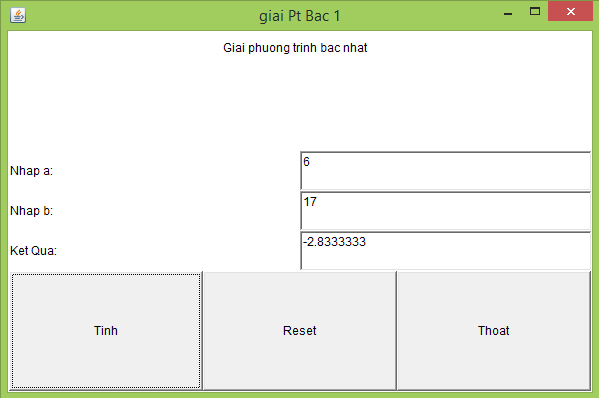
1. **Awt&Swing**

**1.1 Giải phưong trình bậc nhất :**



Code:

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class ptbn\_frame extends Frame implements ActionListener {

Label lb1,lb2,lb3,lb;

TextField tfa,tfb,tfkq;

Button kq,reset,thoat;

Panel pn,pn1,pn2,pn3;

public void Gui()

{

lb = new Label ("Giai phuong trinh bac nhat");

lb1 = new Label("Nhap a: ");

lb2 = new Label("Nhap b: ");

lb3 = new Label("Ket Qua: ");

tfa = new TextField();

tfb = new TextField();

tfkq = new TextField();

kq = new Button("Tinh");

reset = new Button("Reset");

thoat = new Button("Thoat");

kq.addActionListener(this);

reset.addActionListener(this);

thoat.addActionListener(this);

pn = new Panel(new GridLayout(3,1));

pn1 = new Panel(new FlowLayout());

pn2 = new Panel(new GridLayout(3,2));

pn3 = new Panel(new GridLayout(1,3));

pn1.add(lb);

pn2.add(lb1);

pn2.add(tfa);

pn2.add(lb2);

pn2.add(tfb);

pn2.add(lb3);

pn2.add(tfkq);

pn3.add(kq);

pn3.add(reset);

pn3.add(thoat);

pn.add(pn1);

pn.add(pn2);

pn.add(pn3);

add(pn);

setSize(600,400);

show();

}

@Override

public void actionPerformed (ActionEvent e){

if(e.getSource()==kq)

{

int a = Integer.parseInt(tfa.getText());

int b = Integer.parseInt(tfb.getText());

if(a!=0)

tfkq.setText(Float.toString((float)-b/a));

else

if(b==0)

tfkq.setText("Pt vo so nghiem");

else

tfkq.setText("pt vo nghiem");

}

if(e.getSource()==reset)

{

tfa.setText("");

tfb.setText("");

tfkq.setText("");

}

if(e.getSource()==thoat)

{

System.exit(0);

}

}

public ptbn\_frame(String st)

{

super(st);

Gui();

}

public static void main ( String[] args)

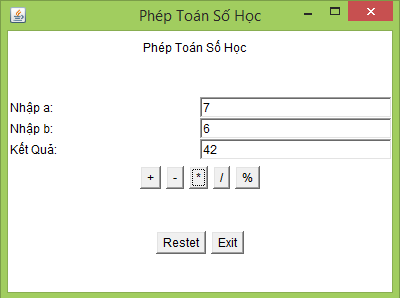
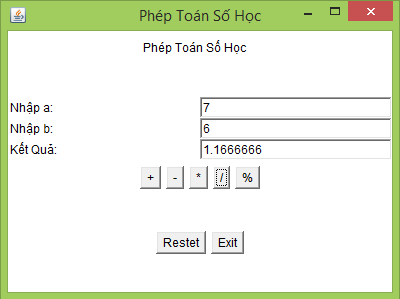
{

new ptbn\_frame("giai Pt Bac 1");

}

}

**1.2 Minh họa các phép toán :**



Code:

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class tinhtoan\_frame extends Frame implements ActionListener {

Label lb,lba,lbb,lbkq;

TextField tfa,tfb,tfkq;

Button cong,tru,nhan,chia,mod,reset,exit;

Panel pn,pn1,pn2,pn3,pn4;

public void Gui(){

lb = new Label("Phép Toán Số Học");

lba = new Label("Nhập a: ");

lbb = new Label("Nhập b: ");

lbkq = new Label("Kết Quả: ");

tfa = new TextField();

tfb = new TextField();

tfkq = new TextField();

cong = new Button("+");

tru = new Button("-");

nhan = new Button("\*");

chia= new Button("/");

mod = new Button("%");

reset = new Button("Restet");

exit = new Button("Exit");

pn = new Panel(new GridLayout(4,1));

pn1 = new Panel(new FlowLayout());

pn2 = new Panel(new GridLayout(3,3));

pn3 = new Panel(new FlowLayout());

pn4 = new Panel(new FlowLayout());

pn1.add(lb);

pn2.add(lba);

pn2.add(tfa);

pn2.add(lbb);

pn2.add(tfb);

pn2.add(lbkq);

pn2.add(tfkq);

pn3.add(cong);

pn3.add(tru);

pn3.add(nhan);

pn3.add(chia);

pn3.add(mod);

pn4.add(reset);

pn4.add(exit);

pn.add(pn1);

pn.add(pn2);

pn.add(pn3);

pn.add(pn4);

add(pn);

cong.addActionListener(this);

tru.addActionListener(this);

nhan.addActionListener(this);

chia.addActionListener(this);

mod.addActionListener(this);

reset.addActionListener(this);

exit.addActionListener(this);

setSize(400,300);

show();

}

@Override

public void actionPerformed(ActionEvent e)

{

if(e.getSource()==cong)

tfkq.setText(Integer.toString(Integer.parseInt(tfa.getText()) + Integer.parseInt(tfb.getText())));

if(e.getSource()==tru)

tfkq.setText(Integer.toString(Integer.parseInt(tfa.getText()) - Integer.parseInt(tfb.getText())));

if(e.getSource()==nhan)

tfkq.setText(Integer.toString(Integer.parseInt(tfa.getText()) \* Integer.parseInt(tfb.getText())));

if(e.getSource()==chia)

tfkq.setText(Float.toString(Float.parseFloat(tfa.getText()) / Float.parseFloat(tfb.getText())));

if(e.getSource()==mod)

tfkq.setText(Float.toString(Float.parseFloat(tfa.getText()) % Float.parseFloat(tfb.getText())));

if(e.getSource()==exit)

System.exit(0);

if(e.getSource()==reset)

{

tfa.setText("");

tfb.setText("");

tfkq.setText("");

}

}

public tinhtoan\_frame(String st)

{

super(st);

Gui();

}

public static void main ( String[] args)

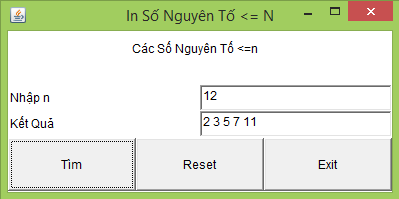
{

new tinhtoan\_frame("Phép Toán Số Học");

}

}

**1.3 In các số nguyên tố nhỏ hơn hoặc bằng số n cho trước :**



Code:

import java.awt.\*;

import java.awt.event.\*;

public class In\_So\_Nguyen extends Frame implements ActionListener {

Label lb,lb1,lb2;

TextField txtn,kq;

Button tim,reset,exit;

Panel pn,pn1,pn2,pn3;

public int Songuyento(int n){

if (n < 2){

return 0;

}

for (int i = 2; i <= Math.sqrt(n); i++)

{

if (n%i == 0)

{

return 0;

}

}

return 1;

}

public void insont(int n){

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

if(Songuyento(i)==1){

kq.setText(kq.getText()+Integer.toString(i)+" ");

}

}

}

public void Gui(){

lb = new Label("Các Số Nguyên Tố <=n");

lb1 = new Label("Nhập n");

lb2 = new Label("Kết Quả");

txtn = new TextField();

kq = new TextField();

tim = new Button("Tìm");

reset = new Button("Reset");

exit = new Button("Exit");

pn = new Panel(new GridLayout(3,1));

pn1 = new Panel( new FlowLayout());

pn2 = new Panel( new GridLayout(2,2));

pn3 = new Panel( new GridLayout(1,3));

pn1.add(lb);

pn2.add(lb1);

pn2.add(txtn);

pn2.add(lb2);

pn2.add(kq);

pn3.add(tim);

pn3.add(reset);

pn3.add(exit);

pn.add(pn1);

pn.add(pn2);

pn.add(pn3);

add(pn);

tim.addActionListener(this);

reset.addActionListener(this);

exit.addActionListener(this);

setSize(400,200);

show();

}

@Override

public void actionPerformed(ActionEvent e) {

if(e.getSource()==tim){

try {

int x=Integer.parseInt(txtn.getText().trim());

insont(x);

}

catch(Exception z){

System.err.printf("vui lòng nhập số");

}

}

if(e.getSource()==reset){

txtn.setText("");

kq.setText("");

}

if(e.getSource()==exit){

System.exit(0);

}

}

public In\_So\_Nguyen(String st){

super(st);

Gui();

}

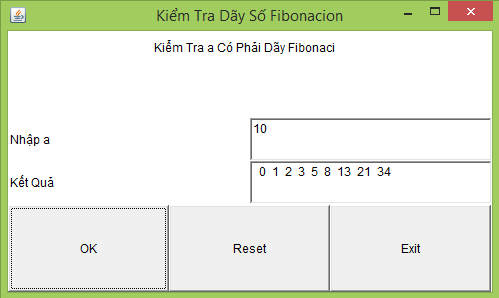
public static void main(String[] args){

new In\_So\_Nguyen("In Số Nguyên Tố <= N");

}

}

**1.4 In dãy Fibonaci ?**



Code:

import java.awt.\*;

import java.awt.event.\*;

public class Fibonaci extends Frame implements ActionListener {

TextField txta,txtkq;

Label lb,lb1,lb2;

Button ok,reset,exit;

Panel pn,pn1,pn2,pn3;

private int Fibo(int k){

int f0=0,f1=1,fn=2;

if(k<0){

return -1;

}else

if((k==0)||(k==1)){

return k;

}else{

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

f0=f1;

f1=fn;

fn=f0+f1;

}

}

return fn;

}

public void Gui(){

lb = new Label("Kiểm Tra a Có Phải Dãy Fibonaci");

lb1 = new Label("Nhập a");

lb2 = new Label("Kết Quả");

txta = new TextField();

txtkq = new TextField();

ok = new Button("OK");

reset = new Button("Reset");

exit = new Button("Exit");

pn = new Panel(new GridLayout(3,1));

pn1 = new Panel(new FlowLayout());

pn2 = new Panel(new GridLayout(2,2));

pn3 = new Panel(new GridLayout(1,3));

pn1.add(lb);

pn2.add(lb1);

pn2.add(txta);

pn2.add(lb2);

pn2.add(txtkq);

pn3.add(ok);

pn3.add(reset);

pn3.add(exit);

pn.add(pn1);

pn.add(pn2);

pn.add(pn3);

add(pn);

ok.addActionListener(this);

reset.addActionListener(this);

exit.addActionListener(this);

setSize(500,300);

show();

}

@Override

public void actionPerformed(ActionEvent e) {

if(e.getSource()==reset){

txta.setText("");

txtkq.setText("");

}

if(e.getSource()== exit){

System.exit(0);

}

if(e.getSource()== ok){

try{

for(int i=0;i<Integer.parseInt(txta.getText())-1;i++){

txtkq.setText(txtkq.getText()+ " " +Integer.toString(Fibo(i)));

}

}catch(Exception a){

System.err.println("Không được nhập số bé hơn 0 và không được nhập kí tự");

}

}

}

public Fibonaci(String st){

super(st);

Gui();

}

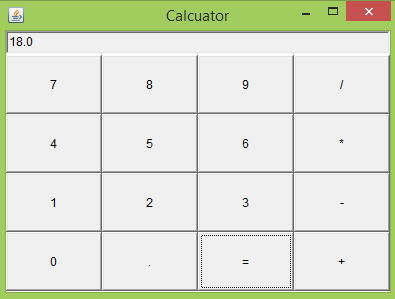
public static void main(String[] args){

new Fibonaci("Kiểm Tra Dãy Số Fibonacion");

}

}

**1.5 Mô tả máy tính điên tử cá nhân**



Code:

import java.awt.\*;

import java.awt.event.\*;

import java.applet.\*;

public class SimpleCalcuator extends Frame implements ActionListener {

private TextField display;

private double arg=0;

private String op="=";

private boolean start=true;

public SimpleCalcuator(){

setTitle("Calcuator");

setSize(400,300);

setLayout(new BorderLayout());

display= new TextField("0");

display.setEditable(false);

add(display,"North");

Panel p = new Panel();

p.setLayout(new GridLayout(4,4));

String buttons[]={"7", "8", "9", "/", "4",

"5", "6", "\*", "1", "2",

"3", "-", "0", ".", "=", "+",};

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

{

Button button = new Button(buttons[i]);

p.add(button);

button.addActionListener(this);

}

add(p, "Center");

addWindowListener(new WindowAdapter()

{

@Override

public void windowClosing(WindowEvent e)

{

System.exit(0);

}

}

);

}

@Override

public void actionPerformed(ActionEvent e) {

String s = e.getActionCommand();

if('0'<=s.charAt(0) && s.charAt(0)<='9' || s.equals("."))

{

if(start)

{

display.setText(s);

}

else

{

display.setText(display.getText() +s);

}

start = false;

}

else

{

if(start)

{

if(s.equals("-"))

{

display.setText(s);

start=false;

}

else{

op=s;

}

}

else{

double x = Double.parseDouble(display.getText());

calculate(x);

op =s;

start = true;

}

}

}

public void calculate(double n){

if(op.equals("+")){

arg += n;

}

else

if(op.equals("-"))

{

arg -= n;

}

else

if(op.equals("\*"))

{

arg \*= n;

}

else

if(op.equals("/"))

{

arg /= n;

}

else

if(op.equals("="))

{

arg = n;

}

display.setText("" + arg);

}

public static void main(String[] args){

SimpleCalcuator frame = new SimpleCalcuator();

frame.show(); }

**1.6 Đổi màu nền :**



Code:

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class Doi\_mau\_nen extends Frame implements ActionListener {

Button green,blue,red,exit,yellow;

Panel pn,pn1;

public void Gui(){

green = new Button("Green");

blue = new Button("Blue");

red = new Button("Red");

yellow = new Button("Yellow");

exit = new Button("Exit");

pn1 = new Panel(new FlowLayout());

pn = new Panel(new FlowLayout());

pn1.add(green);

pn1.add(blue);

pn1.add(red);

pn1.add(yellow);

pn1.add(exit);

pn.add(pn1);

add(pn);

green.addActionListener(this);

blue.addActionListener(this);

red.addActionListener(this);

exit.addActionListener(this);

yellow.addActionListener(this);

setSize(600,400);

show();

}

@Override

public void actionPerformed(ActionEvent e)

{

if(e.getSource()==green)

pn.setBackground(Color.GREEN);

if(e.getSource()==blue)

pn.setBackground(Color.BLUE);

if(e.getSource()==red)

pn.setBackground(Color.RED );

if(e.getSource()==yellow)

pn.setBackground(Color.YELLOW);

if(e.getSource()==exit)

System.exit(0);

}

public Doi\_mau\_nen(String st)

{

super(st);

Gui();

}

public static void main ( String[] args)

{

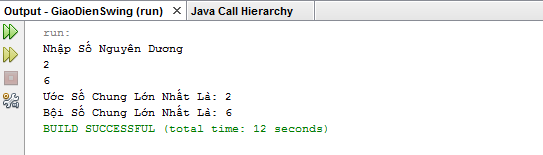
new Doi\_mau\_nen("Phép Toán Số Học");

}

}

1. **Các Luồng Vào Ra**

**2.1 Tìm USCL và BSCNN của 2 số nguyên dương a và b**



Code:

import java.io.\*;

public class In\_US\_BS {

int nhapso() throws IOException {

InputStreamReader luongvao = new InputStreamReader(System.in);

BufferedReader br = new BufferedReader(luongvao);

String s = br.readLine();

return Integer.parseInt(s);

}

public int us(int a, int b) {

while (a != b) {

if (a > b) {

a = a - b;

} else {

b = b - a;

}

}

return a;

}

public int bs(int a, int b) {

return (a \* b) / us(a, b);

}

public static void main(String[] args) {

In\_US\_BS usbs = new In\_US\_BS();

int m = 0, n = 0 , Bs, Us;

try {

do {

System.out.println("Nhập Số Nguyên Dương");

m = usbs.nhapso();

n = usbs.nhapso();

} while (m < 0 && n < 0);

} catch (Exception e) {}

Bs=usbs.us(m, n);

Us=usbs.bs(m, n);

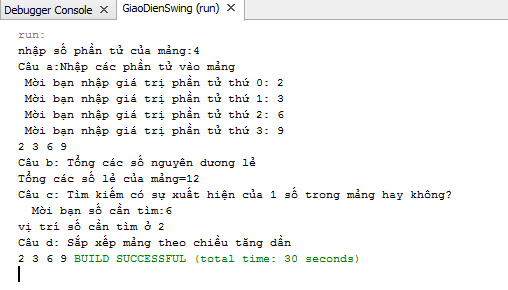
System.out.println("Ước Số Chung Lớn Nhất Là: " + Bs);

System.out.println("Bội Số Chung Lớn Nhất Là: " + Us);

}

}

**2.2 Viết Chương Trình Thực Hiện Các Công Việc Sau:**



Code:

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class Mang {

public int nhap() throws IOException {

InputStreamReader luongvao = new InputStreamReader(System.in);

BufferedReader br = new BufferedReader(luongvao);

String s = br.readLine();

return Integer.parseInt(s);

}

public int tongle(int mang[], int n) {

int tong = 0;

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

if (mang[i] % 2 != 0) {

tong = tong + mang[i];

}

}

return tong;

}

public void xuat(int mang[]) {

int dodai = mang.length;

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

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

}

}

public void sapxep(int mang[]) {

int dodai = mang.length;

int temp;

for (int i = 0; i < dodai - 1; i++) {

for (int j = i + 1; j < dodai; j++) {

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

temp = mang[i];

mang[i] = mang[j];

mang[j] = temp;

}

}

}

}

public void timkiem(int mang[], int n) {

int x = mang.length;

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

if (mang[i] == n) {

System.out.println("vị trí số cần tìm ở " + i);

break;

}

}

}

public static void main(String ag[]) {

Mang mg = new Mang();

int n = 0, tong = 0, x = 0;

try {

do {

System.out.print("nhập số phần tử của mảng:");

n = mg.nhap();

} while (n <= 0);

} catch (Exception e) {

}

int[] mag = new int[n];

System.out.println("Câu a:Nhập các phần tử vào mảng");

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

try {

do {

System.out.print(" Mời bạn nhập giá trị phần tử thứ " + i + ": ");

mag[i] = mg.nhap();

} while (mag[i] < 0);

} catch (Exception e) {

// TODO: handle exception

}

}

tong = mg.tongle(mag, n);

mg.xuat(mag);

System.out.println();

System.out.println("Câu b: Tổng các số nguyên dương lẻ");

System.out.println("Tổng các số lẻ của mảng=" + tong);

System.out.println("Câu c: Tìm kiếm có sự xuất hiện của 1 số trong mảng hay không?");

try {

do {

System.out.print(" Mời bạn số cần tìm:");

x = mg.nhap();

} while (x < 0);

} catch (Exception e) {

// TODO: handle exception

}

mg.timkiem(mag, x);

System.out.println("Câu d: Sắp xếp mảng theo chiều tăng dần");

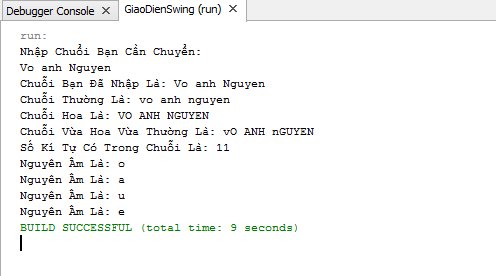
mg.sapxep(mag);

mg.xuat(mag);

}

}

**2.3 Chuỗi Kí Tự:**



Code:

import java.io.\*;

public class Chuoi\_Ki\_Tu {

String nhapso() throws IOException {

InputStreamReader luongvao = new InputStreamReader(System.in);

BufferedReader br = new BufferedReader(luongvao);

String s = br.readLine();

return s;

}

static String thuong(String s) {

String ch = "";

for (int i = 0; i <= s.length() - 1; i++) {

if ((s.charAt(i)) >= 'A' && (s.charAt(i)) <= 'Z') {

ch += (char) (s.charAt(i) + 32);

} else {

ch += (char) s.charAt(i);

}

}

return ch;

}

public void hoa(String s) {

s.trim();

String ch = "";

for (int i = 0; i <= s.length() - 1; i++) {

if ((s.charAt(i)) >= 'a' && (s.charAt(i)) <= 'z') {

ch += (char) (s.charAt(i) - 32);

} else {

ch += (char) (s.charAt(i));

}

}

System.out.println("Chuỗi Hoa Là: " +ch);

}

public void hoathuong(String s){

s.trim();

String ch = "";

for(int i=0;i<=s.length()-1;i++)

{

if ((s.charAt(i)) >= 'A' && (s.charAt(i)) <= 'Z')

{

ch += (char) (s.charAt(i) + 32);

}

else if((s.charAt(i)) >= 'a' && (s.charAt(i)) <= 'z')

{

ch += (char) (s.charAt(i) - 32);

}

else

ch += (char) (s.charAt(i));

}

System.out.println("Chuỗi Vừa Hoa Vừa Thường Là: " +ch);

}

public void demsokt(String s){

int dem=0;

s.trim();

for(int i=0;i<=s.length()-1;i++)

{

if(s.charAt(i)!= ' ')

{

dem++;

}

}

System.out.println("Số Kí Tự Có Trong Chuỗi Là: "+dem);

}

public void nguyenam(String s){

s.trim();

String ch = "";

String thuong = thuong(s);

for(int i=0; i<=thuong.length()-1;i++){

char ngam = thuong.charAt(i);

if( ngam=='u'||ngam=='e'||ngam=='o'||ngam=='a'||ngam=='i'){

System.out.println("Nguyên Âm Là: "+ngam);

}

}

}

public static void main(String[] args) {

Chuoi\_Ki\_Tu kt = new Chuoi\_Ki\_Tu();

String st = "";

try {

System.out.println("Nhập Chuổi Bạn Cần Chuyển: ");

st = kt.nhapso();

} catch (Exception e) {

}

System.out.println("Chuỗi Bạn Đã Nhập Là: " + st);

System.out.println("Chuỗi Thường Là: " + kt.thuong(st));

kt.hoa(st);

kt.hoathuong(st);

kt.demsokt(st);

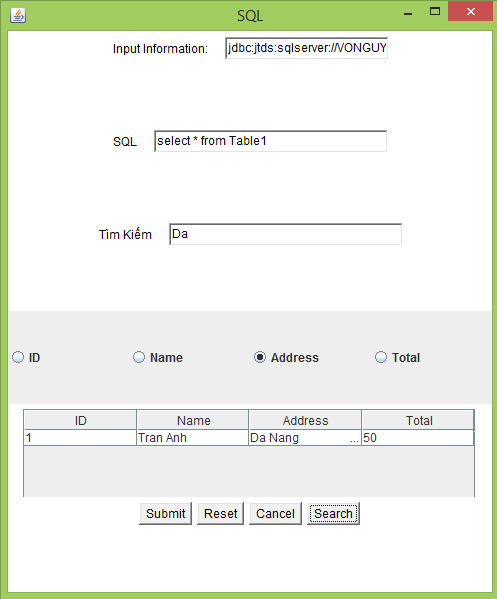
kt.nguyenam(st);

}

}

1. **Lập Trình CSDL**

**3.1 Kết nối csdl với SQL:**



Code:

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

import java.util.Vector;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.\*;

import javax.swing.table.DefaultTableModel;

public class Ket\_Noi\_SQL extends Frame implements ActionListener {

Connection conn = null;

Statement st = null;

ResultSet rs = null;

Label lb1, lb2,lb3;

TextField txt1, txt2,txt3;

JRadioButton rd1,rd2,rd3,rd4;

Button submit, reset, cancel, search;

Panel pn,pn1,pn2,pn3,pn4,pn5,pn6;

String header[] = {"ID", "Name", "Address", "Total"};

DefaultTableModel tblModel = new DefaultTableModel(header,0);

JTable table=new JTable();

JScrollPane scr = new JScrollPane(table);

public Ket\_Noi\_SQL(String st){

super(st);

GUI();

}

public void GUI(){

table.setModel(tblModel);

lb1 = new Label("Input Information:");

lb2 = new Label("SQL");

lb3 = new Label("Tìm Kiếm");

txt1 = new TextField(20);

txt1.setText("jdbc:jtds:sqlserver://VONGUYEN:1433/DATA;instance-SQLEXPRESS");

txt2 = new TextField(30);

txt2.setText("select \* from Table1");

txt3 = new TextField(30);

submit = new Button("Submit");

reset = new Button("Reset");

cancel = new Button("Cancel");

search = new Button("Search");

rd1 = new JRadioButton("ID",true);

rd2 = new JRadioButton("Name");

rd3 = new JRadioButton("Address");

rd4 = new JRadioButton("Total");

ButtonGroup bg = new ButtonGroup();

bg.add(rd1);

bg.add(rd2);

bg.add(rd3);

bg.add(rd4);

pn= new Panel(new GridLayout(6,1));

pn1 = new Panel(new FlowLayout());

pn2 = new Panel(new FlowLayout());

pn3 = new Panel(new FlowLayout());

pn4 = new Panel(new FlowLayout());

pn5 = new Panel(new FlowLayout());

pn6 = new Panel(new GridLayout(1,4));

pn1.add(lb1);

pn1.add(txt1);

pn4.add(lb2);

pn4.add(txt2);

pn5.add(lb3);

pn5.add(txt3);

pn6.add(rd1);

pn6.add(rd2);

pn6.add(rd3);

pn6.add(rd4);

pn2.add(scr);

pn3.add(submit);

pn3.add(reset);

pn3.add(cancel);

pn3.add(search);

pn.add(pn1);

pn.add(pn4);

pn.add(pn5);

pn.add(pn6);

pn.add(pn2);

pn.add(pn3);

add(pn);

submit.addActionListener(this);

reset.addActionListener(this);

cancel.addActionListener(this);

search.addActionListener(this);

setSize(500,600);

show();

}

@Override

public void actionPerformed(ActionEvent e) {

if(e.getActionCommand()=="Submit"){

try {

Class.forName("net.sourceforge.jtds.jdbc.Driver");

conn = DriverManager.getConnection(txt1.getText());

} catch (ClassNotFoundException ex) {

Logger.getLogger(Ket\_Noi\_SQL.class.getName()).log(Level.SEVERE, null, ex);

} catch (SQLException ex) {

Logger.getLogger(Ket\_Noi\_SQL.class.getName()).log(Level.SEVERE, null, ex);

}

try {

st = conn.createStatement();

rs = st.executeQuery(txt2.getText());

Vector data = null;

tblModel.setRowCount(0);

while (rs.next()) {

data = new Vector();

data.add(rs.getString("Id"));

data.add(rs.getString("Name"));

data.add(rs.getString("Address"));

data.add(rs.getString("Total"));

tblModel.addRow(data);

}

} catch (SQLException ex) {

Logger.getLogger(Ket\_Noi\_SQL.class.getName()).log(Level.SEVERE, null, ex);

}

}

if(e.getSource()==cancel){

System.exit(0);

}

if(e.getSource()==reset){

txt1.setText("");

txt2.setText("");

}

if (e.getSource() == search) {

try {

Class.forName("net.sourceforge.jtds.jdbc.Driver");

conn = DriverManager.getConnection("jdbc:jtds:sqlserver://VONGUYEN:1433/DATA;instance-SQLEXPRESS");

st = conn.createStatement();

String sql = "select \* from Table1 ";

if (rd1.isSelected() == true) {

if (txt3.getText().length() > 0) {

sql = sql + " where ID like '%" + txt3.getText() + "%'";

}

} else if (rd2.isSelected() == true) {

if (txt3.getText().length() > 0) {

sql = sql + " where Name like N'%" + txt3.getText() + "%'";

}

} else if (rd3.isSelected() == true) {

if (txt3.getText().length() > 0) {

sql = sql + " where Address like N'%" + txt3.getText() + "%'";

}

}else if (rd4.isSelected() == true) {

if (txt3.getText().length() > 0) {

sql = sql + " where Total like N'%" + txt3.getText() + "%'";

}

}

rs = st.executeQuery(sql);

tblModel.setRowCount(0);

if (rs.isBeforeFirst() == false) {

JOptionPane.showMessageDialog(this, "Dữ Liệu Cần Tìm Không Có ");

return;

}

ResultSetMetaData rstmeta = rs.getMetaData();

int num = rstmeta.getColumnCount();

Vector data = null;

data = new Vector(num);

tblModel.setRowCount(0);

while (rs.next()) {

Vector row = new Vector(num);

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

row.add(rs.getString(i));

}

tblModel.addRow(row);

}

} catch (SQLException ex) {

Logger.getLogger(Ket\_Noi\_SQL.class.getName()).log(Level.SEVERE, null, ex);

} catch (ClassNotFoundException ex) {

Logger.getLogger(Ket\_Noi\_SQL.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

public static void main(String[] args){

new Ket\_Noi\_SQL("SQL");

}

}