

# Tutorial 3 layer with JDBC

## 1. Tạo Database Employees

### Tạo table Employee

```
employee_id int(12) primary key auto_increment,  
employee_name varchar(45),  
phonenumbers int(12),  
address varchar(45)
```

## 2. Các bước cơ bản sử dụng JDBC

### 2.1. Đăng ký Driver (bao gồm thêm thư viện JDBC Driver và đăng ký Driver)

- Thêm thư viện JDBC bằng cách kích chuột phải vào Libraries chọn Add Libraries và chọn MySQL JDBC Driver

```
// nạp driver  
Driver driver = new org.gjt.mm.mysql.Driver();  
DriverManager.registerDriver(driver); // đăng ký driver
```

### 2.2. Mở kết nối đến cơ sở dữ liệu

```
String url = "jdbc:mysql://localhost:3306/employees";  
conn = DriverManager.getConnection(url, "root", ""); // tạo kết nối đến cơ sở  
dữ liệu
```

### 2.3. Tạo và thực thi các câu lệnh sql

```
// tạo và thực thi các câu lệnh sql  
// cung cấp thông tin trích rút từ csdl và cho phép truy xuất từng dòng dữ liệu  
  
public ResultSet executeQuery(String sql) { // dành cho câu lệnh select  
    ResultSet rs = null;  
    try {  
        Statement stm = (Statement) conn.createStatement();  
        rs = stm.executeQuery(sql);  
    } catch (SQLException ex) {  
        displayError(ex);  
    }  
    return rs;  
}
```

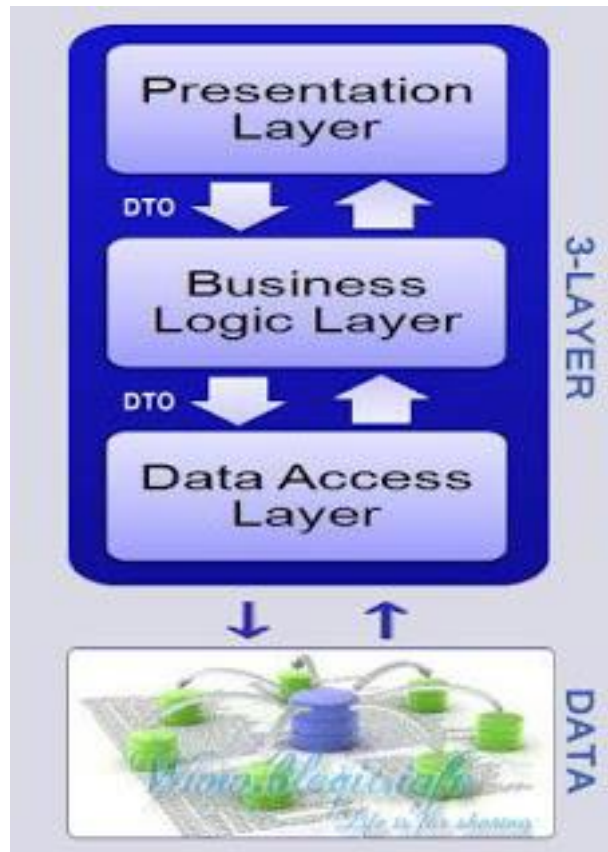
### 2.4. Xử lý các ngoại lệ

```
// xử lý ngoại lệ khi tương tác với csdl  
public void displayError(SQLException ex) {  
    System.out.println("Error Message:" + ex.getMessage());  
    System.out.println("SQL State:" + ex.getSQLState());  
    System.out.println("Error Code:" + ex.getErrorCode());  
}
```

### 2.5. Đóng kết nối csdl

```
public void close(){// dong ket noi co so du lieu
    try {
        if(conn!=null)
            conn.close();
    } catch (SQLException ex) {
        displayError(ex);
    }
}
```

### 3. Mô hình 3 lớp (3-layers)



Chia nhỏ phần mềm thành các tầng, mỗi tầng có vai trò đặc trưng riêng để thuận tiện cho việc dùng lại, sẵn sàng cho việc mở rộng hệ thống, dễ bảo trì, đặc biệt là dễ dàng trong việc phân chia công việc cho các thành viên trong nhóm...bao gồm:

- Presentation Layer (GUI): tầng này để giao tiếp với người dùng để thu thập dữ liệu và hiển thị kết quả thông qua giao diện, tầng này sử dụng các dịch vụ do tầng business cung cấp
- Business Layer (BUS): tầng này để xử lý nghiệp vụ chính của hệ thống sử dụng các dịch vụ do tầng Data cung cấp và cung cấp các dịch vụ cho tầng Presentation
- Datalayer (DAO): tầng này chịu trách nhiệm lưu trữ và xử lý dữ liệu như (insert, update, delete, select ...)

Tất cả các tầng này trao đổi dữ liệu thông qua Data Transfer Object (dto)

### 3.1. DTO

#### Tạo package DTO, tạo employeeDTO.java

```
package dto;
public class employeeDTO {

    private int employee_id;
    private String employee_name;
    private int phonemuber;
    private String address;

    /**
     * @return the employee_id
     */
    public int getEmployee_id() {
        return employee_id;
    }

    /**
     * @param employee_id the employee_id to set
     */
    public void setEmployee_id(int employee_id) {
        this.employee_id = employee_id;
    }

    /**
     * @return the employee_name
     */
    public String getEmployee_name() {
        return employee_name;
    }

    /**
     * @param employee_name the employee_name to set
     */
    public void setEmployee_name(String employee_name) {
        this.employee_name = employee_name;
    }

    /**
     * @return the phonemuber
     */
    public int getPhonemuber() {
        return phonemuber;
    }

    /**
     * @param phonemuber the phonemuber to set
     */
    public void setPhonemuber(int phonemuber) {
        this.phonemuber = phonemuber;
    }

    /**
     * @return the address
     */
    public String getAddress() {
        return address;
    }
}
```

```

/**
 * @param address the address to set
 */
public void setAddress(String address) {
    this.address = address;
}
}

```

### 3.2. DAO

Tạo package DAO

Tạo MysqlDataAccessHelper.java // nhiều project sẽ đặt class này trong package riêng (util)

```

package dao;

import com.mysql.jdbc.Driver;
import com.mysql.jdbc.Statement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;

public class MysqlDataAccessHelper {

    public Connection conn = null;
    // xu ly ngoai le khi tuong tac voi csdl
    public void displayError(SQLException ex){
        System.out.println(" Error Message:" + ex.getMessage());
        System.out.println(" SQL State:" + ex.getSQLState());
        System.out.println(" Error Code:" + ex.getErrorCode());
    }

    public void open(){// mo ket noi den csdl
        try{
            Driver driver = new org.gjt.mm.mysql.Driver();// nap driver
            DriverManager.registerDriver(driver);// dang ky driver

            String url = "jdbc:mysql://localhost:3306/employee";
            conn = DriverManager.getConnection(url, "root", ""); //tao ket noi den co so
            //du lieu

        } catch(SQLException ex){// xu ly ngoai le
            displayError(ex);
        }
    }

    public void close(){// dong ket noi co so du lieu
        try {
            if(conn!=null)
                conn.close();
        } catch (SQLException ex) {
            displayError(ex);
        }
    }

    //tao va thuc thi cac cau lenh sql
    // cung cap thong tin trích rút từ csdl và cho phép truy xuất từng dòng dữ liệu
    public ResultSet excuteQuery(String sql){// danh cho cau lenh secect
        ResultSet rs = null;
        try {

```

```

        Statement stm = (Statement) conn.createStatement();
        rs = stm.executeQuery(sql);
    } catch (SQLException ex) {
        displayError(ex);
    }
    return rs;
}
}

```

## Tạo employeeDAO.java

```

package dao;

import dto.employeeDTO;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
// import util.MysqlDataAccessHelper; // trường hợp có UTIL riêng, phải import vào

public class employeeDAO {

    public static ArrayList<employeeDTO> employeeAll(){

        ArrayList<employeeDTO> arr = new ArrayList<employeeDTO>();
        String sql = "select * from tbl_employee";
        MysqlDataAccessHelper my = new MysqlDataAccessHelper();

        try {
            my.open();
            ResultSet rs = my.excuteQuery(sql);
            while(rs.next()){
                employeeDTO em = new employeeDTO();
                em.setEmployee_id(rs.getInt("employee_id"));
                em.setEmployee_name(rs.getString("employee_name"));
                em.setPhonemuber(rs.getInt("phonenumber"));
                em.setAddress(rs.getString("address"));
                arr.add(em);
            }
        } catch (SQLException ex) {
            my.displayError(ex);
        } finally{
            my.close();
        }
        return arr;
    }
}

```

### 3.3. BUS

Tạo package DTO, tạo employeeDTO.java

```

package bus;

import dao.employeeDAO;
import dto.employeeDTO;
import java.util.ArrayList;

public class employeeBUS {

```

```

        public static ArrayList<employeeDTO> employeeAll(){
            return employeeDAO.employeeAll();
        }
    }
}

```

### 3.4. GUI

Tạo package DTO, tạo employeeDTO.java

```

package gui;

import bus.employeeBUS;
import dto.employeeDTO;
import java.util.ArrayList;
import javax.swing.table.DefaultTableModel;

public class employeeGUI extends javax.swing.JFrame {

    /** Creates new form employee1GUI */
    public employeeGUI() {
        initComponents();
        loadEmployeeAll();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jScrollPane1 = new javax.swing.JScrollPane();
        jTable1 = new javax.swing.JTable();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jTable1.setModel(new javax.swing.table.DefaultTableModel(
            new Object [][] {
                {},
                {},
                {},
                {}
            },
            new String [] {

            }
        ));
        jScrollPane1.setViewportView(jTable1);

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE,
375, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(15, Short.MAX_VALUE))
        );
        layout.setVerticalGroup(

```

```

        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(10, 10, 10)
                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE,
275, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(10, 10, 10)
                .addComponent(jTable1, javax.swing.GroupLayout.PREFERRED_SIZE,
275, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(10, 10, 10)
        );

    pack();
} // </editor-fold>

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new employeeGUI().setVisible(true);
        }
    });
}

// Variables declaration - do not modify
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JTable jTable1;
// End of variables declaration

public void loadEmployeeAll() {
    String[] header = {"STT", "EmployeeName", "PhoneNumber", "Address"};
    DefaultTableModel dtm = new DefaultTableModel(header, 0);
    ArrayList<employeeDTO> arr = new ArrayList<employeeDTO>();
    arr = employeeBUS.employeeAll();
    employeeDTO em = new employeeDTO();
    for(int i = 0; i < arr.size(); i++) {
        em = arr.get(i);
        int id = em.getEmployee_id();
        String name = em.getEmployee_name();
        int phone = em.getPhonemuber();
        String address = em.getAddress();
        Object[] row = {id, name, phone, address};
        dtm.addRow(row);
    }
    jTable1.setModel(dtm);
}
}
}

```

## Main.java

```

package gui;
public class main {

    public static void main(String[] args) {
        employeeGUI frm = new employeeGUI();

        frm.setVisible(true);
    }
}

```