**Spring CRUD operation using Spring JDBC**

Employee.java

**package** com.model;

**public** **class** Employee {

**private** Integer id;

**private** String username;

**private** String password;

**public** Employee() {

}

**public** Employee(Integer id, String username, String password) {

**this**(username,password);

**this**.id=id;

}

**public** Employee(String username, String password) {

**this**.username=username;

**this**.password=password;

}

**public** Integer getId() {

**return** id;

}

**public** **void** setId(Integer id) {

**this**.id = id;

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

**public** String getPassword() {

**return** password;

}

**public** **void** setPassword(String password) {

**this**.password = password;

}

@Override

**public** String toString() {

**return** "Employee [id=" + id + ", username=" + username + ", password=" + password + "]";

}

}

EmployeeController.java

package com.controller;

import java.util.\*;

import javax.servlet.http.HttpServletRequest;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.ModelAttribute;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.bind.annotation.RequestParam;

import org.springframework.web.servlet.ModelAndView;

import com.dao.EmployeeInterface;

import com.model.Employee;

@Controller

public class EmployeeController {

@Autowired

private EmployeeInterface employeeInterface;

@RequestMapping(value = "/")

public ModelAndView listEmployee(ModelAndView model) {

List<Employee> listEmployee = employeeInterface.getList();

model.addObject("listEmployee", listEmployee);

model.setViewName("index");//

return model;

}

@RequestMapping(value = "/new", method = RequestMethod.GET)

public ModelAndView newEmployee(ModelAndView model) {

Employee newEmployee = new Employee();

model.addObject("employee", newEmployee);

model.setViewName("employee\_form");//

return model;

}

@RequestMapping(value = "/save", method = RequestMethod.POST)

public ModelAndView saveEmployee(@ModelAttribute Employee employee) {

if (employee.getId() == null) {

employeeInterface.save(employee); //to save the new records

} else {

employeeInterface.update(employee); //to update

}

return new ModelAndView("redirect:/");

}

@RequestMapping(value = "/edit", method = RequestMethod.GET)

public ModelAndView editEmployee(HttpServletRequest request) {

Integer id = Integer.parseInt(request.getParameter("id")); //to get the id

Employee employee = employeeInterface.get(id);

ModelAndView model = new ModelAndView("employee\_form");

model.addObject("employee", employee);

return model;

}

@RequestMapping(value = "/delete", method = RequestMethod.GET)

public ModelAndView deleteEmployee(@RequestParam Integer id) {

employeeInterface.delete(id);

return new ModelAndView("redirect:/");

}

}

EmployeeImpl.java

package com.dao;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.List;

import javax.sql.DataSource;

import org.springframework.dao.DataAccessException;

import org.springframework.jdbc.core.JdbcTemplate;

import org.springframework.jdbc.core.ResultSetExtractor;

import org.springframework.jdbc.core.RowMapper;

import org.springframework.stereotype.Repository;

import com.model.Employee;

@Repository

public class EmployeeImpl implements EmployeeInterface {

private JdbcTemplate jdbcTemplate;

public EmployeeImpl(DataSource dataSource) {

this.jdbcTemplate = new JdbcTemplate(dataSource);

}

@Override

public int save(Employee employee) {

String sql="insert into employee(username,password)values(?,?)";

return jdbcTemplate.update(sql,employee.getUsername(),employee.getPassword());

}

@Override

public int update(Employee employee) {

String sql="update employee set username=?,password=? where id=?";

return jdbcTemplate.update(sql,employee.getUsername(),employee.getPassword(),employee.getId());

}

@Override

public Employee get(Integer id) {

String sql="select \* from employee where id="+id;

ResultSetExtractor<Employee> resultSetExtractor= new ResultSetExtractor<Employee>() {

@Override

public Employee extractData(ResultSet rs) throws SQLException, DataAccessException {

if(rs.next()) {

String username=rs.getString("username");

String password=rs.getString("password");

return new Employee(id, username, password);

}

return null;

}

};

return jdbcTemplate.query(sql, resultSetExtractor);

}

@Override

public int delete(Integer id) {

String sql="delete from employee where id="+id;

return jdbcTemplate.update(sql);

}

@Override

public List<Employee> getList() {

String sql="select \* from employee";

RowMapper<Employee> rowMapper= new RowMapper<Employee>(){

@Override

public Employee mapRow(ResultSet rs, int rowNum) throws SQLException {

Integer id=rs.getInt("id");

String username=rs.getString("username");

String password=rs.getString("password");

return new Employee(id, username, password);

}

};

return jdbcTemplate.query(sql, rowMapper);

}

}

EmployeeInterface.java

package com.dao;

import java.util.List;

import com.model.Employee;

public interface EmployeeInterface { //crud

public int save(Employee employee);

public int update(Employee employee);

public Employee get(Integer id);

public int delete(Integer id);

public List<Employee> getList();

}

SpringMvcConfig.java

package com.test;

import javax.sql.DataSource;

import org.springframework.context.annotation.Bean;

import org.springframework.context.annotation.ComponentScan;

import org.springframework.context.annotation.Configuration;

import org.springframework.jdbc.datasource.DriverManagerDataSource;

import org.springframework.web.servlet.ViewResolver;

import org.springframework.web.servlet.config.annotation.EnableWebMvc;

import org.springframework.web.servlet.config.annotation.WebMvcConfigurer;

import org.springframework.web.servlet.view.InternalResourceViewResolver;

import com.dao.EmployeeImpl;

import com.dao.EmployeeInterface;

@Configuration

@EnableWebMvc

@ComponentScan(basePackages = "com")

public class SpringMvcConfig implements WebMvcConfigurer {

@Bean

public DataSource getDataSource() {

//To get the database connection object

DriverManagerDataSource dataSource = new DriverManagerDataSource();

dataSource.setDriverClassName("com.mysql.jdbc.Driver");

dataSource.setUrl("jdbc:mysql://localhost:3306/test");

dataSource.setUsername("root");

dataSource.setPassword("root");

return dataSource;

}

@Bean

public ViewResolver getViewResolver() {

InternalResourceViewResolver resolver= new InternalResourceViewResolver();

resolver.setPrefix("/WEB-INF/views/");

resolver.setSuffix(".jsp");

return resolver;

}

public EmployeeInterface getEmployeeInterfaceDao() {

return new EmployeeImpl(getDataSource());

}

}

WebAppInitalizer.java

package com.test;

import javax.servlet.ServletContext;

import javax.servlet.ServletException;

import javax.servlet.ServletRegistration;

import org.springframework.web.WebApplicationInitializer;

import org.springframework.web.context.support.AnnotationConfigWebApplicationContext;

import org.springframework.web.servlet.DispatcherServlet;

public class WebAppInitializer implements WebApplicationInitializer {

@Override

public void onStartup(ServletContext servletContext) throws ServletException {

// TODO Auto-generated method stub

AnnotationConfigWebApplicationContext appContext = new AnnotationConfigWebApplicationContext();

appContext.register(SpringMvcConfig.class);

ServletRegistration.Dynamic dispatcher = servletContext.addServlet("SpringDispatcher",

new DispatcherServlet(appContext));

dispatcher.setLoadOnStartup(1);

dispatcher.addMapping("/");

}

}

Pom.xml

<project xmlns=*"http://maven.apache.org/POM/4.0.0"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"*>

<modelVersion>4.0.0</modelVersion>

<groupId>Spring</groupId>

<artifactId>SpringCRUD</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<build>

<sourceDirectory>src</sourceDirectory>

<plugins>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.7.0</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

<plugin>

<artifactId>maven-war-plugin</artifactId>

<version>3.2.1</version>

<configuration>

<warSourceDirectory>WebContent</warSourceDirectory>

</configuration>

</plugin>

</plugins>

</build>

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.1.9.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.1.9.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-orm</artifactId>

<version>5.1.9.RELEASE</version>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>4.0.1</version>

</dependency>

<dependency>

<groupId>javax.servlet.jsp</groupId>

<artifactId>javax.servlet.jsp-api</artifactId>

<version>2.3.3</version>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>jstl</artifactId>

<version>1.2</version>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.22</version>

</dependency>

</dependencies>

</project>

index.jsp

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<h1>Employee List>></h1>

<h3>

<a href=*"new"*>New Employee</a>

</h3>

<table border=*"1"* cellpadding=*"5"*>

<tr>

<th>ID</th>

<th>Username</th>

<th>Password</th>

<th>Action</th>

</tr>

<c:forEach items=*"*${listEmployee}*"* var=*"employee"* varStatus=*"status"*>

<tr>

<td>${employee.id}</td>

<td>${employee.username}</td>

<td>${employee.password}</td>

<td>

<a href=*"edit?id=*${employee.id}*"*>Edit</a>

&nbsp;&nbsp;

<a href=*"delete?id=*${employee.id}*"*>Delete</a>

</td>

</tr>

</c:forEach>

</table>

</div>

</body>

</html>

employee\_form.jsp

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ taglib uri=*"http://www.springframework.org/tags/form"* prefix=*"form"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Add/Edit Employee</title>

</head>

<body>

<div align=*"center"*>

<h1>New/Edit Employee</h1>

<!-- -spring mvc form tag attributes -->

<form:form action=*"save"* method=*"post"* modelAttribute=*"employee"*>

<table cellpadding=*"5"*>

<form:hidden path=*"id"*/>

<tr>

<td>User Name</td>

<td><form:input path=*"username"* /></td>

</tr>

<tr>

<td>Password</td>

<td><form:input path=*"password"* /></td>

</tr>

<tr>

<td colspan=*"2"* align=*"center"*><input type=*"submit"*

value=*"Save"*></td>

</tr>

</table>

</form:form>

</div>

</body>

</html>

SQL query

CREATE TABLE `employee` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

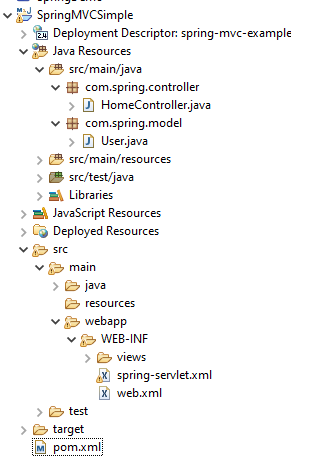
`username` varchar(255) DEFAULT NULL,

`password` varchar(255) DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=16 DEFAULT CHARSET=latin1;

Example-2



User.java

**package** com.spring.model;

**public** **class** User {

**private** String userName;

**public** String getUserName() {

**return** userName;

}

**public** **void** setUserName(String userName) {

**this**.userName = userName;

}

}

Pom.xml

<project xmlns=*"http://maven.apache.org/POM/4.0.0"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"*>

<modelVersion>4.0.0</modelVersion>

<groupId>com.journaldev.spring.mvc</groupId>

<artifactId>spring-mvc-example</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>Spring MVC Example</name>

<description>Spring MVC Hello World Example</description>

<!-- Add Spring Web and MVC dependencies -->

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>4.3.9.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-web</artifactId>

<version>4.3.9.RELEASE</version>

</dependency>

<!-- Servlet -->

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>servlet-api</artifactId>

<version>2.5</version>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>javax.servlet.jsp</groupId>

<artifactId>jsp-api</artifactId>

<version>2.1</version>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>jstl</artifactId>

<version>1.2</version>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.6.1</version>

<configuration>

<source>1.8</source>

<target>1.8</target>

</configuration>

</plugin>

<plugin>

<artifactId>maven-war-plugin</artifactId>

<version>3.0.0</version>

<configuration>

<warSourceDirectory>WebContent</warSourceDirectory>

</configuration>

</plugin>

</plugins>

<finalName>${project.artifactId}</finalName> <!-- added to remove Version from WAR file -->

</build>

</project>

Web.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://java.sun.com/xml/ns/javaee"* xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"* id=*"WebApp\_ID"* version=*"3.0"*>

<display-name>spring-mvc-example</display-name>

<!-- Add Spring MVC DispatcherServlet as front controller -->

<servlet>

<servlet-name>spring</servlet-name>

<servlet-class>

org.springframework.web.servlet.DispatcherServlet

</servlet-class>

<init-param>

<param-name>contextConfigLocation</param-name>

<param-value>/WEB-INF/spring-servlet.xml</param-value>

</init-param>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>spring</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

</web-app>

Spring-servlet.xml

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<beans:beans xmlns=*"http://www.springframework.org/schema/mvc"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns:beans=*"http://www.springframework.org/schema/beans"*

xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc/spring-mvc.xsd*

*http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd*

*http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context.xsd"*>

<!-- DispatcherServlet Context: defines this servlet's request-processing

infrastructure -->

<!-- Enables the Spring MVC @Controller programming model -->

<annotation-driven />

<context:component-scan base-package=*"com.spring"* />

<!-- Resolves views selected for rendering by @Controllers to .jsp resources

in the /WEB-INF/views directory -->

<beans:bean

class=*"org.springframework.web.servlet.view.InternalResourceViewResolver"*>

<beans:property name=*"prefix"* value=*"/WEB-INF/views/"* />

<beans:property name=*"suffix"* value=*".jsp"* />

</beans:bean>

</beans:beans>

home.jsp

<%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<%@ page session=*"false"*%>

<html>

<head>

<title>Home</title>

</head>

<body>

<h1>Hello world!</h1>

<form action=*"user"* method=*"post"*>

<input type=*"text"* name=*"userName"*><br> <input

type=*"submit"* value=*"Login"*>

</form>

</body>

</html>

User.jsp

<%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*

pageEncoding=*"UTF-8"*%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<title>User Home Page</title>

</head>

<body>

<h3>Hi ${userName}</h3>

</body>

</html>

HomeController.java

package com.spring.controller;

import java.text.DateFormat;

import java.util.Date;

import java.util.Locale;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.validation.annotation.Validated;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import com.spring.model.User;

@Controller

public class HomeController {

/\*\*

\* Simply selects the home view to render by returning its name.

\*/

@RequestMapping(value = "/", method = RequestMethod.GET)

public String home() {

return "home"; or **return** **new** ModelAndView("home");

}

@RequestMapping(value = "/user", method = RequestMethod.POST)

public String user(@Validated User user, Model model) {

System.out.println("User Page Requested");

model.addAttribute("userName", user.getUserName());

return "user";

}

Output-