## Chương 1

**package** Chuong1;

**import** java.util.Scanner;

**public** **class** Chuong\_1 {

**public** String max(**int** a,**int** b,**int** c)

{

**int** max = a;

**if**(max<b && b<c) max = b;

**else** max = c;

**return** "Max is : " + max;

}

**public** **void** Second\_quadratic(**float** a, **float** b, **float** c) {

**if** (a == 0) {

**if** (b == 0) {

System.***out***.println("Phương trình vô nghiệm!");

} **else** {

System.***out***.println("Phương trình có một nghiệm: "

+ "x = " + (-c / b));

}

**return**;

}

**float** delta = b\*b - 4\*a\*c;

**float** x1;

**float** x2;

**if** (delta > 0) {

x1 = (**float**) ((-b + Math.*sqrt*(delta)) / (2\*a));

x2 = (**float**) ((-b - Math.*sqrt*(delta)) / (2\*a));

System.***out***.println("Phương trình có 2 nghiệm là: "

+ "x1 = " + x1 + " và x2 = " + x2);

} **else** **if** (delta == 0) {

x1 = (-b / (2 \* a));

System.***out***.println("Phương trình có nghiệm kép: "

+ "x1 = x2 = " + x1);

} **else** {

System.***out***.println("Phương trình vô nghiệm!");

}

}

**public** **void** First\_and\_quadratic(**float** a,**float** b,**float** c)

{

**if** (a == 0) {

**if** (b == 0) {

System.***out***.println("No experience!");

} **else** {

System.***out***.println("One experience: "

+ "x = " + (-c / b));

}

}

**float** delta = b\*b - 4\*a\*c;

**float** x1;

**float** x2;

**if** (delta > 0) {

x1 = (**float**) ((-b + Math.*sqrt*(delta)) / (2\*a));

x2 = (**float**) ((-b - Math.*sqrt*(delta)) / (2\*a));

System.***out***.println("two experience: "

+ "x1 = " + x1 + " và x2 = " + x2);

} **else** **if** (delta == 0) {

x1 = (-b / (2 \* a));

System.***out***.println("double experience : "

+ "x1 = x2 = " + x1);

} **else** {

System.***out***.println("No experience");

}

}

**public** String First\_2(**int** a1,**int** b1,**int** c1,**int** a2,**int** b2, **int** c2)

{

**float** D,Dx,Dy,x,y;

D = a1 \* b2 - a2 \* b1;

Dx = c1 \* b2 - c2 \* b1;

Dy = a1 \* c2 - a2 \* c1;

**if** (D == 0)

{

**if** (Dx + Dy == 0) **return** "Unlimited experience";

**else** **return** "No experience";

}**else**

{

x = Dx / D;

y = Dy / D;

**return** "Two experience is : " + x + ", " + y;

}

}

**public** String Middle\_number(**int** a,**int** b,**int** c)

{

**int** middle = a;

**if**(b<a && b>c || b>a && b<c) middle = b;

**else** **if**(c<b && c>a || c>b && c<a) middle = c;

**return** "the middle number is : " + middle;

}

**public** String Charge(**float** start,**float** end)

{

**float** charge;

**if**(end < 18)

{

charge = (end - start)\*45000;

}**else**

{

charge = (end - start)\*60000;

}

**return** "The charge is : " + charge;

}

**public** String Time\_(**int** mm,**int** year)

{

**int** rs = 0;

**if**(year % 400 == 0 || year % 4 == 0 && year % 100 != 0)

{

**switch**(mm)

{

**case** 1:

rs =31;

**break**;

**case** 2:

rs = 29;

**break**;

**case** 3:

rs = 31;

**break**;

**case** 4:

rs = 30;

**break**;

**case** 5:

rs = 31;

**break**;

**case** 6:

rs = 30;

**break**;

**case** 7:

rs = 31;

**break**;

**case** 8:

rs = 31;

**break**;

**case** 9:

rs = 30;

**break**;

**case** 10:

rs = 31;

**break**;

**case** 11:

rs = 30;

**break**;

**case** 12:

rs = 31;

**break**;

**default** :

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

**break**;

}

}**else**

{

**switch**(mm)

{

**case** 1:

rs =31;

**break**;

**case** 2:

rs = 28;

**break**;

**case** 3:

rs = 31;

**break**;

**case** 4:

rs = 30;

**break**;

**case** 5:

rs = 31;

**break**;

**case** 6:

rs = 30;

**break**;

**case** 7:

rs = 31;

**break**;

**case** 8:

rs = 31;

**break**;

**case** 9:

rs = 30;

**break**;

**case** 10:

rs = 31;

**break**;

**case** 11:

rs = 30;

**break**;

**case** 12:

rs = 31;

**break**;

**default** :

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

**break**;

}

}

**return** "year " + year + " have " + rs + " days .";

}

**public** String Tich(**int** n)

{

**int** i = 1;

**float** Sum = 0;

**while**(i <= n)

{

Sum += 1/i;

i++;

}

**return** "Tong la : " + Sum;

}

**public** **float** sum(**int** n)

{

**float** sum = 0;

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

**return** sum;

}

**public** **float** sum\_(**int** n)

{

**float** sum = 0;

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

{

**if**(i%2 != 0) sum += 1/i;

}

**return** sum;

}

**public** String factorial(**int** n)

{

**int** chan = 1;

**int** le = 1;

**int** i = 1;

**do** {

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

{

le \*= i;

i += 1;

}**else**

{

chan \*= i;

i += 1;

}

}**while**(i<=n);

**return** "Le : " + le + " chan : " + chan;

}

**public** String DoST(**int** n)

{

**int** i = n;

**int** tong = 0;

**int** tich = 1;

**do** {

tong += i%10;

tich \*= i%10;

i /= 10;

}**while**(i>0);

**return** "Tong : " + tong + " tich : " + tich;

}

**public** **int** USCLN(**int** a, **int** b)

{

**if** (b == 0) **return** a;

**return** USCLN(b, a % b);

}

**public** **int** BSCNN(**int** a, **int** b)

{

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

}

**public** String SNT(**int** n)

{

**int** dem = 0;

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

{

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

{

dem += 1;

}

}

**if**(dem <= 1) **return** "Yes";

**else** **return** "No";

}

**public** **boolean** check\_square\_num(**int** n)

{

**int** temp = (**int**)Math.*sqrt*(n);

**if**(temp\*temp == n) {

**return** **true**;

}

**else** {

**return** **false**;

}

}

**public** String Check\_(**int** n)

{

**int** SoDao = 0, Rem;

**while** (n != 0)

{

Rem = n % 10;

SoDao = (SoDao \* 10) + Rem;

n = n / 10;

}

**if**(n == SoDao) **return** "Yes";

**else** **return** "No";

}

**public** String ListSNT(**int** n)

{

String rs = "";

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

{

**if**(SNT(i) == "Yes") rs += " " + i;

}

**return** rs;

}

**public** **boolean** check\_perfect\_num(**int** n)

{

**int** sum\_aliquots = 0;

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

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

sum\_aliquots += i;

}

}

**if**(sum\_aliquots == n) {

**return** **true**;

}

**else** {

**return** **false**;

}

}

**public** String low1000()

{

String tmp = "";

**for**(**int** i=1;i<1000;i++)

{

**if**(**this**.check\_perfect\_num(i) == **true**) tmp = " " + i;

}

**return** tmp;

}

**public** **int** fibonacci(**int** n)

{

**int** f0 = 0;

**int** f1 = 1;

**int** fn = 1;

**if** (n < 0) {

**return** -1;

} **else** **if** (n == 0 || n == 1) {

**return** n;

} **else** {

**for** (**int** i = 2; i < n; i++) {

f0 = f1;

f1 = fn;

fn = f0 + f1;

}

}

**return** fn;

}

**public** String showfibo(**int** n)

{

String tmp = "";

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

{

tmp += " " + **this**.fibonacci(i);

}

**return** tmp;

}

**public** String check\_fibo(**int** n)

{

String tmp = "No";

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

{

**if**(n != **this**.fibonacci(i)) **continue**;

tmp = "Yes";

**break**;

}

**return** tmp;

}

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

{

System.***out***.println("-------------------------------- Bài 1 ------------------------------");

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

**int** a = scanner.nextInt();

**int** b = scanner.nextInt();

**int** c = scanner.nextInt();

System.***out***.println("Max la : " + max(a,b,c));

System.***out***.println("-------------------------------- Bài 2 ------------------------------");

**float** a1 = scanner.nextFloat();

**float** b1 = scanner.nextFloat();

**float** c1 = scanner.nextFloat();

First\_and\_quadratic(a1, b1, c1);

System.***out***.println("-------------------------------- Bài 3 ------------------------------");

**float** a3 = scanner.nextFloat();

**float** b3 = scanner.nextFloat();

**float** c3 = scanner.nextFloat();

Second\_quadratic(a3, b3, c3);

System.***out***.println("-------------------------------- Bài 4 ------------------------------");

**int** a4 = scanner.nextInt();

**int** b4 = scanner.nextInt();

**int** c4 = scanner.nextInt();

System.***out***.println(Middle\_number(a4, b4, c4));

System.***out***.println("-------------------------------- Bài 5 ------------------------------");

**float** a5 = scanner.nextFloat();

**float** b5 = scanner.nextFloat();

System.***out***.println(Charge(a5, b5));

System.***out***.println("-------------------------------- Bài 6 ------------------------------");

**int** a6 = scanner.nextInt();

System.***out***.println(Tich(a6));

System.***out***.println("-------------------------------- Bài 7 ------------------------------");

**int** a7 = scanner.nextInt();

System.***out***.println(sum(a7));

System.***out***.println("-------------------------------- Bài 8 ------------------------------");

**int** a8 = scanner.nextInt();

System.***out***.println(sum(a8));

System.***out***.println("-------------------------------- Bài 9 ------------------------------");

**int** a9 = scanner.nextInt();

System.***out***.println(factorial(a9));

System.***out***.println("-------------------------------- Bài 10 ------------------------------");

**int** a10 = scanner.nextInt();

System.***out***.println(DoST(a10));

System.***out***.println("-------------------------------- Bài 11 ------------------------------");

**int** a11 = scanner.nextInt();

System.***out***.println(SNT(a11));

System.***out***.println("-------------------------------- Bài 12 ------------------------------");

**int** a12 = scanner.nextInt();

**if**(check\_square\_num(a12)) System.***out***.println("la so chinh phuong !!!");

**else** System.***out***.println("khong la so chinh phuong !!!");

System.***out***.println("-------------------------------- Bài 13 ------------------------------");

**int** a13 = scanner.nextInt();

System.***out***.println(Check\_(a13));

System.***out***.println("-------------------------------- Bài 14 ------------------------------");

**int** a14 = scanner.nextInt();

System.***out***.println(ListSNT(a14));

System.***out***.println("-------------------------------- Bài 15 ------------------------------");

System.***out***.println(low1000());

System.***out***.println("-------------------------------- Bài 16 ------------------------------");

**int** a16 = scanner.nextInt();

System.***out***.println(showfibo(a16));

System.***out***.println("-------------------------------- Bài 17 ------------------------------");

**int** a17 = scanner.nextInt();

System.***out***.println(check\_fibo(a17));

System.***out***.println("-------------------------------- Bài 18 ------------------------------");

**int** a18 = scanner.nextInt();

**int** b18 = scanner.nextInt();

System.***out***.println(USCLN(a18, b18));

System.***out***.println(BSCNN(a18, b18));

}

}

## Chương 5

package Chuong1;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

public class Chuong\_5 {

public int Input()

{

try {

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

String str = br.readLine();

return Integer.parseInt(str);

}catch(Exception e)

{

e.printStackTrace();

return 0;

}

}

// bài 1 ------------------

public int USCLN(int a, int b) {

if (b == 0) return a;

return USCLN(b, a % b);

}

public int BSCNN(int a, int b) {

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

}

// bài 2 ------------------

//a

public int Sodaonguoc(int n) {

int inverse\_num = 0;

while(n > 0) {

inverse\_num = n%10 + inverse\_num\*10;

n = n/10;

}

return inverse\_num;

}

//b

public Boolean Sodoixung(int sogoc)

{

int sodao = 0;

while (sogoc > 0)

{

int donvi = sogoc % 10;

sodao = sodao\*10 + donvi;

sogoc = sogoc / 10;

}

if (sodao == sogoc)

return true;

else return false;

}

//c

public int fibonacci(int n)

{

int f0 = 0;

int f1 = 1;

int fn = 1;

if (n < 0) {

return -1;

} else if (n == 0 || n == 1) {

return n;

} else {

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

f0 = f1;

f1 = fn;

fn = f0 + f1;

}

}

return fn;

}

public int check\_fibo(int n)

{

int tmp = 0;

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

{

if(n != this.fibonacci(i)) continue;

tmp = i;

break;

}

return tmp;

}

// bài 3 ------------------

//a

public String InputLine() throws IOException

{

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

String str = br.readLine();

return str;

}

public static String Reverse(String str)

{

String tmp = "";

for(int i = str.length() -1;i>=0;i--)

{

tmp += str.charAt(i);

}

return tmp;

}

//b

public String UpperCase(String str)

{

String tmp = "";

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

{

tmp += Character.toUpperCase(str.charAt(i));

}

return tmp;

}

//c

public String LowerCase(String str)

{

String tmp = "";

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

{

tmp += Character.toLowerCase(str.charAt(i));

}

return tmp;

}

//d

public String LowAndUp(String str)

{

str = str.substring(0,1).toUpperCase() + str.substring(1).toLowerCase();

return str;

}

//bài 4 ----------------------

//a

public String upFirstWord(String string)

{

char[] chars = string.toLowerCase().toCharArray();

boolean found = false;

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

if (!found && Character.isLetter(chars[i])) {

chars[i] = Character.toUpperCase(chars[i]);

found = true;

} else if (Character.isWhitespace(chars[i]) || chars[i]=='.' ){

found = false;

}

}

return String.valueOf(chars);

}

//b

public void printfWord(String string)

{

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

{

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

{

System.out.println(string.charAt(i));

}

}

}

//c

public void printfVowel(String string)

{

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

{

switch(string.charAt(i)) {

case 'a':

System.out.print(string.charAt(i));

break;

case 'e':

System.out.print(string.charAt(i));

break;

case 'i':

System.out.print(string.charAt(i));

break;

case 'o':

System.out.print(string.charAt(i));

break;

case 'u':

System.out.print(string.charAt(i));

break;

default :

break;

}

}

}

//d

public int countWord(String string)

{

int count = 0;

int count2 = 0;

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

{

if(string.charAt(i) == ' ')

{

count++;

}

}

count2 = count+1;

return count;

}

//bài 5 ------------------

//a

public int[] arrayInt(int n)

{

int[] tmp = new int[n];

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

{

tmp[i] = Input();

}

return tmp;

}

//b

public int Sum(int a[])

{

int sum = 0;

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

{

if(a[i] > 0 && a[i]%2 !=0)

{

sum += a[i];

}

}

return sum;

}

//c

public String Check(int a[],int k)

{

String sum = "";

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

{

if(a[i] == k)

{

sum = " k o vi tri thu " + i;

break;

}

}

return sum;

}

//d

public void soft(int arr[]) {

int temp = arr[0];

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

for (int j = i + 1; j < arr.length; j++) {

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

temp = arr[j];

arr[j] = arr[i];

arr[i] = temp;

}

}

}

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

{

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

}

}

//e

public void insert(int a[],int k)

{

a[a.length] = k;

soft(a);

}

//bài 6 -------------------------------

//a

public int[][] TwoDimensionalArray(int m,int n)

{

int[][] tmp = new int[m][n];

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

{

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

{

tmp[i][j] = Input();

}

}

return tmp;

}

//b

public int boicua3(int a[][])

{

int tich = 1;

for(int i = 0 ;i<a[1].length;i++)

{

if(a[0][i] %3 == 0)

{

tich \*= a[0][i];

}

}

return tich;

}

//c

public int[] max(int a[][])

{

int tich[] = new int[a.length];

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

{

int max = 0;

for(int j = 0 ;i<a[1].length;j++)

{

max = a[0][0];

if(a[i][j] > max) max = a[i][j];

}

tich[i] = max;

}

return tich;

}

//d

public int[] max(int a[])

{

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

{

a[i] = a[i+1];

}

return a;

}

}

## Bài tập giao diện

Connect.java

package pack;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

public class Connect {

public Connection Conn = null;

public Connection getConnect()

{

try

{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Conn = DriverManager.getConnection("jdbc:sqlserver://localhost:1433;databaseName=BT\_java\_cuoiky\_2;user=sa;password=010101Qa;integratedSecurity=true");

}catch(Exception ex)

{

return null;

}

return Conn;

}

public String getData() throws SQLException

{

String tmp = "";

Connection connection = getConnect();

Statement state = connection.createStatement();

ResultSet rs = state.executeQuery("Select \* from Course");

while(rs.next())

{

tmp += rs.getString(1) + " | " + rs.getString(2) + " | " + rs.getInt(3) + "\n";

}

return tmp;

}

public int insertData(String code,String Name,int credit)

{

Connection connection = getConnect();

try {

PreparedStatement pstmt = connection.prepareStatement("{call dbo.USP\_insert(?,?,?)}");

pstmt.setString(1, code);

pstmt.setString(2, Name);

pstmt.setInt(3, credit);

pstmt.executeUpdate();

return 1;

} catch (SQLException e) {

e.printStackTrace();

return 0;

}

}

public ArrayList<Object> Search(String code) throws SQLException

{

ArrayList<Object> list = new ArrayList<Object>();

Connection connection = getConnect();

Statement state = connection.createStatement();

ResultSet rs = state.executeQuery("Select \* from Course where Code = " + code);

while(rs.next())

{

list.add(rs.getString(2));

list.add(rs.getInt(3));

}

return list;

}

}

CourseManager.java

package pack;

import pack.Insert;

import pack.display;

import pack.Connect;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.JButton;

import java.awt.event.ActionListener;

import java.sql.SQLException;

import java.awt.event.ActionEvent;

public class CourseManager {

private JFrame frame;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

CourseManager window = new CourseManager();

window.frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

public CourseManager() {

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

private void initialize() {

frame = new JFrame();

frame.setBounds(100, 100, 450, 300);

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

frame.getContentPane().setLayout(null);

JLabel lblNewLabel = new JLabel("Course Management");

lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 20));

lblNewLabel.setBounds(96, 23, 238, 40);

frame.getContentPane().add(lblNewLabel);

JButton btnNewButton = new JButton("Add new course");

btnNewButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

try

{

pack.Insert a = new Insert();

a.Run();

System.out.print("Succeed !!!");

}catch(Exception ex)

{

ex.printStackTrace();

}

}

});

btnNewButton.setBounds(126, 85, 171, 23);

frame.getContentPane().add(btnNewButton);

JButton btnDisplayAllCourse = new JButton("Display all course");

btnDisplayAllCourse.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

pack.Connect a = new Connect();

try {

pack.display b = new display(a.getData());

b.Run(a.getData());

} catch (SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

}

});

btnDisplayAllCourse.setBounds(126, 129, 171, 23);

frame.getContentPane().add(btnDisplayAllCourse);

JButton btnSearchCoureBy = new JButton("Search coure by course code");

btnSearchCoureBy.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

pack.Search c = new Search();

c.Run();

}

});

btnSearchCoureBy.setBounds(126, 174, 171, 23);

frame.getContentPane().add(btnSearchCoureBy);

JButton btnExitApplication = new JButton("Exit application");

btnExitApplication.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

System.exit(0);

}

});

btnExitApplication.setBounds(126, 216, 171, 23);

frame.getContentPane().add(btnExitApplication);

}

}

Display.java

package pack;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.JTextArea;

public class display {

private JFrame frame;

public void Run(String text)

{

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

display window = new display(text);

window.frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

private String tmp = null;

public display(String text) {

initialize();

tmp = text;

}

/\*\*

\* Initialize the contents of the frame.

\*/

private void initialize() {

frame = new JFrame();

frame.setBounds(100, 100, 450, 300);

frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

frame.getContentPane().setLayout(null);

JLabel lblNewLabel = new JLabel("List Course");

lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 22));

lblNewLabel.setBounds(129, 11, 193, 32);

frame.getContentPane().add(lblNewLabel);

JTextArea textArea = new JTextArea();

textArea.setBounds(398, 69, -359, 130);

frame.getContentPane().add(textArea);

textArea.setText(tmp);

}

}

Insert.java

package pack;

import java.awt.EventQueue;

import javax.swing.JFrame;

import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.JTextArea;

public class display {

private JFrame frame;

public void Run(String text)

{

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

display window = new display(text);

window.frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

private String tmp = null;

public display(String text) {

initialize();

tmp = text;

}

/\*\*

\* Initialize the contents of the frame.

\*/

private void initialize() {

frame = new JFrame();

frame.setBounds(100, 100, 450, 300);

frame.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

frame.getContentPane().setLayout(null);

JLabel lblNewLabel = new JLabel("List Course");

lblNewLabel.setFont(new Font("Tahoma", Font.BOLD, 22));

lblNewLabel.setBounds(129, 11, 193, 32);

frame.getContentPane().add(lblNewLabel);

JTextArea textArea = new JTextArea();

textArea.setBounds(398, 69, -359, 130);

frame.getContentPane().add(textArea);

textArea.setText(tmp);

}

}

Search.java

**package** pack;

**import** java.awt.EventQueue;

**import** pack.Connect;

**import** javax.swing.JFrame;

**import** javax.swing.JButton;

**import** javax.swing.JTextField;

**import** javax.swing.JLabel;

**import** javax.swing.JTextArea;

**import** java.awt.event.ActionListener;

**import** java.sql.SQLException;

**import** java.util.ArrayList;

**import** java.awt.event.ActionEvent;

**public** **class** Search {

**private** JFrame frame;

**private** JTextField textField;

**public** **void** Run(){

EventQueue.*invokeLater*(**new** Runnable() {

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

**try** {

Search window = **new** Search();

window.frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the application.

\*/

**public** Search() {

initialize();

}

/\*\*

\* Initialize the contents of the frame.

\*/

**private** **void** initialize() {

frame = **new** JFrame();

frame.setBounds(100, 100, 450, 300);

frame.setDefaultCloseOperation(JFrame.***DISPOSE\_ON\_CLOSE***);

frame.getContentPane().setLayout(**null**);

JButton btnNewButton = **new** JButton("Search");

btnNewButton.setBounds(263, 44, 89, 23);

frame.getContentPane().add(btnNewButton);

textField = **new** JTextField();

textField.setBounds(134, 45, 86, 20);

frame.getContentPane().add(textField);

textField.setColumns(10);

JLabel lblNewLabel = **new** JLabel("Enter Code");

lblNewLabel.setBounds(32, 48, 74, 14);

frame.getContentPane().add(lblNewLabel);

JTextArea textArea = **new** JTextArea();

textArea.setBounds(134, 104, 86, 22);

frame.getContentPane().add(textArea);

JTextArea textArea\_1 = **new** JTextArea();

textArea\_1.setBounds(134, 165, 86, 22);

frame.getContentPane().add(textArea\_1);

JLabel lblCodeName = **new** JLabel("Code name");

lblCodeName.setBounds(32, 109, 74, 14);

frame.getContentPane().add(lblCodeName);

JLabel lblCredit = **new** JLabel("Credit");

lblCredit.setBounds(32, 170, 74, 14);

frame.getContentPane().add(lblCredit);

btnNewButton.addActionListener(**new** ActionListener() {

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

Connect a = **new** Connect();

**try** {

ArrayList<Object> b = a.Search(textField.getText());

textArea.setText(b.get(0).toString());

textArea\_1.setText(b.get(1).toString());

} **catch** (SQLException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

}

});

}

}