import java.io.\*;

import java.sql.\*;

class Bayes1 {

public static void main(String[] args)

{

Connection con;

Statement s;

ResultSet r1;

try

{

DataInputStream in=new DataInputStream(System.in);

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

con=DriverManager.getConnection("jdbc:odbc:dsn1");

System.out.println("success");

***//USE TO RETRIVE THE LIST OF TABLES IN DATABASE BEING SELECTED***

DatabaseMetaData meta = con.getMetaData();

ResultSet rs = meta.getTables(con.getCatalog(),null,"i%",null);

String tableNames = "";

while (rs.next()) {

tableNames = rs.getString(3);

System.out.println(tableNames);

}

System.out.println("select the name of the table given in the list ");

String tablename=in.readLine();

s=con.createStatement();

**//GIVES THE COLUMNNAME FROM THE TABLE BEING SELECTED**

r1=s.executeQuery("select \* from "+tablename);

ResultSetMetaData md=r1.getMetaData();

int m=md.getColumnCount();

String w[]=new String[m];

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

{

w[i-1]=md.getColumnName(i);

}

System.out.println(m);

int count=0;

***//GIVES THE DISTINCT VALUE OF EACH COLUMN***

for(int i=1;i<m-1;i++)

{

r1=s.executeQuery("select distinct ("+w[i]+") from "+tablename);

System.out.print(w[i]+" is: (");

while(r1.next())

{

System.out.print(r1.getString(1)+",");

}

System.out.println(")");

}

**//FIND DIFFERENT CLASSES IN WHCIH THE DATASET BEING CLASIFIED**

r1=s.executeQuery("select distinct ("+w[m-1]+") from "+tablename);

String S="";

while(r1.next())

{

S=S+r1.getString(w[m-1])+" ";

}

String classes[]=S.split(" ");

double countfull[]=new double[m-2];

double count2=0,count3=0;

double counts[][]=new double[classes.length][m+2];

**//GIVES THE TOTAL NO OF ROWS IN THE DATASET**

r1=s.executeQuery("select \* from "+tablename);

while(r1.next())

count2++;

**//FINDS THE TOTAL NUMBER OF ROWS PRESENT IN EACH OF THE CLASSES**

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

{

r1=s.executeQuery("select \* from "+tablename+" where "+w[m-1]+"='"+classes[i]+"'");

while(r1.next())

{

countfull[i]++;

}

counts[i][0]=countfull[i];

System.out.print("number of rows in "+classes[i]+"=");

System.out.println(counts[i][0]);

counts[i][m-1]=counts[i][0]/count2;

}

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

{

countfull[i]=0;

}

System.out.println("enter the conditon u want FOR EXAMPLE COLUMNAME='COLUMN VALUE'");

String data=in.readLine();

String data1[]=data.split(" ");

System.out.println();

System.out.println();

**//DEPENDING ON THE CONDITION BY USER IT CLASSIFIES AND FINDS THE UMBER OF ROWS IN EACH CLASSES//**

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

{

for(int j=0;j<data1.length;j++)

{

r1=s.executeQuery("select \* from "+tablename+" where "+data1[j]+" and "+w[m-1]+"='"+classes[i]+"'");

while(r1.next())

{

count3++;

}

counts[i][j+1]=count3;

System.out.println("p("+data1[j]+"/"+classes[i]+")=="+counts[i][j+1]);

System.out.println();

count3=0;

}

}

double high=1.0,low=1.0,moderate=1.0;

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

{

counts[i][m]=1.0;

}

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

{

for(int j=0;j<m-2;j++)

{

counts[i][m]=counts[i][m]\*counts[i][j+1]/counts[i][0];

}

}

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

System.out.println();

**//CALULATES THE PROBABILITY PRODUCT FOR ECH CLASSES BY USING FORMULA PRODUCT(P(XI|CJ)\*p(Cj|full))//**

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

{

counts[i][m+1]=counts[i][m]\*counts[i][m-1];

System.out.println("p(X/"+classes[i]+")\*p(max"+classes[i]+")=="+counts[i][m+1]);

}

**//FINDS OUT WHICH OF THE CLASS THE DATA WOULD BE BEST CLASSIFIED**

String best=""; int f=0;

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

{

for(int j=1;j<classes.length;j++)

{

if(counts[i][m+1]>counts[j][m+1])

{

best=classes[i];

f=1;

break;

}

}

}

if(f==1)

System.out.println(best+" is best");

}

catch(Exception e)

{

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

}

}

}

/\*output

\*the tables in the database are

iris1

select the name of the table given im the list above

iris1

CREDIT is: (BAD,GOOD,UNKNOWN,)

DEPT is: (HIGH,LOW,)

COLLATRAL is: (ADEQUATE,NONE,)

INCOME is: (>25K,0-10K,10-25K,)

number of rows in HIGH=6.0

number of rows in LOW=5.0

number of rows in MODERATE=3.0

enter the conditon you want, FOR EXAMPLE COLUMNAME='COLUMN VALUE'

credit='unknown' dept='low' collatral='none' income='10-25k'

p(credit='unknown'/HIGH)==2.0

p(dept='low'/HIGH)==2.0

p(collatral='none'/HIGH)==6.0

p(income='10-25k'/HIGH)==2.0

p(credit='unknown'/LOW)==2.0

p(dept='low'/LOW)==3.0

p(collatral='none'/LOW)==3.0

p(income='10-25k'/LOW)==1.0

p(credit='unknown'/MODERATE)==1.0

p(dept='low'/MODERATE)==2.0

p(collatral='none'/MODERATE)==2.0

p(income='10-25k'/MODERATE)==2.0

p(X/HIGH)\*p(maxHIGH)==0.015873015873015872

p(X/LOW)\*p(maxLOW)==0.010285714285714289

p(X/MODERATE)\*p(maxMODERATE)==0.021164021164021163

MODERATE is best