


Servlet and JSP Assessment

Jothilingam-12114

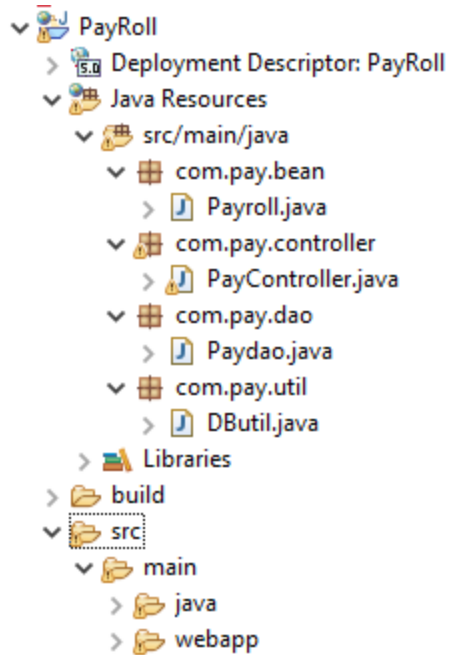
Payroll Management System

Property	Type
Payroll ID	int
Employee Name	String
Emp ID	int
Basic	float
HRA	int
DA	int
TA	int
CTC	Float

Table:

								
	Payrollid	employeenam	empID	basic	hra	da	ta	ctc
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Folder Structure:



Source Code:

Payroll.java

```
package com.pay.bean;

public class Payroll {

    private int payRollID;
    private String employeeName;
    private int empId;
    private float basic;
    private int hra;
    private int da;
    private int ta;
    private float ctc;
    public Payroll(int payRollID, String employeeName, int
emoId, float basic, int hra, int da, int ta, float
ctc) {
        super();
        this.payRollID = payRollID;
        this.employeeName = employeeName;
        this.empId = emoId;
    }
}
```

```
this.basic = basic;
this.hra = hra;
this.da = da;
this.ta = ta;
this.ctc = ctc;
}
public int getPayRollID() {
return payRollID;
}
public void setPayRollID(int payRollID) {
this.payRollID = payRollID;
}
public String getEmployeeName() {
return employeeName;
}
public void setEmployeeName(String employeeName) {
this.employeeName = employeeName;
}
public int getEmpId() {
return empId;
}
public void setEmpId(int empId) {
this.empId = empId;
}
public float getBasic() {
return basic;
}
public void setBasic(float basic) {
this.basic = basic;
}
public int getHra() {
return hra;
}
public void setHra(int hra) {
this.hra = hra;
}
public int getDa() {
return da;
}
```

```

public void setDa(int da) {
    this.da = da;
}
public int getTa() {
    return ta;
}
public void setTa(int ta) {
    this.ta = ta;
}
public float getCtc() {
    return ctc;
}
public void setCtc(float ctc) {
    this.ctc = ctc;
}
}

```

PayController.java

```

package com.pay.controller;

import jakarta.servlet.ServletException;
import jakarta.servlet.annotation.WebServlet;
import jakarta.servlet.http.HttpServlet;
import jakarta.servlet.http.HttpServletRequest;
import jakarta.servlet.http.HttpServletResponse;
import java.io.IOException;
import java.util.List;
import com.pay.bean.Payroll;
import com.pay.dao.Paydao;
public class PayController extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private static Paydao dao = new Paydao();
    protected void doPost(HttpServletRequest request,
        HttpServletResponse response) throws ServletException,
        IOException {
        int n=0;
        String action = request.getParameter("button");
    }
}

```

```

if(action.equals("Insert")) {
if(request.getParameter("payRollID")=="" ||
request.getParameter("employeeName")=="" ||
request.getParameter("empId")==""
|| request.getParameter("basic")=="" ||
request.getParameter("hra")=="" ||
request.getParameter("da")=="" ||
request.getParameter("ta")==")
{
response.sendRedirect("Empty.jsp?action=Insert");

}
else {
Payroll pay = new
Payroll(Integer.parseInt(request.getParameter("payRoll
ID")), request.getParameter("employeeName"),
Integer.parseInt(request.getParameter("empId")),
Float.parseFloat(request.getParameter("basic")),Intege
r.parseInt(request.getParameter("hra")),Integer.parseI
nt(request.getParameter("da")),Integer.parseInt(reques
t.getParameter("ta")),
Float.parseFloat(request.getParameter("ctc")));
n =dao.insert(pay) ;
if(n==1) {
response.sendRedirect("Success.jsp?action=Insert");

}
else {
response.sendRedirect("Failure.jsp?action=Insert");
}
}
}

else if(action.equals("Delete")) {
if(request.getParameter("payRollID")==") {
response.sendRedirect("Empty.jsp?action=Delete");
}
else {
int id =
Integer.parseInt(request.getParameter("payRollID"));
n = dao.delete(id);

```

```

if(n==1) {
response.sendRedirect("Success.jsp?action=Delete");
}else {
response.sendRedirect("Failure.jsp?action=Delete");
}
}
}
else if(action.equals("Delete All")) {

```

```

n = dao.deleteAll();
if(n==0) {
response.sendRedirect("Success.jsp?action=Delete");
}else {
response.sendRedirect("Failure.jsp?action=Delete");

}
}

```

```

else if(action.equals("Update")) {
if(request.getParameter("payRollID")=="" ||
request.getParameter("employeeName")=="" ||
request.getParameter("empId")==""
|| request.getParameter("basic")=="" ||
request.getParameter("hra")=="" ||
request.getParameter("da")=="" ||
request.getParameter("ta")==")
{
response.sendRedirect("Empty.jsp?action=Update");

}
else {
Payroll pay = new
Payroll(Integer.parseInt(request.getParameter("payRoll
ID")), request.getParameter("employeeName"),
Integer.parseInt(request.getParameter("empId")),
Float.parseFloat(request.getParameter("basic")),Integer.parseInt(request.getParameter("hra")),Integer.parseInt(request.getParameter("da")),Integer.parseInt(request.getParameter("ta")));
}
}
}

```

```

nt(request.getParameter("da")), Integer.parseInt(request
t.getParameter("ta")),
Float.parseFloat(request.getParameter("ctc"))));
n =dao.update(pay);
if(n==1) {
response.sendRedirect("Success.jsp?action=Update");

}else {
response.sendRedirect("Failure.jsp?action=Update");
}
}
}
else if(action.equals("Search")) {
if(request.getParameter("payRollID")== "") {
response.sendRedirect("Empty.jsp?action=Search");
}else {
int id =
Integer.parseInt(request.getParameter("payRollID"));
Payroll emp = dao.search(id);
if(emp==null) {
response.sendRedirect("Failure.jsp?action=Search");
}else {
request.getSession().setAttribute("bean", emp);
response.sendRedirect("Success.jsp?action=Search");
}
}
}
else if(action.equals("View All")) {
List<Payroll> list = dao.viewAll();
if(list.isEmpty()) {
response.sendRedirect("Failure.jsp?action=View");
}else {
request.getSession().setAttribute("list", list);
response.sendRedirect("Success.jsp?action=View");
}
}
}
}
}
}

```

Paydao.java

```
package com.pay.dao;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;
import com.pay.bean.Payroll;
import com.pay.util.DBUtil;

public class Paydao {
    int n=0;
    private List<Payroll> list ;
    private Connection con = DBUtil.getDBConnection();

    public int insert(Payroll pay) {

        try {
            String q = "insert into payroll.payroll_tbl
            values(?,?,?,?,?,?,?,?)";
            PreparedStatement ps = con.prepareStatement(q);
            ps.setInt(1, pay.getPayRollID());
            ps.setString(2, pay.getEmployeeName());
            ps.setInt(3, pay.getEmpId());
            ps.setFloat(4, pay.getBasic());
            ps.setInt(5, pay.getHra());
            ps.setInt(6, pay.getDa());
            ps.setInt(7, pay.getTa());
            ps.setFloat(8, pay.getCtc());

            n=ps.executeUpdate();
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return n;
    }
}
```



```

public int delete(int id) {

    try {
        String q = "delete from payroll.payroll_tbl where Payrollid = ?";
        PreparedStatement ps = con.prepareStatement(q);
        ps.setInt(1, id);
        n=ps.executeUpdate();
    } catch (SQLException e) {
        e.printStackTrace();
    }

    return n;
}

public int deleteAll() {

    try {
        String q = "TRUNCATE `payroll`.`payroll_tbl`";
        PreparedStatement ps = con.prepareStatement(q);

        n=ps.executeUpdate();
    } catch (SQLException e) {
        e.printStackTrace();
    }

    return n;
}

public int update(Payroll pay) {

    try {
        String q="UPDATE payroll.payroll_tbl SET employeename
        = ?, empID = ?, basic = ?, hra = ?, da = ?, ta = ?,
        ctc = ? WHERE Payrollid = ?";

        PreparedStatement ps = con.prepareStatement(q);

        ps.setString(1, pay.getEmployeeName());
        ps.setInt(2, pay.getEmpId());
    }
}

```

```

ps.setFloat(3, pay.getBasic());
ps.setInt(4, pay.getHra());
ps.setInt(5, pay.getDa());
ps.setInt(6, pay.getTa());
ps.setFloat(7, pay.getCtc());
ps.setInt(8, pay.getPayRollID());
n=ps.executeUpdate();
} catch (SQLException e) {

e.printStackTrace();
}
return n;
}

public Payroll search(int id) {
ResultSet rs;

try {
String q = "select * from payroll.payroll_tbl where
Payrollid = ?";
PreparedStatement ps = con.prepareStatement(q);
ps.setInt(1, id);
rs = ps.executeQuery();
if(rs.next())
return new
Payroll(rs.getInt(1),rs.getString(2),rs.getInt(3),rs.g
etFloat(4),rs.getInt(5),rs.getInt(6),rs.getInt(7),rs.g
etFloat(8));
} catch (SQLException e) {
e.printStackTrace();
}
return null;
}

public List<Payroll> viewAll(){
ResultSet rs;

try {
list = new ArrayList<Payroll>();
String q = "select * from payroll.payroll_tbl";
PreparedStatement ps = con.prepareStatement(q);

```

```

rs=ps.executeQuery();
while(rs.next())
list.add(new
Payroll(rs.getInt(1),rs.getString(2),rs.getInt(3),rs.g
etFloat(4),rs.getInt(5),rs.getInt(6),rs.getInt(7),rs.g
etFloat(8)));
} catch (SQLException e) {
e.printStackTrace();
}
return list;
}
}

```

DBUtil.java

```

package com.pay.util;

import java.sql.Connection;
import java.sql.DriverManager;

public class DBUtil {
public static Connection getDBConnection() {
    Connection con=null;
    final String
URL="jdbc:mysql://localhost:3306/payroll";
    final String User="root";
    final String pass="root";
    try
    {
        con=DriverManager.getConnection(URL,User
,pass);
    } catch (Exception e) {
        System.out.println(e);
    }
    return con;
}
}

```

Screen Shots:

Insert Details:

Payroll Management System

Insert

Delete

Update

Search

Enter The Details

Enter Payroll Id

Enter Employee Name

Enter Employee ID

Enter Basic Salary

Enter HRA

Enter DA

Enter TA

Enter CTC

+ Add Details

Record Inserted Successfully

InsertDB:

	Payrollid	employeename	empID	basic	hra	da	ta	ctc
▶	101	Kirubakaran	1005	50500	1200	1100	1750	56000
	102	Jothilingam	1001	25000	4545	963	3516	40000

Update:

Payroll Management System

Insert

Delete

Update

Search

Update Details

Enter Payroll Id

Enter Employee Name

Enter Employee ID

Enter Basic Salary

Enter HRA

Enter DA

Enter TA

Enter CTC

Update

Record Updateded Successfully

Update DB:

Result Grid								
		Filter Rows:		Edit:		Export/Import:		
	Payrollid	employeename	empID	basic	hra	da	ta	ctc
▶	101	Kirubakaran	1005	50500	1200	1100	1750	56000
	102	Jothilingam	1001	25000	4545	963	3516	40000
✱	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Search:

Payroll Management System

Search Details

Payroll ID	Employee Name	Employee ID	Basic	HRA	DA	TA	CTC
101	Kirubakaran	1005	50500.0	1200	1100	1750	56000.0

Search ALL:

Payroll Management System

Search Details

No Of Record : 3

Payroll ID	Employee Name	Employee ID	Basic	HRA	DA	TA	CTC
101	Kirubakaran	1005	50500.0	1200	1100	1750	56000.0
102	Jothilingam	1001	25000.0	4545	963	3516	40000.0
103	Bishek	1009	60000.0	520	900	2000	80000.0

Delete:

Payroll Management System

Insert

Delete

Update

Search

Delete Detail

Enter Pay Roll Id

Delete

Delete All Details

Record Deleted Successfully

Delete All:

Payroll Management System

Insert

Delete

Update

Search

Delete Detail

Enter Pay Roll Id

Delete

Delete All Details

Record Deleted Successfully