DBMS ASSIGNMENT-2

**DISEASE SEVERITY TRACKER ON MAIZE**

Name: D. Lalitha sowjanya Roll no: 1602-20-737-019

**ABSTRACT:**

Disease severity tracker on the crop maize is a console-based project designed with sql and java that helps farmers to know the severity of the disease on the crop by just giving the information that they observe on the leaves, surroundings, etc. It gives the information about the disease that the crop is suffering with and the severity of the disease.

**REQUIREMENT ANALYSIS:**

**LIST OF TABLES:**

Deficiency

Diseases

Factors

Severity

**LIST OF ATTRIBUTES WITH THEIR DOMAIN TYPES:**

**Deficiency:**

lcolor varchar2(20)

elem varchar2(20)

did number(5)

**Diseases:**

scolor varchar2(20)

name varchar2(40)

sid number(5)

**Factors:**

fid number(5)

factor varchar2(20)

**Severity:**

sid number(5)

did number(5)

fid number(5)

severity varchar2(20)

**ER DIAGRAM:**

**Diagram

Description automatically generated**

**RELATIONAL MODEL:**

**DDL OPERATIONS:**

**Deficiency:**

create table deficiency(

lcolor VARCHAR2(20),

elem VARCHAR2(20),

did NUMBER(5) PRIMARY KEY);

Text

Description automatically generated

**Graphical user interface

Description automatically generated**

**Diseases:**

create table diseases(

scolor VARCHAR2(20),

name VARCHAR2(20),

sid NUMBER(5) REFERENCES deficiency(did));

Text

Description automatically generated

Graphical user interface

Description automatically generated

**Factors:**

create table factors(

fid NUMBER(5) PRIMARY KEY,

factor VARCHAR2(30));

Text

Description automatically generated

Graphical user interface

Description automatically generated

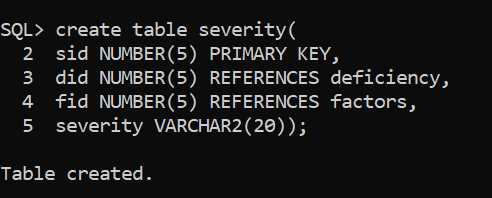
**Severity:**

create table severity(

did NUMBER(5) REFERENCES deficiency,

fid NUMBER(5) REFERENCES factors,

severity VARCHAR2(20));



Graphical user interface

Description automatically generated with medium confidence

**DML OPERATIONS:**

**Deficiency:**

insert into deficiency values(‘lcolor’,’elem’,did);

Text

Description automatically generated

Text

Description automatically generated

**Diseases:**

insert into diseases values(‘scolor’,’name’,sid);

Text

Description automatically generated

Text

Description automatically generated

**Factors:**

insert into factors values(fid,’factor’);

Text

Description automatically generated

Text

Description automatically generated

**Severity:**

insert into severity values(did,fid,’severe’);

**Text

Description automatically generated**

**Text

Description automatically generated**

**A picture containing graphical user interface

Description automatically generated**

**A picture containing text

Description automatically generated**

**Front End Implementation:**

Java SWING is a GUI widget toolkit for Java. It is part of Oracle's Java Foundation Classes (JFC) - an API for providing a graphical user interface (GUI) for Java programs. Swing was developed to provide a more sophisticated set of GUI components than the earlier AWT. Swing provides a look and feel that emulates the look and feel of several platforms and supports a pluggable look and feel that allows applications to have a look and feel unrelated to the underlying platform. It has more powerful and flexible components than AWT. In addition to familiar components such as buttons, check boxes and labels, Swing provides several advanced components such as tabbed panel, scroll panes, trees, tables, and lists.

**Home Page:**

**Code:**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class HomePage extends JFrame{

private JFrame frame= new JFrame();

private JButton b1=new JButton("Deficiency Page");

private JButton b2=new JButton("Diseases Page");

private JButton b3=new JButton("Factors Page");

private JButton b4=new JButton("Severity Page");

private JLabel l1=new JLabel("DIRECT TO");

private JMenuBar mBar;

private JMenu mnuHelp;

private JMenuItem abt;

public HomePage(){

frame.setTitle("Home Page");

frame.setLayout(null);

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

frame.setBounds(100,150,1000,400);

Container c=frame.getContentPane();

frame.getContentPane().add(b1);

frame.getContentPane().add(b2);

frame.getContentPane().add(b3);

frame.getContentPane().add(b4);

frame.getContentPane().add(l1);

initializeMenuBar();

frame.setJMenuBar(mBar);

abt.addActionListener(new HelpMenuActionListener());

JLabel label=new JLabel("HOME PAGE OF DISEASE SEVERITY TRACKER ON MAIZE!!");

JLabel label1=new JLabel();

label1.setIcon(new ImageIcon("C:/Users/sasid/Downloads/maize.jpg"));

Dimension size = label1.getPreferredSize();

label.setBounds(65,5,700,50);

label.setFont(new Font("Serif",Font.PLAIN,20));

label.setForeground(Color.BLUE);

l1.setBounds(670,5,700,50);

l1.setFont(new Font("Serif",Font.PLAIN,20));

l1.setForeground(Color.BLUE);

label1.setBounds(200,60, size.width,size.height);

b1.setBounds(640,60,170,40);

b1.setFont(new Font("Times New Roman",Font.BOLD,17));

b2.setBounds(640,110,170,40);

b2.setFont(new Font("Times New Roman",Font.BOLD,17));

b3.setBounds(640,160,170,40);

b3.setFont(new Font("Times New Roman",Font.BOLD,17));

b4.setBounds(640,210,170,40);

b4.setFont(new Font("Times New Roman",Font.BOLD,17));

b1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent ae) {

new Deficiency();

}

});

b2.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent ae) {

new Diseases();

}

});

b3.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent ae) {

new Factors();

}

});

b4.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent ae) {

new Severity();

}

});

c.add(label);

c.add(label1);

frame.getContentPane().setBackground(Color.CYAN);

frame.setVisible(true);

}

public void initializeMenuBar()

{

mBar=new JMenuBar();

mnuHelp=new JMenu("Help");

abt=new JMenuItem("About");

mnuHelp.add(abt);

mBar.add(mnuHelp);

}

private class HelpMenuActionListener implements ActionListener {

public void actionPerformed(ActionEvent ae) {

if(ae.getSource()==abt)

{

String details;

details = "This project is about tracking the severity of the disease on maize crop"+"\n"+

"It has 4 tables:"+"\n"+

"1.Deficiency table with rows containing deficiency Id as did,deficient element as elem and leaf colour as lcolor"+"\n"+

"2.Diseases table with rows containing disease Id as did,name of the disease as name and colour as scolor"+"\n"+

"3.Factors table with rows containing factor Id as fid and factor as factor"+"\n"+

"4.Severity table with rows containing severity Id as sid, deficiency Id as did,factor Id as fid and severity as severity";

JOptionPane.showMessageDialog(null,details,"INFORMATION", JOptionPane.INFORMATION\_MESSAGE);

}

}

}

public static void main(String args[]){

new HomePage();

}

}

**Compilation:**

Text

Description automatically generated

**Output:**

Graphical user interface, website

Description automatically generated

**Deficiency:**

**Code:**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

import java.util.\*;

public class Deficiency implements ActionListener{

private JFrame f=new JFrame("Deficiency Page");

private JLabel l=new JLabel("");

private JLabel l1=new JLabel("Enter Color,Element,deficiency Id to insert");

private JLabel l2=new JLabel("Enter Color,Element,deficiency Id to update");

private JLabel l3=new JLabel("Enter deficiency Id to delete");

private JLabel l4=new JLabel("Result");

private JLabel l5=new JLabel("DEFICIENCY");

private JButton b1=new JButton("Insert");

private JButton b2=new JButton("Update");

private JButton b3=new JButton("Delete");

private JButton b4=new JButton("Retrieve");

private JTextField t1=new JTextField();

private JTextField t2=new JTextField();

private JTextField t3=new JTextField();

private JTextField t=new JTextField();

private JTextArea t4=new JTextArea();

private JScrollPane scrollBar=new JScrollPane(t4,JScrollPane.VERTICAL\_SCROLLBAR\_ALWAYS,JScrollPane.HORIZONTAL\_SCROLLBAR\_ALWAYS);

public Deficiency() {

f.setDefaultCloseOperation(JFrame.HIDE\_ON\_CLOSE);

f.setBounds(100,150,1000,400);

Container c=f.getContentPane();

f.getContentPane().add(l1);

f.getContentPane().add(l2);

f.getContentPane().add(l3);

f.getContentPane().add(l4);

f.getContentPane().add(l5);

f.getContentPane().add(b1);

f.getContentPane().add(b2);

f.getContentPane().add(b3);

f.getContentPane().add(b4);

f.getContentPane().add(t1);

f.getContentPane().add(t2);

f.getContentPane().add(t3);

f.getContentPane().add(t);

f.getContentPane().add(scrollBar);

scrollBar.setBounds(690,80,250,150);

l.setBounds(20,30,50,50);

l1.setBounds(60,80,250,30);

l1.setOpaque(true);

l1.setBackground(Color.WHITE);

l2.setBounds(60,120,250,30);

l2.setOpaque(true);

l2.setBackground(Color.WHITE);

l3.setBounds(60,160,250,30);

l3.setOpaque(true);

l3.setBackground(Color.WHITE);

l4.setBounds(60,200,250,30);

l4.setOpaque(true);

l4.setBackground(Color.WHITE);

l5.setBounds(430,5,700,50);

l5.setFont(new Font("Serif",Font.PLAIN,20));

l5.setForeground(Color.BLUE);

b1.setBounds(570,80,100,30);

b1.setFont(new Font("Times New Roman",Font.BOLD,17));

b2.setBounds(570,120,100,30);

b2.setFont(new Font("Times New Roman",Font.BOLD,17));

b3.setBounds(570,160,100,30);

b3.setFont(new Font("Times New Roman",Font.BOLD,17));

b4.setBounds(570,200,100,30);

b4.setFont(new Font("Times New Roman",Font.BOLD,17));

t1.setBounds(330,80,220,30);

t2.setBounds(330,120,220,30);

t3.setBounds(330,160,220,30);

t.setBounds(330,200,220,30);

t4.setEditable(false);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

c.add(l);

f.getContentPane().setBackground(Color.ORANGE);

f.setVisible(true);

}

public void actionPerformed(ActionEvent ae){

String s=new String(ae.getActionCommand());

if((s).equals("Insert")){

try{

t.setText("1 row Inserted "+t1.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

StringTokenizer st=new StringTokenizer(t1.getText(),",");

String lcolor=st.nextToken();

String elem=st.nextToken();

int did=Integer.parseInt(st.nextToken());

stmt.executeUpdate("insert into deficiency values('"+lcolor+"','"+elem+"',"+did+")");

con.close();

}

catch (Exception e) {

t.setText("Error Occured!!");

}

t2.setText("");

t3.setText("");

t4.setText("");

}

else if((s).equals("Update")){

try{

t.setText("1 row Updated "+t2.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

StringTokenizer st=new StringTokenizer(t2.getText(),",");

String lcolor=st.nextToken();

String elem=st.nextToken();

int did=Integer.parseInt(st.nextToken());

stmt.executeUpdate("Update deficiency set elem='"+elem+"' where lcolor='"+lcolor+"' and did="+did+"");

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t3.setText("");

t4.setText("");

}

else if((s).equals("Delete")){

try{

t.setText("Deleted 1 row with did "+t3.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

int did=Integer.parseInt(t3.getText());

stmt.executeUpdate("delete from deficiency where did="+did+"");

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t2.setText("");

t4.setText("");

}

else if((s).equals("Retrieve")){

try{

t.setText("Retrieved rows from Deficiency table");

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery("select \* from deficiency");

String str=new String();

while(rs.next())

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

t4.setText(str);

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t2.setText("");

t3.setText("");

}

}

public static void main(String[] args){

new Deficiency();

}

}

**Compilation:**

Logo

Description automatically generated with low confidence

**Output:**

**Graphical user interface, website

Description automatically generated**

**Diseases Page:**

**Code:**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

import java.util.\*;

public class Diseases implements ActionListener{

private JFrame f=new JFrame("Diseases Page");

private JLabel l=new JLabel("");

private JLabel l1=new JLabel("Enter Color,Disease,Disease Id to insert");

private JLabel l2=new JLabel("Enter Color,Disease,Disease Id to update");

private JLabel l3=new JLabel("Enter Disease name to delete");

private JLabel l4=new JLabel("Result");

private JLabel l5=new JLabel("DISEASES");

private JButton b1=new JButton("Insert");

private JButton b2=new JButton("Update");

private JButton b3=new JButton("Delete");

private JButton b4=new JButton("Retrieve");

private JTextField t1=new JTextField();

private JTextField t2=new JTextField();

private JTextField t3=new JTextField();

private JTextField t=new JTextField();

private JTextArea t4=new JTextArea();

private JScrollPane scrollBar=new JScrollPane(t4,JScrollPane.VERTICAL\_SCROLLBAR\_ALWAYS,JScrollPane.HORIZONTAL\_SCROLLBAR\_ALWAYS);

public Diseases() {

f.setDefaultCloseOperation(JFrame.HIDE\_ON\_CLOSE);

f.setBounds(100,150,1000,400);

Container c=f.getContentPane();

f.getContentPane().add(l1);

f.getContentPane().add(l2);

f.getContentPane().add(l3);

f.getContentPane().add(l4);

f.getContentPane().add(l5);

f.getContentPane().add(b1);

f.getContentPane().add(b2);

f.getContentPane().add(b3);

f.getContentPane().add(b4);

f.getContentPane().add(t1);

f.getContentPane().add(t2);

f.getContentPane().add(t3);

f.getContentPane().add(t);

f.getContentPane().add(scrollBar);

scrollBar.setBounds(690,80,250,150);

l.setBounds(20,30,50,50);

l1.setBounds(60,80,250,30);

l1.setOpaque(true);

l1.setBackground(Color.WHITE);

l2.setBounds(60,120,250,30);

l2.setOpaque(true);

l2.setBackground(Color.WHITE);

l3.setBounds(60,160,250,30);

l3.setOpaque(true);

l3.setBackground(Color.WHITE);

l4.setBounds(60,200,250,30);

l4.setOpaque(true);

l4.setBackground(Color.WHITE);

l5.setBounds(430,5,700,50);

l5.setFont(new Font("Serif",Font.PLAIN,20));

l5.setForeground(Color.BLUE);

b1.setBounds(570,80,100,30);

b1.setFont(new Font("Times New Roman",Font.BOLD,17));

b2.setBounds(570,120,100,30);

b2.setFont(new Font("Times New Roman",Font.BOLD,17));

b3.setBounds(570,160,100,30);

b3.setFont(new Font("Times New Roman",Font.BOLD,17));

b4.setBounds(570,200,100,30);

b4.setFont(new Font("Times New Roman",Font.BOLD,17));

t1.setBounds(330,80,220,30);

t2.setBounds(330,120,220,30);

t3.setBounds(330,160,220,30);

t.setBounds(330,200,220,30);

t4.setEditable(false);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

c.add(l);

f.getContentPane().setBackground(Color.ORANGE);

f.setVisible(true);

}

public void actionPerformed(ActionEvent ae){

String s=new String(ae.getActionCommand());

if((s).equals("Insert")){

try{

t.setText("1 row Inserted "+t1.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

StringTokenizer st=new StringTokenizer(t1.getText(),",");

String scolor=st.nextToken();

String name=st.nextToken();

int sid=Integer.parseInt(st.nextToken());

stmt.executeUpdate("insert into diseases values('"+scolor+"','"+name+"',"+sid+")");

con.close();

}

catch (Exception e) {

t.setText("Error Occured!!");

}

t2.setText("");

t3.setText("");

t4.setText("");

}

else if((s).equals("Update")){

try{

t.setText("1 row Updated "+t2.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

StringTokenizer st=new StringTokenizer(t2.getText(),",");

String scolor=st.nextToken();

String name=st.nextToken();

int sid=Integer.parseInt(st.nextToken());

stmt.executeUpdate("Update diseases set name='"+name+"' where scolor='"+scolor+"' and sid="+sid+"");

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t3.setText("");

t4.setText("");

}

else if((s).equals("Delete")){

try{

t.setText("Deleted 1 row with Disease name "+t3.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

String name=t3.getText();

stmt.executeUpdate("delete from diseases where name='"+name+"'");

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t2.setText("");

t4.setText("");

}

else if((s).equals("Retrieve")){

try{

t.setText("Retrieved rows from Diseases table");

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery("select \* from diseases");

String str=new String();

while(rs.next())

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

t4.setText(str);

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t2.setText("");

t3.setText("");

}

}

public static void main(String[] args){

new Diseases();

}

}

**Compilation:**

Text

Description automatically generated

**Output:**

**Graphical user interface, website

Description automatically generated**

**Factors Page:**

**Code:**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

import java.util.\*;

public class Factors implements ActionListener{

private JFrame f=new JFrame("Factors Page");

private JLabel l=new JLabel("");

private JLabel l1=new JLabel("Enter Factor Id,Factor to insert");

private JLabel l2=new JLabel("Enter Factor Id,Factor to update");

private JLabel l3=new JLabel("Enter Factor Id to delete");

private JLabel l4=new JLabel("Result");

private JLabel l5=new JLabel("FACTORS");

private JButton b1=new JButton("Insert");

private JButton b2=new JButton("Update");

private JButton b3=new JButton("Delete");

private JButton b4=new JButton("Retrieve");

private JTextField t1=new JTextField();

private JTextField t2=new JTextField();

private JTextField t3=new JTextField();

private JTextField t=new JTextField();

private JTextArea t4=new JTextArea();

private JScrollPane scrollBar=new JScrollPane(t4,JScrollPane.VERTICAL\_SCROLLBAR\_ALWAYS,JScrollPane.HORIZONTAL\_SCROLLBAR\_ALWAYS);

public Factors() {

f.setDefaultCloseOperation(JFrame.HIDE\_ON\_CLOSE);

f.setBounds(100,150,1000,400);

Container c=f.getContentPane();

f.getContentPane().add(l1);

f.getContentPane().add(l2);

f.getContentPane().add(l3);

f.getContentPane().add(l4);

f.getContentPane().add(l5);

f.getContentPane().add(b1);

f.getContentPane().add(b2);

f.getContentPane().add(b3);

f.getContentPane().add(b4);

f.getContentPane().add(t1);

f.getContentPane().add(t2);

f.getContentPane().add(t3);

f.getContentPane().add(t);

f.getContentPane().add(scrollBar);

scrollBar.setBounds(690,80,250,150);

l.setBounds(20,30,50,50);

l1.setBounds(60,80,250,30);

l1.setOpaque(true);

l1.setBackground(Color.WHITE);

l2.setBounds(60,120,250,30);

l2.setOpaque(true);

l2.setBackground(Color.WHITE);

l3.setBounds(60,160,250,30);

l3.setOpaque(true);

l3.setBackground(Color.WHITE);

l4.setBounds(60,200,250,30);

l4.setOpaque(true);

l4.setBackground(Color.WHITE);

l5.setBounds(430,5,700,50);

l5.setFont(new Font("Serif",Font.PLAIN,20));

l5.setForeground(Color.BLUE);

b1.setBounds(570,80,100,30);

b1.setFont(new Font("Times New Roman",Font.BOLD,17));

b2.setBounds(570,120,100,30);

b2.setFont(new Font("Times New Roman",Font.BOLD,17));

b3.setBounds(570,160,100,30);

b3.setFont(new Font("Times New Roman",Font.BOLD,17));

b4.setBounds(570,200,100,30);

b4.setFont(new Font("Times New Roman",Font.BOLD,17));

t1.setBounds(330,80,220,30);

t2.setBounds(330,120,220,30);

t3.setBounds(330,160,220,30);

t.setBounds(330,200,220,30);

t4.setEditable(false);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

c.add(l);

f.getContentPane().setBackground(Color.ORANGE);

f.setVisible(true);

}

public void actionPerformed(ActionEvent ae){

String s=new String(ae.getActionCommand());

if((s).equals("Insert")){

try{

t.setText("1 row Inserted "+t1.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

StringTokenizer st=new StringTokenizer(t1.getText(),",");

int fid=Integer.parseInt(st.nextToken());

String factor=st.nextToken();

stmt.executeUpdate("insert into factors values("+fid+",'"+factor+"')");

con.close();

}

catch (Exception e) {

t.setText("Error Occured!!");

}

t2.setText("");

t3.setText("");

t4.setText("");

}

else if((s).equals("Update")){

try{

t.setText("1 row Updated "+t2.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

StringTokenizer st=new StringTokenizer(t2.getText(),",");

int fid=Integer.parseInt(st.nextToken());

String factor=st.nextToken();

stmt.executeUpdate("Update factors set factor='"+factor+"' where fid="+fid+"");

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t3.setText("");

t4.setText("");

}

else if((s).equals("Delete")){

try{

t.setText("Deleted 1 row with fid "+t3.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

int fid=Integer.parseInt(t3.getText());

stmt.executeUpdate("delete from factors where fid="+fid+"");

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t2.setText("");

t4.setText("");

}

else if((s).equals("Retrieve")){

try{

t.setText("Retrieved rows from Factors table");

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery("select \* from factors");

String str=new String();

while(rs.next())

str=str+(rs.getInt(1)+" "+rs.getString(2)+"\n");

t4.setText(str);

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t2.setText("");

t3.setText("");

}

}

public static void main(String[] args){

new Factors();

}

}

**Compilation:**

Text

Description automatically generated

**Output:**

Graphical user interface

Description automatically generated

**Severity Page:**

**Code:**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

import java.util.\*;

public class Severity implements ActionListener{

private JFrame f=new JFrame("Severity Page");

private JLabel l=new JLabel("");

private JLabel l1=new JLabel("Enter sid,did,fid,Severity to insert");

private JLabel l2=new JLabel("Enter sid,did,fid to update");

private JLabel l3=new JLabel("Enter sid to delete");

private JLabel l4=new JLabel("Result");

private JLabel l5=new JLabel("SEVERITY");

private JButton b1=new JButton("Insert");

private JButton b2=new JButton("Update");

private JButton b3=new JButton("Delete");

private JButton b4=new JButton("Retrieve");

private JTextField t1=new JTextField();

private JTextField t2=new JTextField();

private JTextField t3=new JTextField();

private JTextField t=new JTextField();

private JTextArea t4=new JTextArea();

private JScrollPane scrollBar=new JScrollPane(t4,JScrollPane.VERTICAL\_SCROLLBAR\_ALWAYS,JScrollPane.HORIZONTAL\_SCROLLBAR\_ALWAYS);

public Severity() {

f.setDefaultCloseOperation(JFrame.HIDE\_ON\_CLOSE);

f.setBounds(100,150,1000,400);

Container c=f.getContentPane();

f.getContentPane().add(l1);

f.getContentPane().add(l2);

f.getContentPane().add(l3);

f.getContentPane().add(l4);

f.getContentPane().add(l5);

f.getContentPane().add(b1);

f.getContentPane().add(b2);

f.getContentPane().add(b3);

f.getContentPane().add(b4);

f.getContentPane().add(t1);

f.getContentPane().add(t2);

f.getContentPane().add(t3);

f.getContentPane().add(t);

f.getContentPane().add(scrollBar);

scrollBar.setBounds(690,80,250,150);

l.setBounds(20,30,50,50);

l1.setBounds(60,80,250,30);

l1.setOpaque(true);

l1.setBackground(Color.WHITE);

l2.setBounds(60,120,250,30);

l2.setOpaque(true);

l2.setBackground(Color.WHITE);

l3.setBounds(60,160,250,30);

l3.setOpaque(true);

l3.setBackground(Color.WHITE);

l4.setBounds(60,200,250,30);

l4.setOpaque(true);

l4.setBackground(Color.WHITE);

l5.setBounds(430,5,700,50);

l5.setFont(new Font("Serif",Font.PLAIN,20));

l5.setForeground(Color.BLUE);

b1.setBounds(570,80,100,30);

b1.setFont(new Font("Times New Roman",Font.BOLD,17));

b2.setBounds(570,120,100,30);

b2.setFont(new Font("Times New Roman",Font.BOLD,17));

b3.setBounds(570,160,100,30);

b3.setFont(new Font("Times New Roman",Font.BOLD,17));

b4.setBounds(570,200,100,30);

b4.setFont(new Font("Times New Roman",Font.BOLD,17));

t1.setBounds(330,80,220,30);

t2.setBounds(330,120,220,30);

t3.setBounds(330,160,220,30);

t.setBounds(330,200,220,30);

t4.setEditable(false);

b1.addActionListener(this);

b2.addActionListener(this);

b3.addActionListener(this);

b4.addActionListener(this);

c.add(l);

f.getContentPane().setBackground(Color.ORANGE);

f.setVisible(true);

}

public void actionPerformed(ActionEvent ae){

String s=new String(ae.getActionCommand());

if((s).equals("Insert")){

try{

t.setText("1 row Inserted "+t1.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

StringTokenizer st=new StringTokenizer(t1.getText(),",");

int sid=Integer.parseInt(st.nextToken());

int did=Integer.parseInt(st.nextToken());

int fid=Integer.parseInt(st.nextToken());

String severity=st.nextToken();

stmt.executeUpdate("insert into severity values("+sid+","+did+","+fid+",'"+severity+"')");

con.close();

}

catch (Exception e) {

t.setText("Error Occured!!");

}

t2.setText("");

t3.setText("");

t4.setText("");

}

else if((s).equals("Update")){

try{

t.setText("1 row Updated "+t2.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

StringTokenizer st=new StringTokenizer(t2.getText(),",");

int sid=Integer.parseInt(st.nextToken());

int did=Integer.parseInt(st.nextToken());

int fid=Integer.parseInt(st.nextToken());

String severity=st.nextToken();

stmt.executeUpdate("Update severity set severity='"+severity+"' where sid="+sid+" and did="+did+" and fid="+fid+"");

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t3.setText("");

t4.setText("");

}

else if((s).equals("Delete")){

try{

t.setText("Deleted 1 row with sid"+t3.getText());

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

int sid=Integer.parseInt(t3.getText());

stmt.executeUpdate("delete from severity where sid="+sid+"");

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t2.setText("");

t4.setText("");

}

else if((s).equals("Retrieve")){

try{

t.setText("Retrieved rows from Severity table");

Class.forName("oracle.jdbc.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","lalitha","vasavi");

Statement stmt=con.createStatement();

ResultSet rs=stmt.executeQuery("select \* from severity");

String str=new String();

while(rs.next())

str=str+(rs.getInt(1)+" "+rs.getInt(2)+" "+rs.getInt(3)+" "+rs.getString(4)+"\n");

t4.setText(str);

con.close();

}

catch(Exception e){

t.setText("Error Occured!!");

}

t1.setText("");

t2.setText("");

t3.setText("");

}

}

public static void main(String[] args){

new Severity();

}

}

**Compilation:**



**Output:**

**Graphical user interface

Description automatically generated**

**TESTING**

**To Know about the project:** clicking on help menu shows a menu item named about by clicking on it user can know about the project.

Graphical user interface

Description automatically generated

Graphical user interface, text, application

Description automatically generated

**For going to other tables:** click on the buttons given with names of the tables.

**Example:** By clicking on deficiency button the page will be directed to deficiency page.

Graphical user interface

Description automatically generated

**To retrieve rows from tables:** click on retrieve button to get rows from table.

Graphical user interface

Description automatically generated

**To make Insertions in tables:** Insertion of values into table can be done by giving the values in the format given in the label and then click on insert button. To verify the insertion click on retrieve and check.

Graphical user interface

Description automatically generated

Graphical user interface, table

Description automatically generated

**Insertion in wrong format gives error message:**

Graphical user interface

Description automatically generated

**To Update rows in tables:** Updation Of values into table can be done by giving the values in the format given in the label and then click on update. Click on retrieve and check for updation.

Graphical user interface

Description automatically generated

Graphical user interface, table

Description automatically generated

**Updation in wrong format gives error message:**

**Graphical user interface

Description automatically generated**

**For deletions in table:** Enter the values according to the format given in the label and click on delete button to make deletions. Click on retrieve button to check the deletion.

Graphical user interface

Description automatically generated

Graphical user interface

Description automatically generated

**Deletion in wrong format gives error message:**

**Graphical user interface

Description automatically generated**

**GitHubLink:** [**https://github.com/Lalitha-Sowjanya/Disease-Severity-Tracker-on-Maize.git**](https://github.com/Lalitha-Sowjanya/Disease-Severity-Tracker-on-Maize.git)