

Series 2: Making Database Software is Easy with Java.....

Simple Employee Database

Ver 1.0



Volume 1.0

by

Goen-Ghin

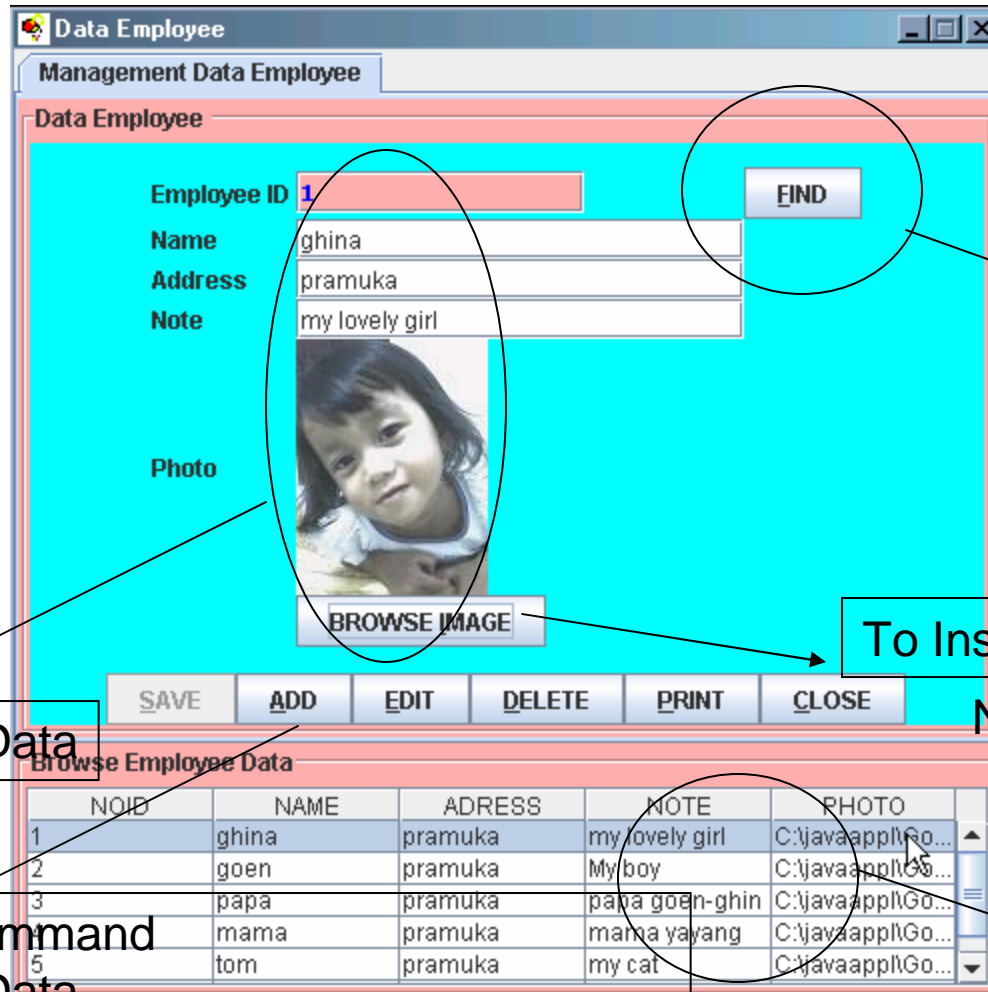
<http://javageo.com>

<http://groups.yahoo.com/group/JavaGeo/>

Introduction

- This volume 1, will bring you to the world of java programming to create a software which connected to a database easily
- It's help you to know how to add,edit,find,delete and print data from a database system
- help you to learn how to make a database application in java
- And also give you an idea to design java swing components to any software that you want to create !

The Snapshot



Jbutton
Help you to find a data by ID

To Insert Image Data

To Insert Data

Note: use small size picture

To Run command
-to Insert Data
-to Update Data
-to Delete Data
-to Print Data (preview and print to paper)

JTable
click and keypress then You will have data

Class and Method

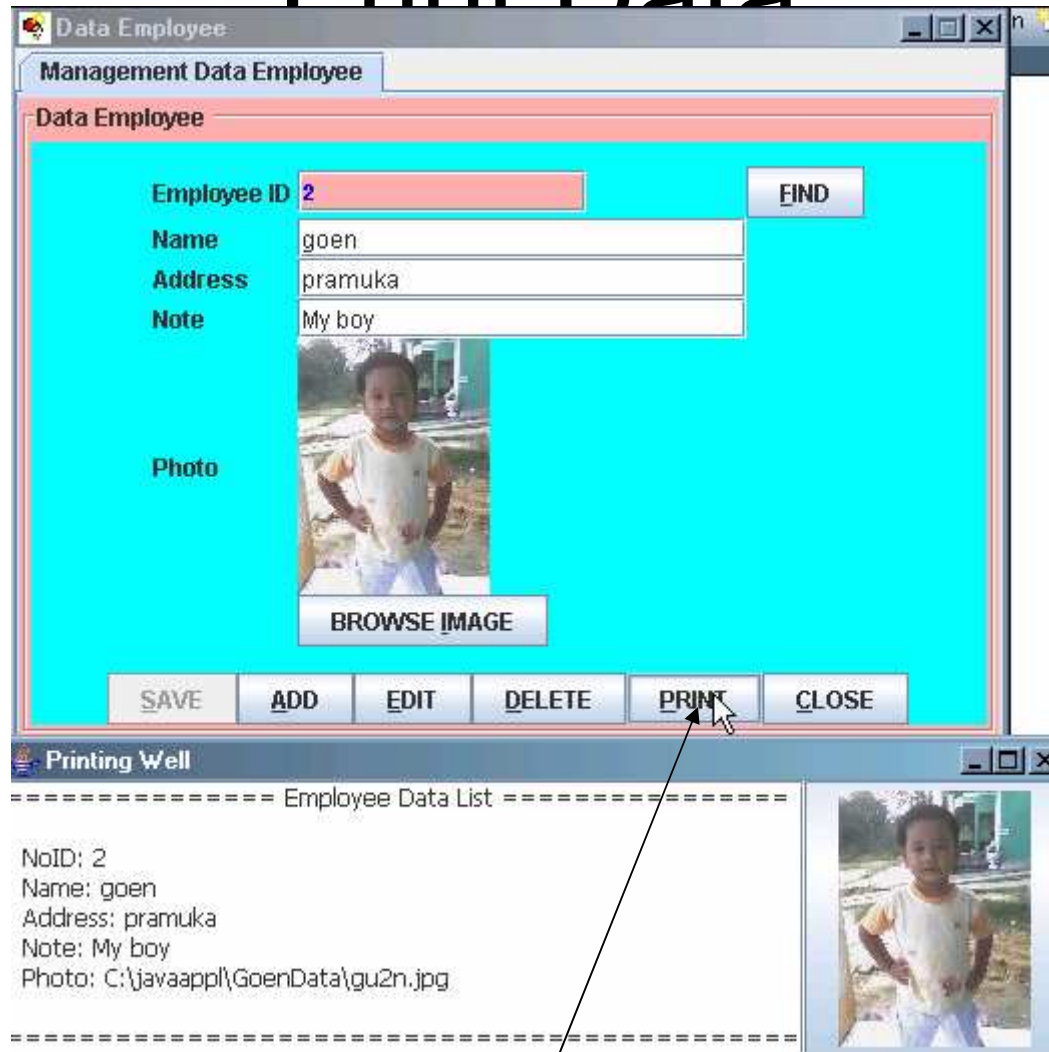
Class

Employee
ConnectData
Photo
PrintData

Method

ambilData()
clearText()
hapusTabel()
findData()
tampilData()

Print Data



By Click Button PRINT you can print and preview your data (print image and text)

set a connection

```
class ConnectData
{
    private Connection connection = null;
    private Statement statement = null;
    private ResultSet resultSet = null;
    private PreparedStatement prepareSet = null;
    private String driver = "org.hsqldb.jdbcDriver";
    private String URL = "jdbc:hsqldb:file:test";

    ConnectData()
    {
        try {
            Class.forName(driver);
            connection = DriverManager.getConnection(URL);
        }
        catch (ClassNotFoundException SQLLe) {
            JOptionPane.showMessageDialog(null, SQLLe.toString(), "Error", JOptionPane.ERROR_MESSAGE);
        }
        catch (Exception ex) {
            JOptionPane.showMessageDialog(null, ex.toString(), "Error", JOptionPane.ERROR_MESSAGE);
        }
    }
}
```

Image data

```
//start photo
class Photo extends JTextField {
    Image img;
    public Image Photo (String ph) {
        if (ph == null) JOptionPane.showMessageDialog(null, "No Data", "Error", JOptionPane.ERROR_MESSAGE);

        else img = this.getToolkit().getImage(ph);

        return img;
    }
    public void paint (Graphics g) {
        g.drawImage(img, 0, 0, 200, 200, this);
    }
}
```

Put Image on as ImageIcon of a JButton

```
else if(stringChoose=="BROWSE IMAGE")
{
    int option = myFileChooser.showOpenDialog(Employee.this);
    String dir = myFileChooser.getCurrentDirectory().toString();
    myStrPhoto = dir+"\\"+((myFileChooser.getSelectedFile() !=null)? myFileChooser.getSelectedFile().getName():"nothing");
    if (option == JFileChooser.APPROVE_OPTION) {
        myFoto.setIcon(new ImageIcon(photo.Photo(myStrPhoto)));
    }
}
```

Query

```
try{
String sql_tambah = "INSERT INTO Employee "+
                    "VALUES("+ a +","+ b +","+c+","+d+","+myStrPhoto+)";

                    dt = new ConnectData();
                    dt.prepareStatement(sql_tambah);
                    tampilData();
                    clearText();
}catch(Exception se){
JOptionPane.showMessageDialog(null,"Data Exist");
}
```

```
try{

String sql_ubah = "UPDATE Employee "+
"SET Name = "+ bb +",Adress = "+ cc +",Note="+dd+",Photo="+myStrPhoto+" WHERE NoID = "+ aa +"";
dt = new ConnectData();
dt.prepareStatement(sql_ubah);
tampilData();
clearText();

}catch(Exception se){
JOptionPane.showMessageDialog(null,"Failed to Update "+se);
}
```

Mouse and Key Adapter

```
• MouseAdapter ma = new MouseAdapter(){
•     public void mouseClicked(MouseEvent me){
•         ambilData();
•         myButtoninput.setEnabled(false);
•         myButtongedit.setEnabled(true);
•         myButtongdelete.setEnabled(true);
•         myButtongkosong.setEnabled(true);
•     }
• };
•
• KeyAdapter ka = new KeyAdapter(){
•     public void keyPressed(KeyEvent ke){
•         ambilData();
•         myButtoninput.setEnabled(false);
•         myButtongedit.setEnabled(true);
•         myButtongdelete.setEnabled(true);
•         myButtongkosong.setEnabled(true);
•     }
• };
•
• myDataTable.addKeyListener(ka);
•
• myDataTable.addMouseListener(ma);
•
```

For make GUI interactively by using MOUSE and KEY for selecting data in Jtable
To display in JPanel

Source Code

- The completed source code is inside the directory UKM in file javageo group or
- <http://www.javageo.com>
- You can download the source code...
Play ,Modify and be succesfull with it...!
Good luck !!

See you on next volume and series !!