive(0);

//this on localhost only (For a subnet set it as 1)

socket.joinGroup(group);

Thread t = new Thread(new

ReadThread(socket,group,port));

// Spawn a thread for reading messages

t.start();

// sent to the current group

System.out.println("Start typing messages...\n");

while(true)

{

String message;

message = sc.nextLine();

if(message.equalsIgnoreCase(GroupChat.TERMINATE))

{

finished = true;

socket.leaveGroup(group);

socket.close();

break;

}

message = name + ": " + message;

byte[] buffer = message.getBytes();

DatagramPacket datagram = new DatagramPacket(buffer,buffer.length,group,port);

socket.send(datagram);

}

}

catch(SocketException se)

{

System.out.println("Error creating socket");

se.printStackTrace();

}

catch(IOException ie)

{

System.out.println("Error reading/writing from/to socket");

ie.printStackTrace();

}

}

}

}

class ReadThread implements Runnable

{

private MulticastSocket socket;

private InetAddress group;

private int port;

private static final int MAX\_LEN = 1000;

ReadThread(MulticastSocket socket,InetAddress group,int port)

{

this.socket = socket;

this.group = group;

this.port = port;

}

@Override

public void run()

{

while(!GroupChat.finished)

{

byte[] buffer = new byte[ReadThread.MAX\_LEN];

DatagramPacket datagram = new

DatagramPacket(buffer,buffer.length,group,port);

String message;

try

{

socket.receive(datagram);

message = new

String(buffer,0,datagram.getLength(),"UTF-8");

if(!message.startsWith(GroupChat.name))

System.out.println(message);

}

catch(IOException e)

{

System.out.println("Socket closed!");

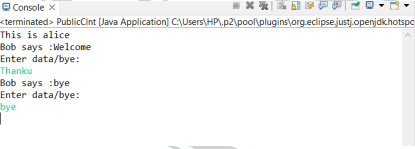
}

}

}

}

**Output:**



|  |
| --- |
| **Program 61 Date: 17-02-2023** |
| **Write a program to get the protocol, file name, host, path and port of a given URL.** |

**Code:**

import java.io.IOException;

import java.io.InputStream;

import java.net.URL;

import java.net.URLConnection;

public class Details {

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

URL u = new URL("https://github.com/openai/chatgpt-retrieval-plugin");

URLConnection uc = u.openConnection();

System.out.println("Protocol is "+u.getProtocol());

System.out.println("File Name: "+u.getFile());

System.out.println("Host is: "+u.getHost());

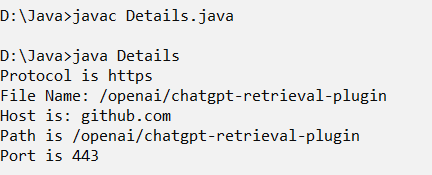
System.out.println("Path is "+u.getPath());

System.out.println("Port is "+u.getDefaultPort());

}

}

**Output:**

****

|  |
| --- |
| **Program 62 Date: 17-02-2023** |
| **Write a program to download a file from a given URL** |

**Code:**

package db\_package;

import java.net.\*;

import java.util.Scanner;

import java.io.\*;

public class FileDownURL

{

public static void main(String[] args)

{

try

{

Scanner sc = new Scanner(System.*in*);

System.*out*.println ("Enter the url to download: ");

String link = sc.nextLine();

URL u = new URL(link);

String s = u.getFile();

String ext = s.substring(s.indexOf(".") + 1);

System.*out*.println ("File type: "+ext);

InputStream is = u.openStream();

FileOutputStream os = new FileOutputStream("E:\\URL\\URLsample.pdf");

int l;

byte[] b = new byte[2048];

while ((l=is.read(b))!=-1)

{

os.write(b,0,l);

}

System.*out*.println ("File written");

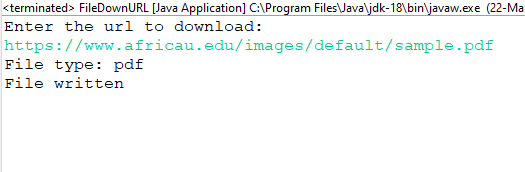
}

catch (Exception e) { System.*out*.println(e); }

}

}

**Ouput:**



|  |
| --- |
| **Program 63 Date: 17-02-2023** |
| **Implement Two- way Communication using UDP Protocol.** |

**Code:**

import java.io.IOException;

import java.net.\*;

public class UDPTwoWayCli

{

public static void main String[] args) throws IOException

{

int i = 10;

byte[] b = (String.valueOf(i)).getBytes();

DatagramSocket ds = new DatagramSocket();

InetAddress my = InetAddress.getLocalHost();

DatagramPacket dp = new DatagramPacket(b,b.length,my,1520);

ds.send(dp);

System.out.println ("Data is sent");

DatagramSocket ds1 = new DatagramSocket(1750);

byte [] b2 = new byte[2048];

DatagramPacket dp1 = new DatagramPacket(b2,b2.length);

ds1.receive(dp1);

System.out.println("Data is received again");

String str2 = new String(dp1.getData());

int num = Integer.parseInt(str2.trim());

int sq = num\*num;

System.out.println("Data2 is "+sq);//display 11^2=121

}

}

import java.io.IOException;

import java.net.\*;

public class UDPTwoWaySer

{

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

{

byte[] b1 = new byte[2048];

DatagramSocket ds = new DatagramSocket(1520);

DatagramPacket dp = new DatagramPacket(b1,b1.length);

ds.receive(dp);

String str = new String(dp.getData());

int num = Integer.parseInt(str.trim());

num++;

System.out.println ("Data 1 is " +num);

byte[] b = (String.valueOf(num)).getBytes();

InetAddress my = InetAddress.getLocalHost();

DatagramPacket dp1 = new DatagramPacket(b,b.length,my,1750);

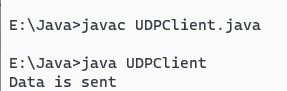
ds.send(dp1);

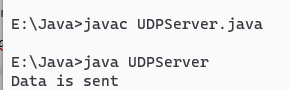
System.out.println("Data is sent again");

}

}

**Output:**





|  |
| --- |
| **Program 64 Date: 22-02-2023** |
| **Write a program to create a table Citizen( Id(Primary), Name, age, address, DOB), insert records, and display the records** |

**Code:**

package lab;

import java.sql.\*;

public class Citizen {

public static void main(String[] args) {

try {

Class.forName("oracle.jdbc.driver.OracleDriver");

java.sql.Connection con= DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl","hr"," A121619\_a");

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery("select \* from Citizen");

while(rs.next()) {

System.out.println(rs.getString(1)+" "+rs.getString(2)+" "+rs.getInt(3)+" "+rs.getString(4)+" "+rs.getString(5));

}

System.out.println("done");

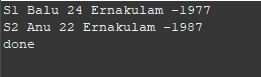
con.close();

}catch(Exception e){ System.out.println(e);}

}

}

**Output:**

****

|  |
| --- |
| **Program 65 Date: 22-02-2023** |
| **Assume that login is a table which has Uname, Upass. Check whether a record with “Uname=”Bob” and “UPass=”Alice123#”is present in the table.** |

**Code:**

package cw;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.Scanner;

public class Pro\_002 {

static Scanner sc = new Scanner(System.in);

public static void main(String[] args) {

// TODO Auto-generated method stub

try (Connection conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl", "hr", " A121619\_a");

Statement st = conn.createStatement();) {

Class.forName("oracle.jdbc.driver.OracleDriver");

String uname;

String upass;

System.out.println("Enter the Username: ");

uname = sc.nextLine();

System.out.println("Enter the Password: ");

upass = sc.nextLine();

String s = String.format("SELECT \* FROM login where uname='%s' and upass='%s'",uname,upass);

int res = st.executeUpdate(s);

if(res > 0) {

System.out.println("Login Successful");

}else {

System.out.println("Invalid Credentials");

}

} catch (SQLException se) {

System.out.println(se.getMessage());

} catch (Exception e) {

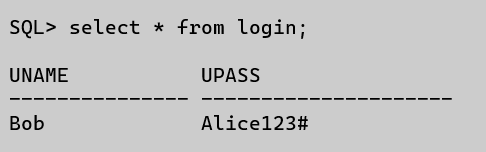
System.out.println(e.getMessage());

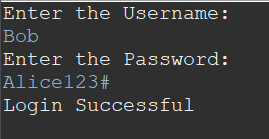
}

}

}

**Output:**

****

****

|  |
| --- |
| **Program 66 Date: 24-02-2023** |
| **Construct the following tables:**  **Department (dno(Primary), dname, dloc)**  **Emp ( eno(Primary), ename, esal ,dno(Foreign))** |

**Code:**

package cw;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.sql.Statement;

public class Pro\_003 {

public static void main(String[] args) {

// TODO Auto-generated method stub

try (Connection conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl", "hr", " A121619\_a");

Statement st = conn.createStatement();) {

Class.forName("oracle.jdbc.driver.OracleDriver");

System.out.println("Quering...");

boolean res = st.execute("CREATE TABLE Department(dno number primary key,dname varchar(10),dloc varchar2(25))");

if (res) {

System.out.println("Department Table Created Successfully!");

}

res = st.execute("CREATE TABLE EMP(eno number primary key,ename varchar(10),esal float,dno number references Department(dno))");

if (res) {

System.out.println("Employee Table Created Successfully!");

}

String deps[] = { "CS", "BCom", "Lib", "Stat", "MSW" };

for (int i = 0; i < deps.length; i++) {

String q = String.format("INSERT INTO Department VALUES(%d,'%s','%s')", i+1,deps[i],"RCSS");

System.out.println(q);

st.executeUpdate(q);

}

System.out.println("Departments Added Successfully");

String emps[] = {"A","B","C","D","E"};

for (int i = 0; i < emps.length; i++) {

String q = String.format("INSERT INTO EMP VALUES(%d,'%s',%f,%d)", i+1,emps[i],(float)100\*(i+1),i+1);

System.out.println(q);

st.executeUpdate(q);

}

System.out.println("Employees Added Successfully");

System.out.println("Done");

} catch (SQLException se) {

System.out.println(se.getMessage());

} catch (Exception e) {

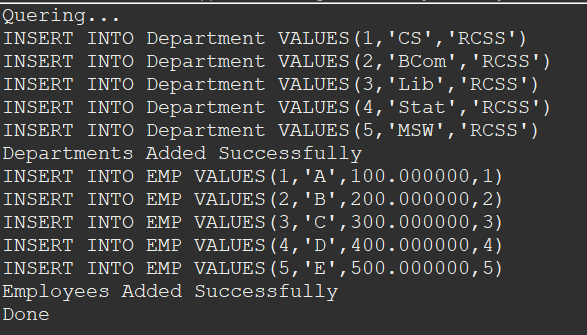
System.out.println(e.getMessage());

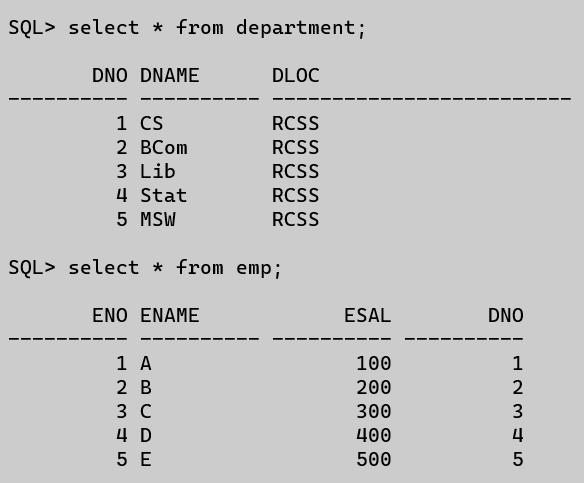
}

}

}

**Output:**

****

****

|  |
| --- |
| **Program 67 Date: 24-02-2023** |
| **Write a program for displaying information in the following order:**  **eno ename esal dname dloc**  **101 Rani 10,000 MCA Kochi**  **102 Vani 20,000 MSW Delhi** |

**Code:**

package cw;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

public class Pro\_004 {

public static void main(String[] args) {

// TODO Auto-generated method stub

try (Connection conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl", "hr", " A121619\_a");

Statement st = conn.createStatement();) {

Class.forName("oracle.jdbc.driver.OracleDriver");

ResultSet rs = st.executeQuery(

"select e.eno,e.ename,e.esal,d.dname,d.dloc from emp e inner join department d on d.dno=e.dno");

System.out.println("eno ename esal dname dloc");

while (rs.next()) {

System.out.println(rs.getInt(1) + " " + rs.getString(2) + " " + rs.getDouble(3) + " " + rs.getString(4)

+ " " + rs.getString(5));

}

System.out.println("Done");

} catch (SQLException se) {

System.out.println(se.getMessage());

} catch (Exception e) {

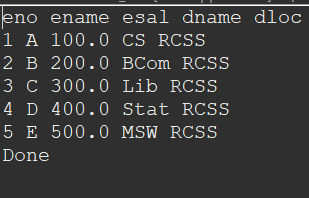
System.out.println(e.getMessage());

}

}

}

**Output:**

****

|  |
| --- |
| **Program 68 Date: 01-03-2023** |
| **Write a JDBC program with Parameterized queries to update a given record (Rani’s salary to 15,000) in the Emp table.** |

**Code:**

package cw;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.Scanner;

public class Pro\_005 {

static Scanner sc = new Scanner(System.in);

public static void main(String[] args) {

// TODO Auto-generated method stub

try (Connection conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "hr", " A121619\_a");

Statement st = conn.createStatement();) {

Class.forName("oracle.jdbc.driver.OracleDriver");

double salary = 15000;

String name = "Rani";

int rs = st.executeUpdate("Update emp set esal="+salary+" where ename='"+name+"'");

if(rs > 0) {

System.out.println("Done");

}

} catch (SQLException se) {

System.out.println(se.getMessage());

} catch (Exception e) {

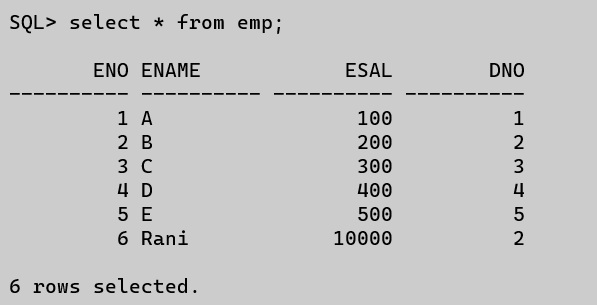
System.out.println(e.getMessage());

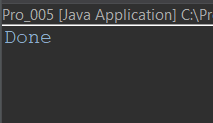
}

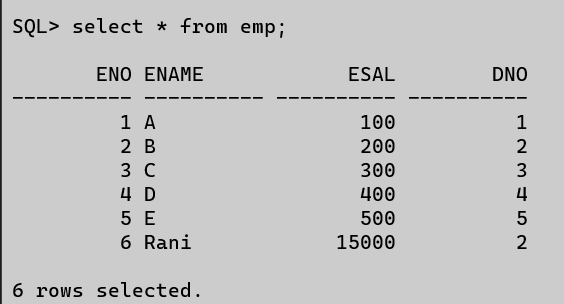
}

}

**Output:**

****

****

****

|  |
| --- |
| **Program 69 Date: 01-03-2023** |
| **Write a JDBC program with Parameterized queries to list the records of the Emp table which has records whose names start with the alphabet “R”.** |

**Code:**

package cw;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.Scanner;

public class Pro\_006 {

static Scanner sc = new Scanner(System.in);

public static void main(String[] args) {

// TODO Auto-generated method stub

try (Connection conn = DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe", "hr", " A121619\_a");

Statement st = conn.createStatement();) {

Class.forName("oracle.jdbc.driver.OracleDriver");

ResultSet rs = st.executeQuery("SELECT \* FROM EMP WHERE ENAME like'R%'");

while(rs.next()) {

System.out.println(rs.getString(2));

}

} catch (SQLException se) {

System.out.println(se.getMessage());

} catch (Exception e) {

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

}

}

}

**Output:**

****

|  |
| --- |
| **Program 70 Date: 01-03-2023** |
| **Write a JDBC program with PreparedStatement to delete the records of the Emp table which has records whose salary is less than 10,000.** |

**Code:**

package db\_package;

import java.sql.\*;

import java.util.Scanner;

public class EmpPara3

{

public static void main(String[] args)

{

try

{

Scanner sc = new Scanner(System.*in*);

Class.*forName*("oracle.jdbc.driver.OracleDriver");

Connection con = DriverManager.*getConnection* ("jdbc:oracle:thin:@localhost:1521:orcl","hr","hr");

PreparedStatement stmt = con.prepareStatement("delete from Employee1 where esal<=15000");

int l = stmt.executeUpdate();

System.*out*.println(l+" row(s) updated");

System.*out*.println("Records Deleted");

con.close();

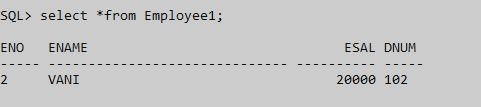
}

catch(Exception e) { System.*out*.println(e); }

}

}

**Output:**



|  |
| --- |
| **Program 71 Date: 03-03-2023** |
| **Implement a JDBC program which uses a Stored Procedure to insert records into the Department table.** |

**SQL:**

create or replace procedure "EMPINSERT3"

(dno IN varchar2,dname IN varchar2)

is

begin

insert into Dept values(dno,dname);

end;

/

**Code:**

package dbprg;

import java.sql.\*;

import java.io.\*;

import java.util.\*;

public class InsrtDept

{

public static void main(String[] args)

{

try

{

Class.*forName*("oracle.jdbc.driver.OracleDriver");

Connection con = DriverManager.*getConnection* ("jdbc:oracle:thin:@localHost:1521:orcl","hr"," A121619\_a");

CallableStatement cs=con.prepareCall("{call EMPINSERT3(?,?)}");

cs.setString(1,"103");

cs.setString(2,"Ammu");

cs.executeUpdate();

System.*out*.println("Value inserted");

con.close();

}

catch (ClassNotFoundException a)

{

System.*out*.println( "ERROR"+a);

}

catch(SQLException e)

{

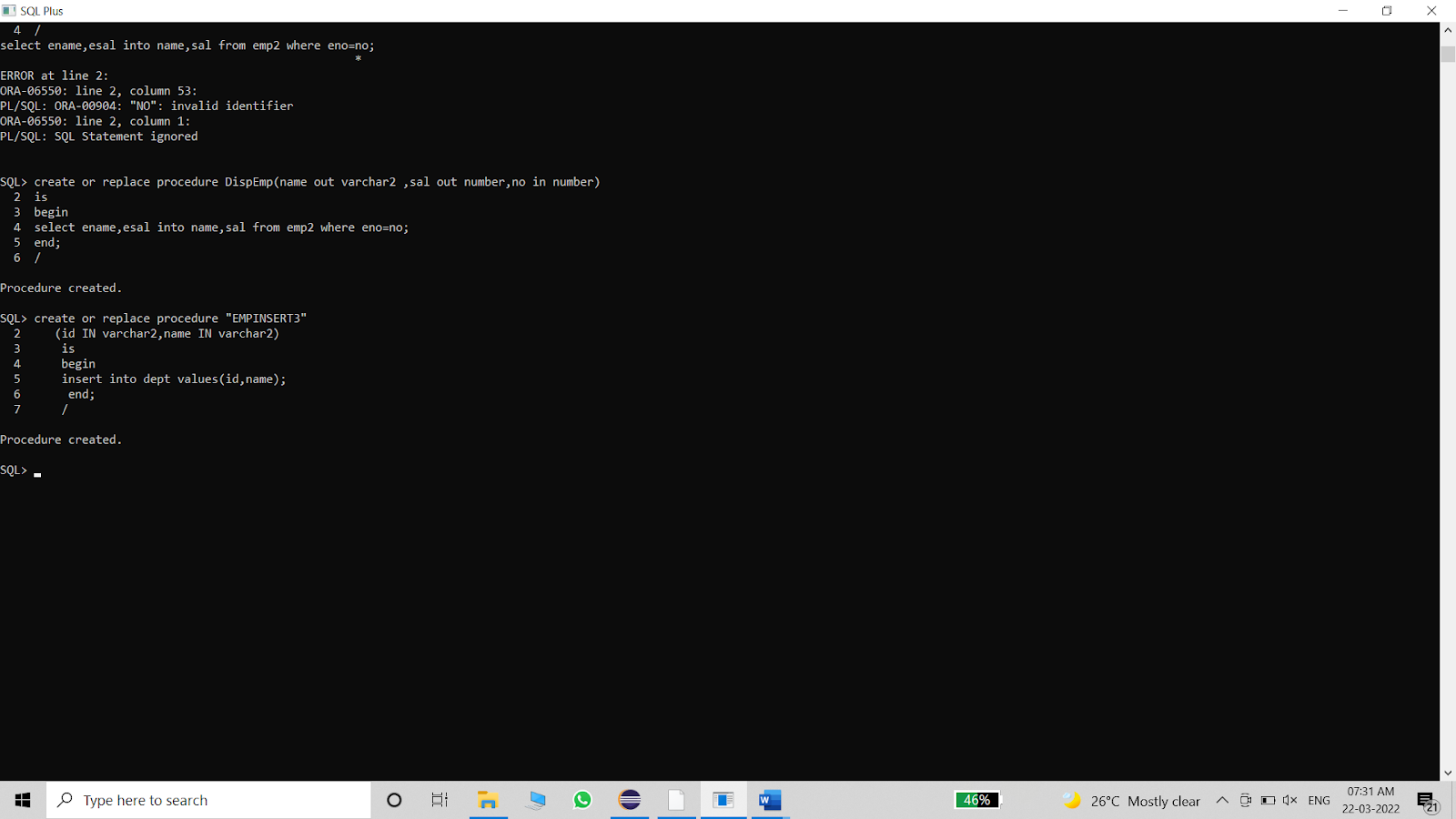
System.*out*.println( "ERROR"+e);

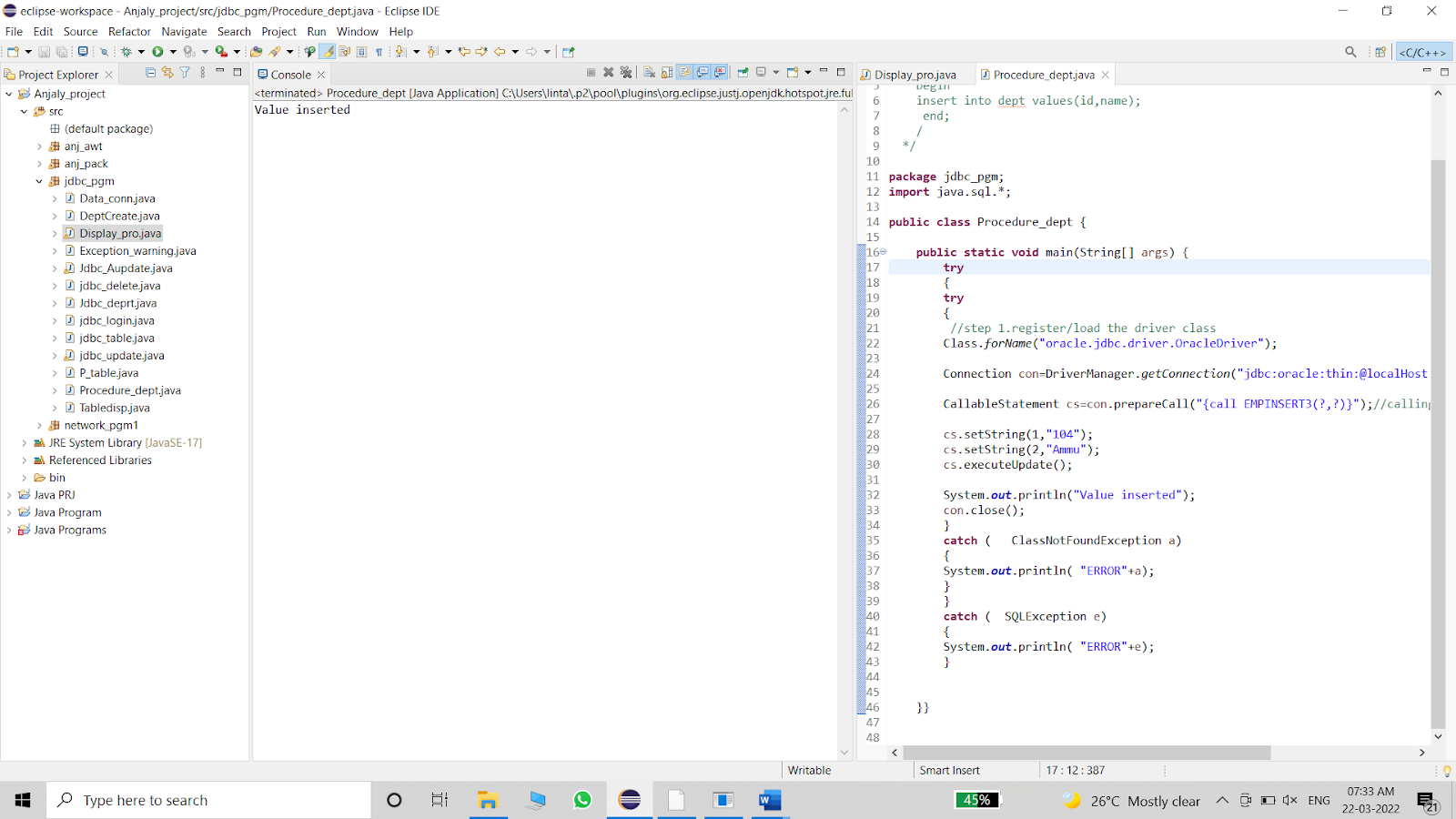
}

}

}

**Output:**





|  |
| --- |
| **Program 72 Date: 03-03-2023** |
| **Use Callable statement to implement a Stored Procedure to display the Ename and Salary of all employees.** |

**Code:**

package dbprg;

import java.sql.\*;

import java.util.\*;

public class DispEmp

{

public static void main(String[] args)

{

try

{

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con=

DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl","hr",” A121619\_a");

CallableStatement cs = con.prepareCall("{call DispEmp(?,?,?)}");

Statement stmt = con.createStatement();

ResultSet rs =stmt.executeQuery("select \* from Employee2");

System.out.println("Employee Name\tSalary");

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

while(rs.next())

{

Scanner sc = new Scanner(System.in);

int id = rs.getInt(1);

cs.registerOutParameter(1, java.sql.Types.VARCHAR);

cs.registerOutParameter(2, java.sql.Types.NUMERIC);

cs.setInt(3, id);

cs.executeUpdate();

String name = cs.getString(1);

String salary = cs.getString(2);

System.out.println(name+"\t\t"+salary);

}

}

catch(Exception e)

{

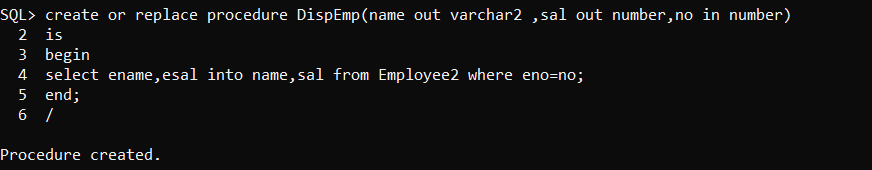
System.out.println("error "+e);

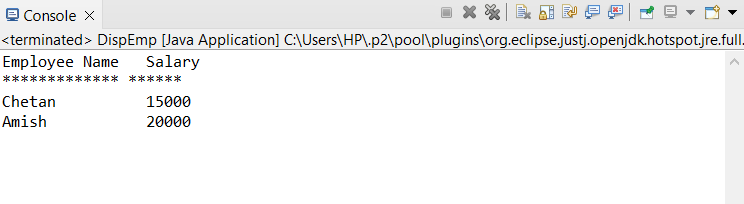
}

}

}

**Output:**

****

****

|  |
| --- |
| **Program 73 Date: 03-03-2023** |
| **Write a JDBC program to implement Transaction Management in the Department table.** |

**Code:**

package dbprg;

import java.sql.\*;

import java.io.\*;

public class TransDept {

public static void main(String[] args) {

try

{

Class.*forName*("oracle.jdbc.driver.OracleDriver");

Connection con=DriverManager.*getConnection*("jdbc:oracle:thin:@localhost:1521:orcl","hr", " A121619\_a");

PreparedStatement pst=con.prepareStatement("insert into department values(?,?,?)");

con.setAutoCommit(false);

BufferedReader br=new BufferedReader(new

InputStreamReader(System.*in*));

do

{

System.*out*.println("Enter No :");

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

System.*out*.println("Enter name :");

String name=br.readLine();

System.*out*.println("Enter location :");

String loc=br.readLine();

pst.setInt(1,id);

pst.setString(2,name);

pst.setString(3,loc);

pst.executeUpdate();

System.*out*.println("Commit/Rollback?(c/r)");

String ans=br.readLine();

if(ans.startsWith("c"))

con.commit();

else

con.rollback();

System.*out*.println("Yes/No");

String s=br.readLine();

if(s.startsWith("n"))

break;

}while(true);

System.*out*.println("Records updated!");

con.close();

}

catch(Exception e)

{

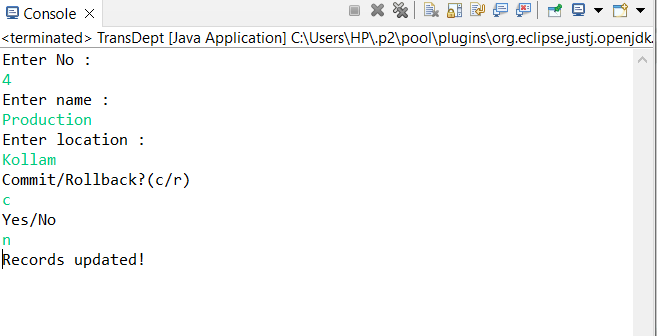
System.*out*.println(e);

}

}

}

**Output:**

****

|  |
| --- |
| **Program 74 Date: 03-03-2023** |
| **Write a JDBC program to depict the usage of SQLException Class and SQLWarning Class** |

**Code:**

package dbprg;

import java.sql.\*;

public class Excep4

{

public static void main(String[] args)

{

try

{

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con = DriverManager.getConnection ("jdbc:oracle:thin:@localhost:1521:orcl","hr”,” A121619\_a");

Statement stmt=con.createStatement();

stmt.executeUpdate("select \* from Employee2 where ename=Anjali ");

}

catch(SQLException e)

{

System.out.println("SQL message :"+e.getMessage());

System.out.println("SQL state :" +e.getSQLState());

System.out.println("SQL error code :"+e.getErrorCode());

System.out.println("SQL cause :"+e.getCause());

e.printStackTrace();

}

catch(Exception e)

{

System.out.println("error "+e);

}

}

}

**Output:**

****