***1. How to create a Mail Validation?***

System.exit(0);

private void Reset(){

txtMail.setText("");

txtpass.setText("");

}

private void Loging() {

String m, p;

int atpos, dotpos;

m = txtMail.getText();

p = txtpass.getText();

if (txtMail.getText().isEmpty()) {

JOptionPane.showMessageDialog(rootPane, "Enter your Mail");

txtMail.requestFocus();

} else {

atpos = m.indexOf("@");

dotpos = m.lastIndexOf(".");

if (atpos < 1 || dotpos <= atpos + 2 || dotpos + 2 >= m.length()) {

JOptionPane.showMessageDialog(rootPane, "Invalide your Mail", "Warning", JOptionPane.WARNING\_MESSAGE);

txtMail.requestFocus();

} else {

if (txtpass.getText().isEmpty()) {

JOptionPane.showMessageDialog(rootPane, "Enter your Password");

txtpass.requestFocus();

} else {

if (p.length() < 9) {

JOptionPane.showMessageDialog(rootPane, "Invalide your Password", "Erroe", JOptionPane.ERROR\_MESSAGE);

txtpass.requestFocus();

} else {

JOptionPane.showMessageDialog(rootPane, "Congratulation!!!!");

}

}

}

}

}

***2. How to create a Form Validation with Mail?***

public class Application\_Form extends javax.swing.JFrame {

String n, a, c, g, em, p;

String d="";

String gap=" ";

File file =new File("D:\\noman.txt");

DefaultTableModel model;

public FormVWMAndP() {

initComponents();

model=new DefaultTableModel();

tblStudent.setModel(model);

model.addColumn("Name");

model.addColumn("Address");

model.addColumn("Counry");

model.addColumn("Gender");

model.addColumn("Degree");

model.addColumn("Gmail");

model.addColumn("Password");

}

System.exit(0);

private void AllReset() {

txtName.setText("");

txtAddress.setText("");

txtMail.setText("");

txtpass.setText("");

GenderGroup.clearSelection();

cmBox.setSelectedIndex(0);

txtTArea.setText("");

for (int i = PanelDegree.getComponentCount() - 1; i > -1; i--) {

JCheckBox checkBox = (JCheckBox) PanelDegree.getComponent(i);

if (checkBox.isSelected()) {

checkBox.setSelected(false);

}

}

if (tblStudent.getRowCount() > 0) {

for (int i = tblStudent.getRowCount()-1; i > -1; i--) {

model.removeRow(i);

}

}

}

private void ShoeDispTable(){

InputStream is =null;

try {

is=new FileInputStream(file);

Scanner sc=new Scanner(is);

String Arr[];

while (sc.hasNext()) {

String l = sc.nextLine();

if (l.indexOf(",")>1){

Arr=l.split(",");

}else{

Arr=l.split("\n");

}

Object data[]=new Object[Arr.length];

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

data[i] = Arr[i];

}

model.addRow(data);

}

} catch (Exception e) {

e.printStackTrace();

}

}

private void WiriteAll() throws IOException {

BufferedWriter buf = null;

try {

buf = new BufferedWriter(new FileWriter(file, true));

buf.write(n + "," + a + "," + c + "," + g + "," + d + "," + em + "," + p);

buf.newLine();

JOptionPane.showMessageDialog(rootPane, "Document Saved");

} catch (Exception e) {

e.printStackTrace();

} finally {

if (buf != null) {

buf.close();

}

}

}

Save button Problem

if (txtName.getText().isEmpty()) {

JOptionPane.showMessageDialog(rootPane, "Please Enter your Name");

txtName.requestFocus();

} else {

if (txtAddress.getText().isEmpty()) {

JOptionPane.showMessageDialog(rootPane, "Please Enter your Address");

txtAddress.requestFocus();

} else {

if (cmBox.getSelectedIndex() == 0) {

JOptionPane.showMessageDialog(rootPane, "Please select Country");

cmBox.requestFocus();

} else {

}

}

}

model.addRow(new Object[]{n, a, c, g, d, em, p});

try {

WiriteAll();

} catch (Exception e) {

e.printStackTrace();

}d="";

}