**Source Code**

package labtest;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.MouseAdapter;

import java.awt.event.MouseEvent;

import java.io.File;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

import java.util.ArrayList;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.JTextField;

import javax.swing.table.DefaultTableModel;

/\*\*

\*

\* @izzaq 265094

\*/

public final class Labtest {

ArrayList<CattleData> cattledatalist = new ArrayList<CattleData>();

String header[] = new String[]{"Weight"," Breed", "ID", "Date"};

DefaultTableModel dtm = new DefaultTableModel(header, 1);

Labtest() {

JFrame frame = new JFrame("Farm Information System");

frame.setSize(470, 550);

JLabel jlabelc = new JLabel("Weight:");

jlabelc.setBounds(10, 50, 60, 20);

frame.add(jlabelc);

JLabel jlabel = new JLabel("Breed:");

jlabel.setBounds(10, 10, 60, 20);

frame.add(jlabel);

JLabel jlabela = new JLabel("ID:");

jlabela.setBounds(10, 30, 60, 20);

frame.add(jlabela);

JLabel jlabelb = new JLabel("Date:");

jlabelb.setBounds(10, 70, 60, 20);

frame.add(jlabelb);

JTextField jtfweight = new JTextField();

jtfweight.setBounds(80, 50, 300, 20);

frame.add(jtfweight);

JTextField jtfbreedtype = new JTextField();

jtfbreedtype.setBounds(80, 10, 300, 20);

frame.add(jtfbreedtype);

JTextField jtfid = new JTextField();

jtfid.setBounds(80, 30, 300, 20);

frame.add(jtfid);

JTextField jtfdate = new JTextField();

jtfdate.setBounds(80, 70, 300, 20);

frame.add(jtfdate);

JButton jbuttoninsert = new JButton("INSERT");

jbuttoninsert.setBounds(180, 100, 90, 20);

frame.add(jbuttoninsert);

//table creation

JTable jtable = new JTable();

jtable.setBounds(20, 140, 400, 400);

frame.add(jtable);

jtable.setModel(dtm);

JScrollPane scrollPane = new JScrollPane(jtable);

scrollPane.setBounds(20, 140, 400, 400);

frame.add(scrollPane);

jtable.getColumnModel().getColumn(0).setPreferredWidth(70);

jtable.getColumnModel().getColumn(1).setPreferredWidth(70);

jtable.getColumnModel().getColumn(2).setPreferredWidth(150);

jtable.getColumnModel().getColumn(3).setPreferredWidth(100);

jbuttoninsert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent ae) {

String weight = jtfweight.getText();

String breed = jtfbreedtype.getText().toUpperCase();

String ID = jtfid.getText();

String date = jtfdate.getText();

if (breed.length() < 5) {

JOptionPane.showMessageDialog(frame, "Breedtype should contain more than 5 char!!!");

return;

}

if (!isInteger(ID)) {

JOptionPane.showMessageDialog(frame, "ID should only contain integer!!!");

return;

}

CattleData cattledata = new CattleData( weight, breed, ID, date);

cattledatalist.add(cattledata);//create object list array

writeData();

}

});

readData();

jtable.addMouseListener(new MouseAdapter() {

@Override

public void mouseClicked(MouseEvent e) {

int row = jtable.getSelectedRow();

jtfweight.setText(dtm.getValueAt(row, 0).toString());

jtfbreedtype.setText(dtm.getValueAt(row, 1).toString());

jtfid.setText(dtm.getValueAt(row, 2).toString());

jtfdate.setText(dtm.getValueAt(row, 3).toString());

}

});

frame.setResizable(false);

frame.setLocationRelativeTo(null);

frame.setLayout(null);

frame.setVisible(true);

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

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

// TODO code application logic here

Labtest labtest = new Labtest();

}

void readData() { //read data from "data.txt" and display on table

try {

File file = new File("data.txt"); //create file

file.createNewFile();//if not exit

FileReader f = new FileReader("data.txt");

StringBuffer sb = new StringBuffer();

while (f.ready()) {

char c = (char) f.read();

if (c == '-') {

System.out.println(sb);

String cattledataarray[] = sb.toString().split(",");

CattleData cattledata = new CattleData(cattledataarray[0], cattledataarray[1], cattledataarray[2], cattledataarray[3]);

cattledatalist.add(cattledata);

sb = new StringBuffer();

} else {

sb.append(c);

}

}

dtm.setRowCount(0); //update table

for (int i = 0; i < cattledatalist.size(); i++) {//populate table using object list

Object[] objs = {cattledatalist.get(i).getWeight(), cattledatalist.get(i).getBreedtype(), cattledatalist.get(i).getId(), cattledatalist.get(i).getDate()};

dtm.addRow(objs);

}

} catch (IOException e) {

}

}

private void writeData() { //write data to file "data.txt"

try (FileWriter f = new FileWriter("data.txt")) {

StringBuilder sb = new StringBuilder();

for (int i = 0; i < cattledatalist.size(); i++) {

sb.append(cattledatalist.get(i).getWeight()).append(",").append(cattledatalist.get(i).getBreedtype()).append(",").append(cattledatalist.get(i).getId ()).append(",").append(cattledatalist.get(i).getDate()).append("-");

}

f.write(sb.toString());

f.close();

} catch (IOException e) {

return;

}

dtm.setRowCount(0); //update table content

for (int i = 0; i < cattledatalist.size(); i++) {//populate table using object list

Object[] objs = {cattledatalist.get(i).getWeight(),cattledatalist.get(i).getBreedtype(), cattledatalist.get(i).getId(), cattledatalist.get(i).getDate()};

dtm.addRow(objs);

}

}

public boolean isInteger(String str) {

if (str == null) {

return false;

}

int length = str.length();

if (length == 0) {

return false;

}

int i = 0;

if (str.charAt(0) == '-') {

if (length == 1) {

return false;

}

i = 1;

}

for (; i < length; i++) {

char c = str.charAt(i);

if (c < '0' || c > '9') {

return false;

}

}

return true;

}

}

**Class**

package labtest;

/\*\*

\*

\* @author User

\*/

public class CattleData {

private String weight,breedtype,date;

final private String id;

CattleData(String weight,String breedtype,String id,String date){

this.weight=weight;

this.breedtype = breedtype;

this.id = id;

this.date = date;

}

void setWeight(String weight){

this.weight = weight;

}

void setBreedtype(String breedtype){

this.breedtype = breedtype;

}

void setDate(String date){

this.date = date;

}

String getWeight(){

return weight;

}

String getBreedtype(){

return breedtype;

}

String getId(){

return id;

}

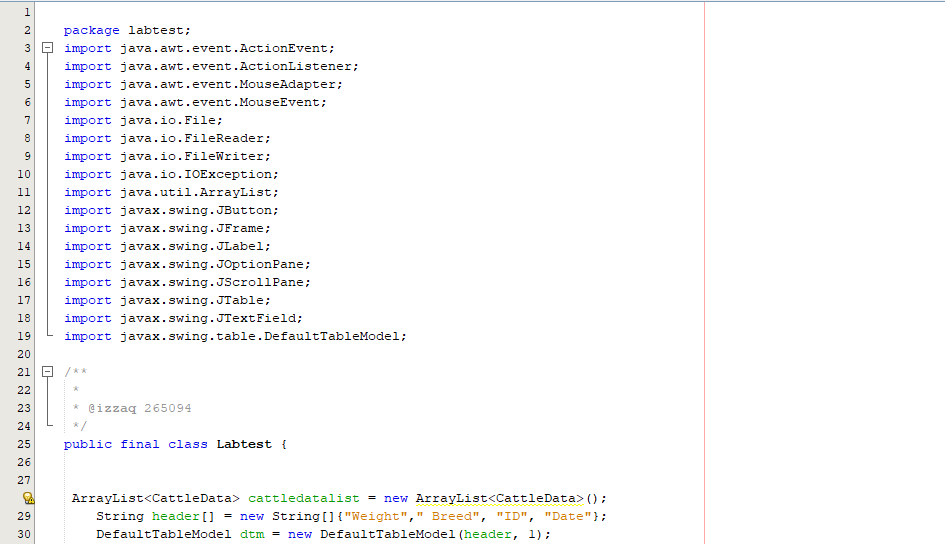
String getDate(){

return date;

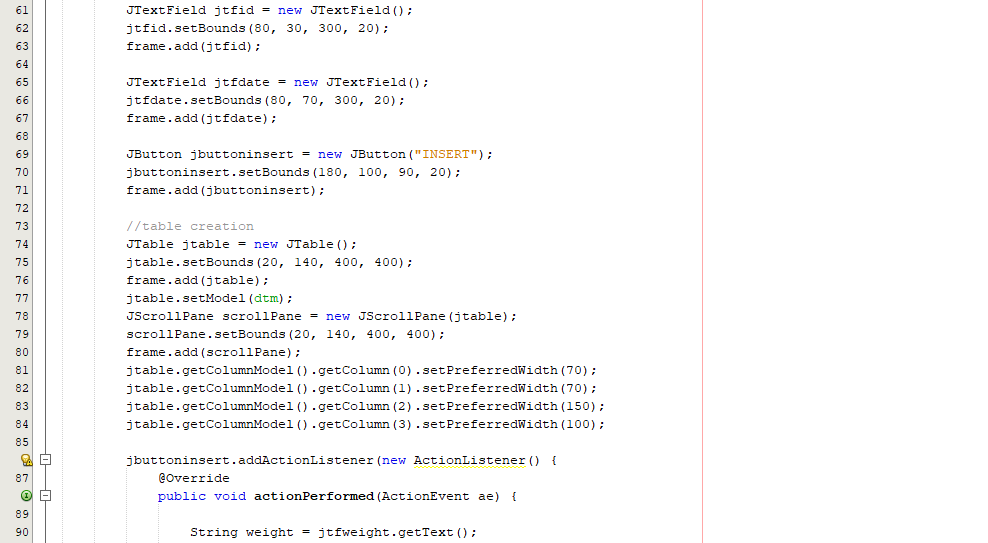
}

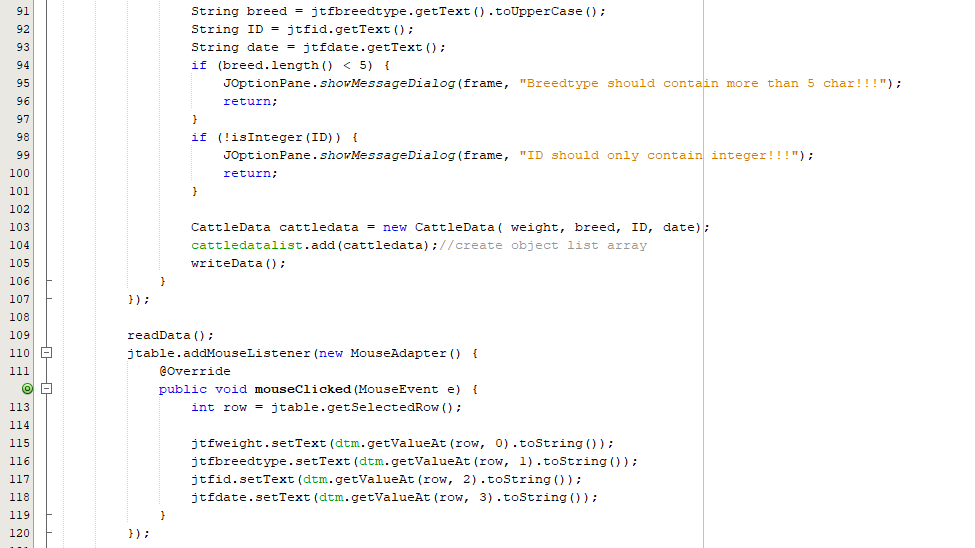
}

**Screenshot source code**

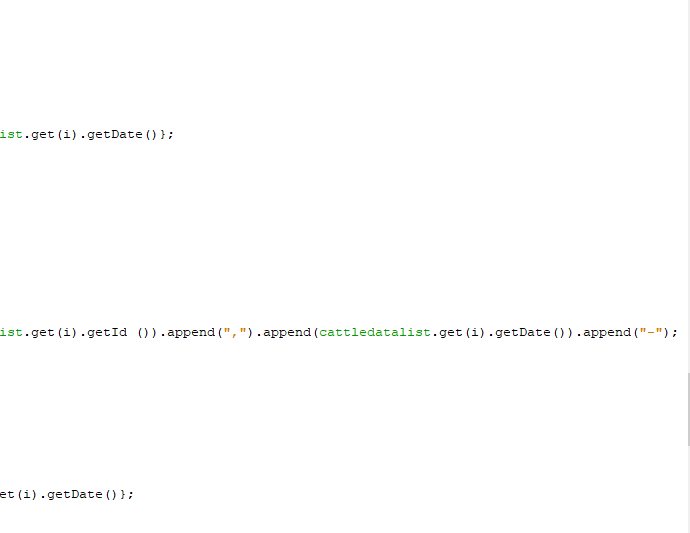


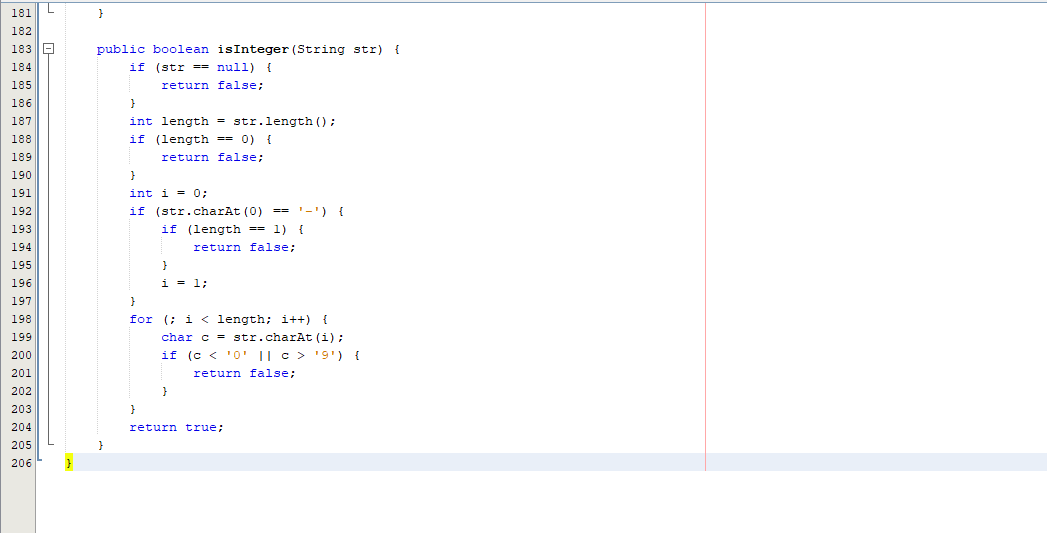




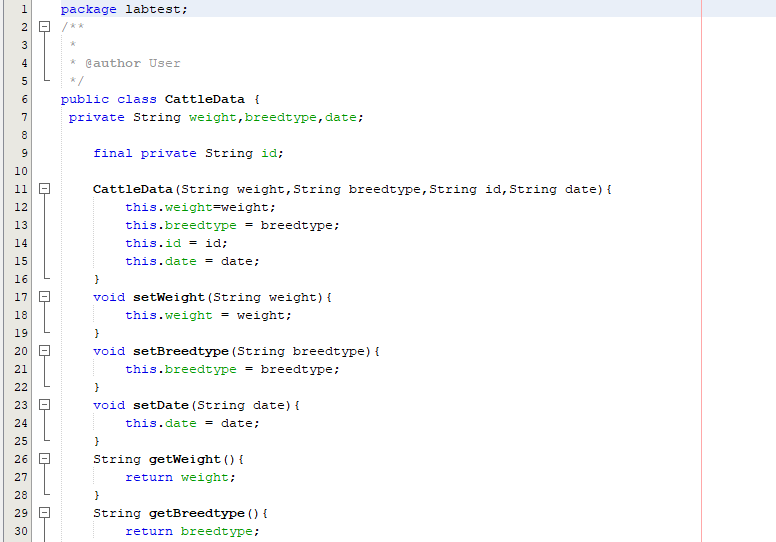






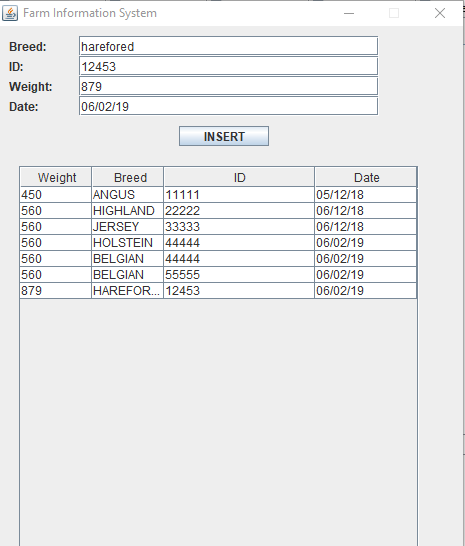


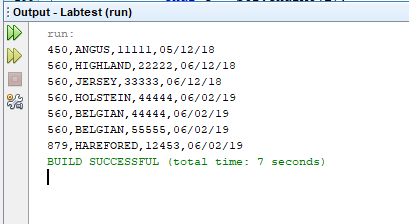
**Class**

****

****

**Interface and Output**

****

****