**Practical no. 1**

**Aim:** Accept two integer number using command line argument and display greatest number between two numbers.

**Date:** **Roll no.:** **sign:**

**Program:**

class G

{

public static void main(String args[])

{

int n1,n2;

n1 = Integer.parseInt(args[0]);

n2 = Integer.parseInt(args[1]);

if(n1 > n2)

{

System.out.println(n1 +" is greater than "+ n2);

}

else

{

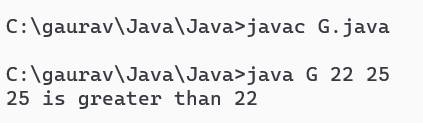
System.out.println(n2 +" is greater than "+ n1);

}

}

}

**Output:**



**Practical no. 2**

**Aim:** Accept any character from user and Display its ASCII code.

**Date:** **Roll no.:** **sign:**

**Program:**

import java.io.\*;

class Asc

{

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

{

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

char ch;

System.out.println("Enter any character");

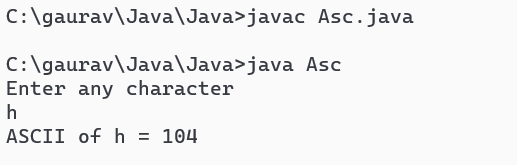
ch = (br.readLine()).charAt(0);

System.out.println("ASCII of "+ ch +" = "+ (int)(ch));

}

}

**Output:**



**Practical no. 3**

**Aim:** Accept number of rows from user and Display pyramid pattern.

**Date:** **Roll no.:** **sign:**

**Program:**

import java.io.\*;

class Pyrl

{

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

{

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

int i,j,k,row,col,pos,star;

System.out.println("Enter no. of rows");

row=Integer.parseInt(br.readLine());

col=(2\*row)-1;

pos = row;

star=1;

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

{

for(j=1;j<=col;j++)

{

if(j==pos)

{

for(k=1;k<=star;k++)

{

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

j++;

}

j--;

}

else

{

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

}

}

pos--;

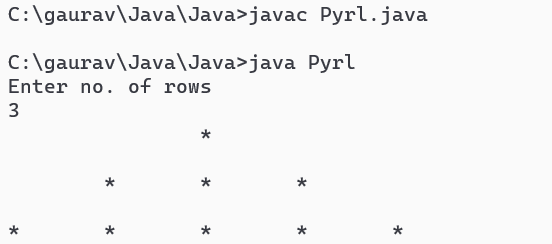
star+=2;

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

}

}

}

**Output:**

**Practical no. 4**

**Aim:** Inheritance

* Single Inheritance
* Multilevel Inheritance
* Hierarchical Inheritance

**Date:** **Roll no.:** **sign:**

**Single Inheritance:**

**Program:**

class A

{

private int x1;

protected int y1;

public int z1;

public A()

{

System.out.println("constructor of A get fired\n");

x1=100;

y1=200;

z1=300;

}

public void disA()

{

System.out.println("In class A x1= "+x1);

System.out.println("In class A y1= "+y1);

System.out.println("In class A z1= "+z1);

}

}

class B extends A

{

public int z2;

public B()

{

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

z2=y1+z1;

}

public void disB()

{

System.out.println("Inside class B z2= "+z2);

}

public static void main(String args[])

{

B b1=new B();

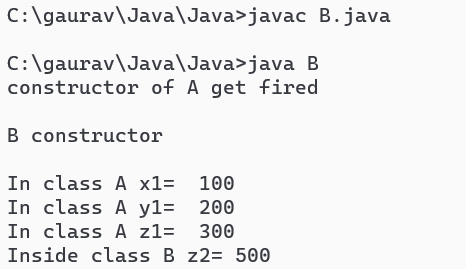
b1.disA();

b1.disB();

}

}

**Output:**



**Multilevel Inheritance:**

**Program:**

class A

{

private int x1;

protected int y1;

public int z1;

public A()

{

System.out.println("constructor of A get fired\n");

x1=100;

y1=200;

z1=300;

}

public void disA()

{

System.out.println("In class A x1= "+x1);

System.out.println("In class A y1= "+y1);

System.out.println("In class A z1= "+z1);

}

}

class B extends A

{

public int z2;

protected int y2;

private int x2;

public B()

{

System.out.println("B constructor is fired\n");

z2=y1+z1;

y2=555;

x2=1000;

}

public void disB()

{

System.out.println("Inside class B z2= "+z2+"\nx2="+x2+"\ny2="+y2);

}

}

class D extends B

{

public int z3;

public D()

{

z3=1234;

}

public void disD()

{

System.out.println("Inside class D z3="+z3);

}

public static void main(String args[])

{

D c1=new D();

c1.disA();

c1.disB();

c1.disD();

System.out.println("Inside main z1= "+c1.z1);

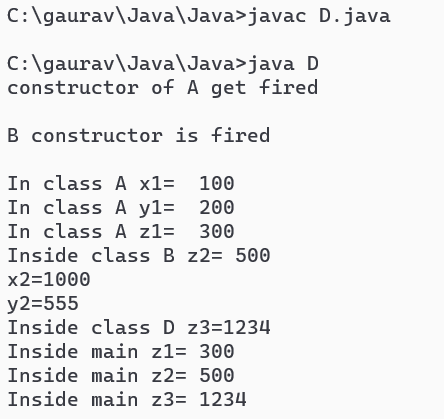
System.out.println("Inside main z2= "+c1.z2);

System.out.println("Inside main z3= "+c1.z3);

}

}

**Output:**

****

**Hierarchical Inheritance:**

**Program:**

class A

{

private int x1;

protected int y1;

public int z1;

public A()

{

x1=1;y1=2;z1=3;

}

public void seeA()

{

System.out.println("Inside A x1="+x1);

System.out.println("Inside A y1="+y1);

System.out.println("Inside A z1="+z1);

}

}

class B extends A

{

public int z2;

public B()

{

z2=100;

}

public void seeB()

{

System.out.println("Inside B z2="+z2);

System.out.println("Inside B y1="+y1);

System.out.println("Inside B z1="+z1);

}

}

class CC extends A

{

public int z3;

public CC()

{

z3=111;

}

public void seeC()

{

System.out.println("Inside C z3="+z3);

System.out.println("Inside C y1="+y1);

System.out.println("Inside C z1="+z1);

}

public static void main(String args[])

{

CC c = new CC();

c.seeA();

c.seeC();

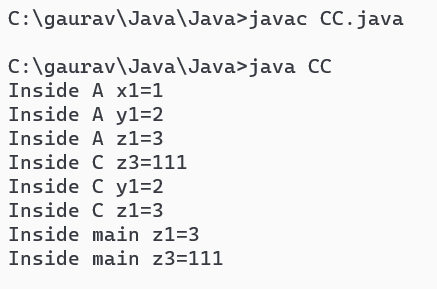
System.out.println("Inside main z1="+c.z1);

System.out.println("Inside main z3="+c.z3);

}

}

**Output:**

****

**Practical no. 5**

**Aim:** Polymorphism

* Constructor overloading
* Method overloading
* Method overriding

**Date:** **Roll no.:** **sign:**

**Constructor overloading:**

**Program:**

class Constructor

{

private int x1;

protected int y1;

public int z1;

public Constructor()

{

System.out.println("constructor without parameter");

x1=11;y1=22;z1=33;

}

public Constructor(int a)

{

System.out.println("constructor with one parameter");

x1=880;y1=22;z1=99;

}

public Constructor(int b,int c)

{

System.out.println("constructor with two parameter");

x1=b;y1=c;z1=b+c;

}

public Constructor(int a,int b,int c)

{

System.out.println("constructor with three parameter");

x1=a;y1=b;z1=c;

}

public void showConstructor()

{

System.out.println("Inside class Constructor x1= "+x1);

System.out.println("Inside class Constructor y1= "+y1);

System.out.println("Inside class Constructor z1= "+z1);

}

public static void main(String args[])

{

Constructor c1=new Constructor();

c1.showConstructor();

Constructor c2=new Constructor(100);

c2.showConstructor();

Constructor c3=new Constructor(100,220);

c3.showConstructor();

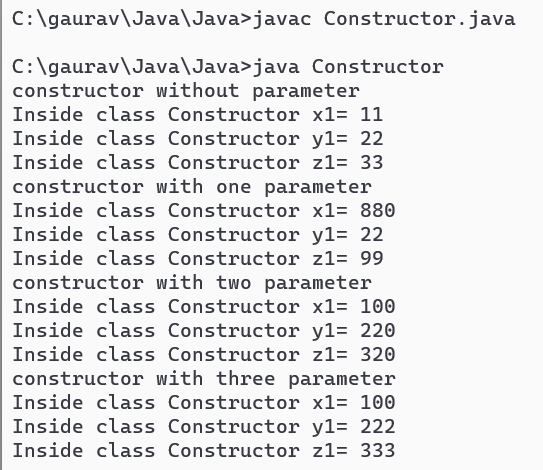
Constructor c4=new Constructor(100,222,333);

c4.showConstructor();

}

}

**Output:**

****

**Method overloading:**

**Program:**

class EE

{

private int x1;

protected int y1;

public int z1;

public void showE()

{

x1=11;y1=22;z1=33;

System.out.println("x1= "+x1+"\ny1= "+y1+"\nz1= "+z1);

}

public void showE(int a)

{

x1=a;y1=22;z1=33;

System.out.println("x1= "+x1+"\ny1= "+y1+"\nz1= "+z1);

}

public void showE(int a,int b)

{

x1=a;y1=b;z1=a+b;

System.out.println("x1= "+x1+"\ny1= "+y1+"\nz1= "+z1);

}

public void showE(int a,int b,int c)

{

x1=a;y1=b;z1=c;

System.out.println("x1= "+x1+"\ny1= "+y1+"\nz1= "+z1);

}

public static void main(String args[])

{

EE e=new EE();

e.showE();

e.showE(100);

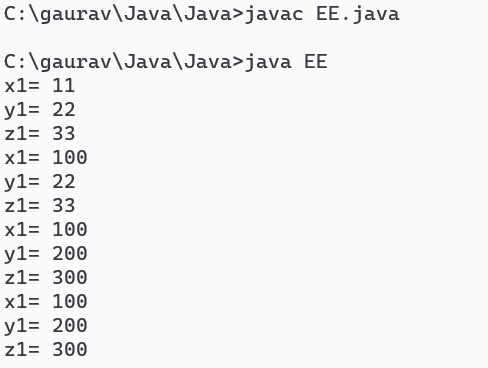
e.showE(100,200);

e.showE(100,200,300);

}

}

**Output:**

****

**Method overriding:**

**Program**

class EEE

{

private int x1;

protected int y1;

public int z1;

public void showE()

{

System.out.println("\n Inside class EEE\n");

x1=111;

y1=222;

z1=333;

System.out.println(" x1= "+x1+" y1= "+y1+" z1= "+z1);

}

}

class DDD extends EEE

{

public void showE()

{

System.out.print("\n Inside class DDD\n");

y1=343;

z1=939;

System.out.println(" y1= "+y1+" z1= "+z1);

}

public static void main(String args[])

{

DDD d=new DDD();

d.showE();

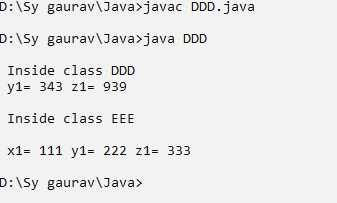
EEE e=new EEE();

e.showE();

}

}

**Output:**

****

**Practical no. 6**

**Aim:** Graphical User Interface program using AWT and Swing package.

**Date:** **Roll no.:** **sign:**

**AWT Package:**

**Program:**

//Graphical User Interface Checkbox form

import java.awt.\*;

import java.awt.event.\*;

class CheckDemo extends Frame implements ActionListener

{

Label l1;

Checkbox c1,c2,c3;

Button b1,b2;

Panel p1,p2;

public CheckDemo()

{

setLayout(new GridLayout(3,1,20,20));

setSize(300,300);

p1=new Panel();

p2=new Panel();

l1=new Label("order: ");

b1=new Button("Order");

b2=new Button("Cancel");

c1=new Checkbox("tea");

c2=new Checkbox("coffe");

c3=new Checkbox("soft drink");

addComp();

b1.addActionListener(this);

b2.addActionListener(this);

setVisible(true);

}

public void actionPerformed(ActionEvent ae)

{

String x="order:";

if(ae.getSource()==b2)

{

c1.setState(false);

c2.setState(false);

c3.setState(false);

l1.setText("Order: ");

}

if(ae.getSource()==b1)

{

if(c1.getState()==true)

{

x+="tea, ";

}

if(c2.getState()==true)

{

x+="coffe, ";

}

if(c3.getState()==true)

{

x+="soft drink ";

}

l1.setText(x);

}

}

private void addComp()

{

this.p1.setLayout(new GridLayout(3,1,20,20));

this.p1.add(c1);

this.p1.add(c2);

this.p1.add(c3);

this.p2.setLayout(new GridLayout(2,1,20,20));

p2.add(b1);

p2.add(b2);

this.add(p1);

this.add(p2);

this.add(l1);

}

public static void main(String args[])

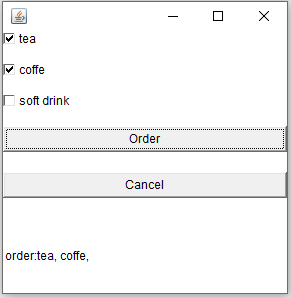
{

new CheckDemo();

}

}

**Output:**

****

**Swing Package:**

**Program:**

//Graphical User Interface RadioButton form

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

class JradioDemo1 extends JFrame implements ItemListener

{

JLabel l1;

JRadioButton c1,c2,c3;

JButton b1,b2;

JPanel p1,p2;

public JradioDemo1()

{

setLayout(new GridLayout(3,1,20,20));

ButtonGroup bg=new ButtonGroup();

setSize(300,300);

p1=new JPanel();

p2=new JPanel();

l1=new JLabel("order: ");

b1=new JButton("Order");

b2=new JButton("Cancel");

c1=new JRadioButton("tea",false);

c2=new JRadioButton("coffe",false);

c3=new JRadioButton("soft drink",false);

bg.add(c1);

bg.add(c2);

bg.add(c3);

c1.addItemListener(this);

c2.addItemListener(this);

c3.addItemListener(this);

addComp();

//b1.addActionListener(this);

//b2.addActionListener(this);

setVisible(true);

}

public void actionPerformed(ActionEvent ae)

{

String x="order:";

/\*

if(ae.getSource()==b2)

{

c1.setSelected(false);

c2.setSelected(false);

c3.setSelected(false);

l1.setText("Order: ");

}

\*/

//if(ae.getSource()==b1)

//{

if(c1.isSelected()==true)

{

x+="tea ";

}

else if(c2.isSelected()==true)

{

x+="coffe ";

}

else if(c3.isSelected()==true)

{

x+="soft drink ";

}

l1.setText(x);

//}

}

private void addComp()

{

this.p1.setLayout(new GridLayout(3,1,20,20));

this.p1.add(c1);

this.p1.add(c2);

this.p1.add(c3);

this.p2.setLayout(new GridLayout(2,1,20,20));

p2.add(b1);

p2.add(b2);

this.add(p1);

this.add(p2);

this.add(l1);

}

public static void main(String args[])

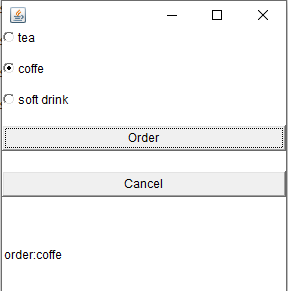
{

new JradioDemo();

}

}

**Output:**

****

**Practical no. 7**

**Aim:** Write a program to fetch all records from table and display them.

**Date:** **Roll no.:** **sign:**

**Program:**

import java.sql.\*;

class DbDemo {

public static void main(String args[]) {

Connection con=null;

try {

Class.forName("com.mysql.cj.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/sycs", "root", "");

Statement stmt = con.createStatement();

String sql = "Select \* from employee";

ResultSet res = stmt.executeQuery(sql);

while (res.next()) {

System.out.println(res.getString(1) + "\t" + res.getString(2) + "\t" + res.getString(3));

}

} catch (Exception se) {

System.out.print("exception"+se.toString());

}

}

}

**Output:**



**Practical no. 8**

**Aim:** Prepared Statement.

**Date:** **Roll no.:** **sign:**

**Program:**

import java.sql.\*;

class DbDemoPs {

public static void main(String args[]) {

Connection con=null;

String ename=args[1];

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

int esalary=Integer.parseInt(args[2]);

String sql="insert into employee values(?,?,?)" ;

try {

Class.forName("com.mysql.cj.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/sycs", "root", "");

PreparedStatement ps = con.prepareStatement(sql);

ps.setInt(1,id);

ps.setString(2,ename);

ps.setInt(3,esalary);

int x=ps.executeUpdate();

if(x>0)

{

System.out.println("record inserted successfully");

}

else

{

System.out.println("error");

}

con.close();

} catch (Exception se) {

System.out.print("exception"+se.toString());

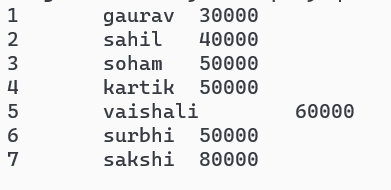
}

}

}

**Output:**

****



**Practical no. 9**

**Aim:** Batch Processing.

**Date:** **Roll no.:** **sign:**

**Program:**

import java.sql.\*;

import java.io.\*;

class DbDemoPs2 {

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

Connection con=null;

String ename;

int id,esalary;

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

String sql="insert into employee values(?,?,?)" ;

try {

Class.forName("com.mysql.cj.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/sycs", "root", "");

PreparedStatement ps = con.prepareStatement(sql);;

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

{

System.out.println("Enter the values of id, ename, esalary,dNo");

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

ename=br.readLine();

esalary=Integer.parseInt(br.readLine());

ps.setInt(1,id);

ps.setString(2,ename);

ps.setInt(3,esalary);

ps.addBatch();

}

ps.executeBatch();

con.close();

} catch (Exception se) {

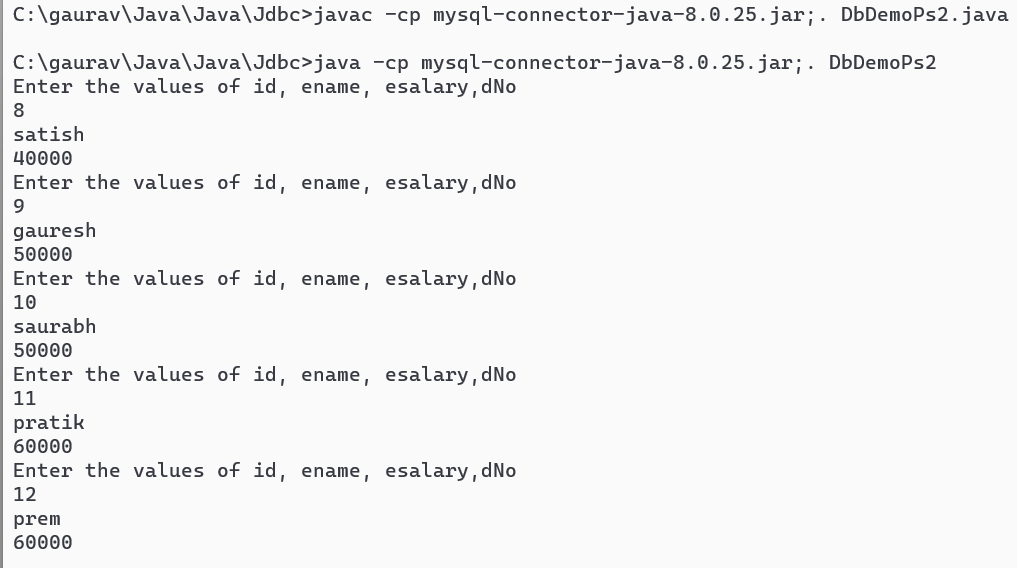
System.out.print("exception"+se.toString());

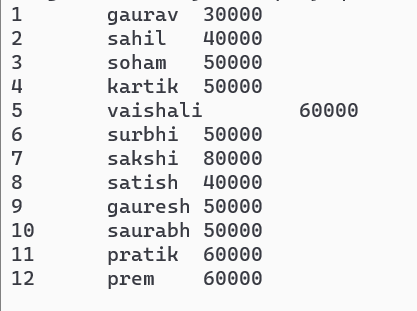
}

}

}

**Output:**

****



**Practical no. 10**

**Aim:** Write a JDBC program to Insert, Update and Delete in Database.

**Date:** **Roll no.:** **sign:**

**Insert records:**

**Program:**

import java.sql.\*;

class DbDemo3 {

public static void main(String args[]) {

Connection con=null;

String ename=args[1];

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

int esalary=Integer.parseInt(args[2]);

String sql="insert into employee values("+id+",'"+ename+"',"+esalary+")" ;

try {

Class.forName("com.mysql.cj.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/sycs", "root", "");

Statement stmt = con.createStatement( );

int x=stmt.executeUpdate(sql);

if(x>0)

{

System.out.println("record inserted successfully");

}

else

{

System.out.println("error");

}

con.close();

} catch (Exception se) {

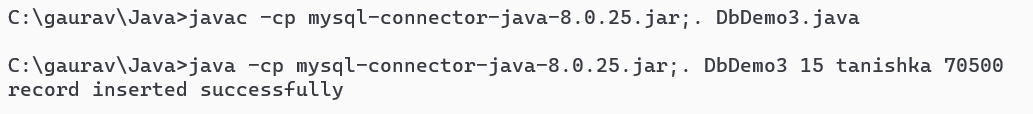
System.out.print("exception"+se.toString());

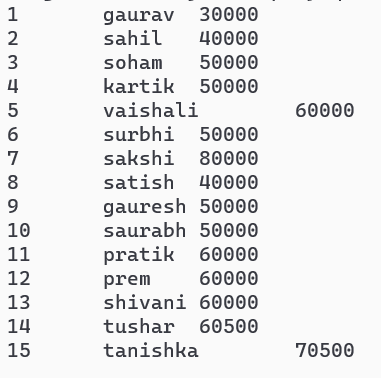
}

}

}

**Output:**

****



**Update records:**

**Program:**

import java.sql.\*;

class DbDemo3 {

public static void main(String args[]) {

Connection con=null;

String ename=args[1];

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

String sql="update employee set ename='"+ename+"'where id="+id;

try {

Class.forName("com.mysql.cj.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/sycs", "root", "");

Statement stmt = con.createStatement( );

int x=stmt.executeUpdate(sql);

if(x>0)

{

System.out.println("record update successfully");

}

else

{

System.out.println("error");

}

con.close();

} catch (Exception se) {

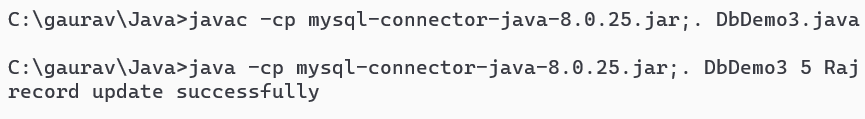
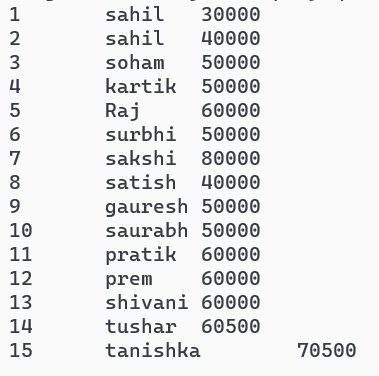
System.out.print("exception"+se.toString());

}

}

}

**Output:**



**Delete records:**

**Program:**

import java.sql.\*;

class DbDemo4 {

public static void main(String args[]) {

Connection con=null;

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

String sql="delete from employee where id="+id ;

try {

Class.forName("com.mysql.cj.jdbc.Driver");

con = DriverManager.getConnection("jdbc:mysql://localhost:3306/sycs", "root", "");

Statement stmt = con.createStatement( );

int x=stmt.executeUpdate(sql);

if(x>0)

{

System.out.println("record Delete successfully");

}

else

{

System.out.println("error");

}

con.close();

} catch (Exception se) {

System.out.print("exception"+se.toString());

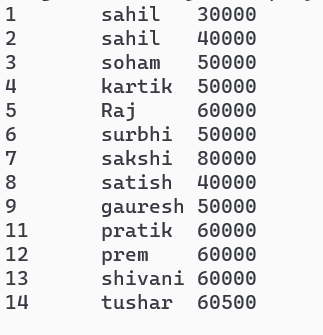
}

}

}

**Output:**

****

****