作业号：JW\_28 得分：

题目要求(作业题干)

|  |
| --- |
| JW\_28 令DepartmentController提供按学院查询的功能  1. DepartmentDao中定义方法findAllBySchool(Integer schoolId)，返回schoolId指定的系  2. DepartmentService中定义方法findAllBySchool(Integer schoolId)，调用Dao中同名方法  3. 在DepartmentController中doGet方法中  3.1 令department.ctl接收两个参数：  3.1.1 department.ctl?paraType=school&id=1表示要求服务器响应SchoolId=1的所有系  3.1.2 department.ctl?id=1表示要求服务器响应Id=1的所有系 （功能同前）  3.1.3 department.ctl表示要求服务器响应所有系 （功能同前）  提交时间：11月25日23：00之前 |

源程序(打印版本不得含有注释，不能修改字体字号)

**Filter0\_Encoding类**

|  |
| --- |
| 01 package filter;  02  03 import javax.servlet.\*;  04 import javax.servlet.annotation.WebFilter;  05 import javax.servlet.http.HttpServletRequest;  06 import javax.servlet.http.HttpServletResponse;  07 import java.io.IOException;  08  09 @WebFilter(filterName = "Filter 0", urlPatterns = {"/\*"})  10  11 public class Filter0\_Encoding implements Filter {  12 @Override  13 public void init(FilterConfig filterConfig) throws ServletException {}  14  15 @Override  16 public void destroy() {}  17  18 @Override  19 public void doFilter(ServletRequest servletRequest, ServletResponse servletResponse,  20 FilterChain filterChain) throws IOException, ServletException {  21 System.out.println("Filter 0 - encoding begins");  22 HttpServletRequest request = (HttpServletRequest)servletRequest;  23 HttpServletResponse response=(HttpServletResponse)servletResponse;  24 String path= request.getRequestURI();  25 String method=request.getMethod();  26 if (path.contains("/login")){  27 System.out.println("未设置字符编码格式");  28 }else {  29 System.out.println(method);  30 response.setContentType("text/html;charset=UTF-8");  31  32 if (("POST-PUT").contains(method)){  33 request.setCharacterEncoding("UTF-8");  34  35 }  36 }  37 filterChain.doFilter(servletRequest,servletResponse);  38 System.out.println("Filter 0 - encoding ends"+"\n");  39 }  40 } |

**Filter1\_Date类**

|  |  |
| --- | --- |
| 01 package filter;  02  03 import javax.servlet.\*;  04 import javax.servlet.annotation.WebFilter;  05 import javax.servlet.http.HttpServletRequest;  06 import java.io.IOException;  07 import java.util.Calendar;  08  09 @WebFilter(filterName = "Filter 1", urlPatterns = {"/\*"})  10 public class Filter1\_Date implements Filter {  11 @Override  12 public void init(FilterConfig filterConfig) throws ServletException {}  13  14 @Override  15 public void destroy() {}  16  17 @Override  18 public void doFilter(ServletRequest servletRequest, ServletResponse servletResponse, FilterChain filterChain) throws IOException, ServletException {  19 System.out.println("Filter 1 - date begins");  20 HttpServletRequest request=(HttpServletRequest)servletRequest;  21 String path=request.getRequestURI();  22 Calendar cal = Calendar.getInstance();  23 String time= cal.get(Calendar.YEAR)+"年"+(cal.get(Calendar.MONTH) + 1)+"月"+cal.get(Calendar.DATE)+"日"+cal.get(Calendar.HOUR\_OF\_DAY)+": "+cal.get(Calendar.MINUTE);  24 System.out.println(path+" @ "+time);  25 filterChain.doFilter(servletRequest,servletResponse);  26 System.out.println("Filter 1 - date ends");  27 }  28 } |  |

**DepartmentDao类**

|  |
| --- |
| 001 package controller.department;  002  003 import com.alibaba.fastjson.JSON;  004 import com.alibaba.fastjson.JSONObject;  005 import com.alibaba.fastjson.serializer.SerializerFeature;  006 import domain.Department;  007 import service.DepartmentService;  008 import util.JSONUtil;  009  010 import javax.servlet.ServletException;  011 import javax.servlet.annotation.WebServlet;  012 import javax.servlet.http.HttpServlet;  013 import javax.servlet.http.HttpServletRequest;  014 import javax.servlet.http.HttpServletResponse;  015 import java.io.IOException;  016 import java.sql.SQLException;  017 import java.util.Collection;  018  019  020 @WebServlet("/department.ctl")  021 public class DepartmentController extends HttpServlet {  022 //请使用以下JSON测试增加功能  023 //{"description":"造价","no":"07","remarks":"","school":{"id":1}}  024 //请使用以下JSON测试修改功能  025 //{"description":"应用化学","id":6,"no":"05","remarks":"","school":{"id":2}}  026  027 /\*\*  028 \* POST, http://49.234.109.117:8080/bysj/department.ctl, 增加专业  029 \*/  030 @Override  031 protected void doPost(HttpServletRequest request, HttpServletResponse response)  032 throws ServletException, IOException {  033 String department\_json = JSONUtil.getJSON(request);  034  035 Department departmentToAdd = JSON.parseObject(department\_json, Department.class);  036  037 JSONObject message = new JSONObject();  038 try {  039 DepartmentService.getInstance().add(departmentToAdd);  040 message.put("message", "增加成功");  041 }catch (SQLException e){  042 message.put("message", "数据库操作异常");  043 e.printStackTrace();  044 }catch(Exception e){  045 message.put("message", "网络异常");  046 e.printStackTrace();  047 }  048 response.getWriter().println(message);  049 }  050  051 /\*\*  052 \* DELETE, http://49.234.109.117:8080/bysj/department.ctl?id=1, 删除id=1的专业  053 \*/  054 @Override  055 protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  056 String id\_str = request.getParameter("id");  057 int id = Integer.parseInt(id\_str);  058  059 JSONObject message = new JSONObject();  060  061 try {  062 DepartmentService.getInstance().delete(id);  063 message.put("message", "删除成功");  064 }catch (SQLException e){  065 message.put("message", "数据库操作异常");  066 e.printStackTrace();  067 }catch(Exception e){  068 message.put("message", "网络异常");  069 e.printStackTrace();  070 }  071 response.getWriter().println(message);  072 }  073  074  075 /\*\*  076 \* PUT, http://49.234.109.117:8080/bysj/department.ctl, 修改专业  077 \*/  078 @Override  079 protected void doPut(HttpServletRequest request, HttpServletResponse response)  080 throws ServletException, IOException {  081 String department\_json = JSONUtil.getJSON(request);  082 Department departmentToAdd = JSON.parseObject(department\_json, Department.class);  083  084 JSONObject message = new JSONObject();  085 try {  086 DepartmentService.getInstance().update(departmentToAdd);  087 message.put("message", "修改成功");  088 }catch (SQLException e){  089 message.put("message", "数据库操作异常");  090 e.printStackTrace();  091 }catch(Exception e){  092 message.put("message", "网络异常");  093 e.printStackTrace();  094 }  095 response.getWriter().println(message);  096 }  097  098 /\*\*  099 \* GET, http://49.234.109.117:8080/bysj/department.ctl?id=1, 查询id=1的专业  100 \* GET, http://49.234.109.117:8080/bysj/department.ctl, 查询所有的专业  101 \* GET, http://49.234.109.117:8080/bysj/department.ctl?id=1&paraType=school  102 \* 把一个或所有专业对象响应到前端  103 \*/  104 @Override  105 protected void doGet(HttpServletRequest request, HttpServletResponse response)  106 throws ServletException, IOException {  107 String id\_str = request.getParameter("id");  108 String paraType=request.getParameter("paraType");  109 JSONObject message = new JSONObject();  110 try {  111 if (id\_str == null&&paraType==null) {  112 responseDepartments(response);  113 } else if (paraType==null){  114 int id = Integer.parseInt(id\_str);  115 responseDepartment(id, response);  116 }else if (paraType.equals("school")&&id\_str!=null){  117 int id = Integer.parseInt(id\_str);  118 responseDepartmentsOfSchool(id,response);  119 }else{  120 response.getWriter().println("参数设置错误");  121 }  122 }catch (SQLException e){  123 message.put("message", "数据库操作异常");  124 e.printStackTrace();  125 response.getWriter().println(message);  126 }catch(Exception e){  127 message.put("message", "网络异常");  128 e.printStackTrace();  129 response.getWriter().println(message);  130 }  131 }  132 private void responseDepartment(int id, HttpServletResponse response)  133 throws ServletException, IOException, SQLException {  134 Department department = DepartmentService.getInstance().find(id);  135 String department\_json = JSON.toJSONString(department);  136  137 response.getWriter().println(department\_json);  138 }  139 private void responseDepartments(HttpServletResponse response)  140 throws ServletException, IOException, SQLException {  141 Collection<Department> departments = DepartmentService.getInstance().findAll();  142 String departments\_json = JSON.toJSONString(departments, SerializerFeature.DisableCircularReferenceDetect);  143  144 response.getWriter().println(departments\_json);  145 }  146 private void responseDepartmentsOfSchool(int id,HttpServletResponse response)  147 throws ServletException, IOException, SQLException {  148 Collection<Department> departments = DepartmentService.getInstance().findAllBySchool(id);  149 String departments\_json = JSON.toJSONString(departments, SerializerFeature.DisableCircularReferenceDetect);  150  151 response.getWriter().println(departments\_json);  152 }  153 } |

**DepartmentService类**

|  |
| --- |
| 01 package service;  02  03  04 import dao.DepartmentDao;  05 import domain.Department;  06  07 import java.sql.SQLException;  08 import java.util.Set;  09  10 public final class DepartmentService {  11 private static DepartmentDao departmentDao= DepartmentDao.getInstance();  12 private static DepartmentService departmentService=new DepartmentService();  13 private DepartmentService(){}  14  15 public static DepartmentService getInstance(){  16 return departmentService;  17 }  18  19 public Set<Department> findAll() throws SQLException {  20 return departmentDao.findAll();  21 }  22  23 public Department find(Integer id) throws SQLException {  24 return departmentDao.find(id);  25 }  26  27 public Set<Department> findAllBySchool(Integer schoolId) throws SQLException {  28 return departmentDao.findAllBySchool(schoolId);  29 }  30  31 public boolean update(Department department) throws SQLException {  32 return departmentDao.update(department);  33 }  34  35 public boolean add(Department department) throws SQLException {  36 return departmentDao.add(department);  37 }  38  39 public boolean delete(Integer id) throws SQLException {  40 return departmentDao.delete(id);  41 }  42  43 } |

**DepartmentController类**

|  |
| --- |
| 001 package controller.department;  002  003 import com.alibaba.fastjson.JSON;  004 import com.alibaba.fastjson.JSONObject;  005 import com.alibaba.fastjson.serializer.SerializerFeature;  006 import domain.Department;  007 import service.DepartmentService;  008 import util.JSONUtil;  009  010 import javax.servlet.ServletException;  011 import javax.servlet.annotation.WebServlet;  012 import javax.servlet.http.HttpServlet;  013 import javax.servlet.http.HttpServletRequest;  014 import javax.servlet.http.HttpServletResponse;  015 import java.io.IOException;  016 import java.sql.SQLException;  017 import java.util.Collection;  018  019 @WebServlet("/department.ctl")  020 public class DepartmentController extends HttpServlet {  021 //请使用以下JSON测试增加功能  022 //{"description":"造价","no":"07","remarks":"","school":{"id":1}}  023 //请使用以下JSON测试修改功能  024 //{"description":"应用化学","id":6,"no":"05","remarks":"","school":{"id":2}}  025  026 /\*\*  027 \* POST, http://49.234.109.117:8080/bysj/department.ctl, 增加专业  028 \*/  029 @Override  030 protected void doPost(HttpServletRequest request, HttpServletResponse response)  031 throws ServletException, IOException {  032  033 String department\_json = JSONUtil.getJSON(request);  034  035 Department departmentToAdd = JSON.parseObject(department\_json, Department.class);  036  037  038 JSONObject message = new JSONObject();  039 try {  040 DepartmentService.getInstance().add(departmentToAdd);  041 message.put("message", "增加成功");  042 }catch (SQLException e){  043 message.put("message", "数据库操作异常");  044 e.printStackTrace();  045 }catch(Exception e){  046 message.put("message", "网络异常");  047 e.printStackTrace();  048 }  049 response.getWriter().println(message);  050 }  051  052 /\*\*  053 \* DELETE, http://49.234.109.117:8080/bysj/department.ctl?id=1, 删除id=1的专业  054 \*/  055 @Override  056 protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  057 String id\_str = request.getParameter("id");  058 int id = Integer.parseInt(id\_str);  059  060  061 JSONObject message = new JSONObject();  062  063 try {  064 DepartmentService.getInstance().delete(id);  065 message.put("message", "删除成功");  066 }catch (SQLException e){  067 message.put("message", "数据库操作异常");  068 e.printStackTrace();  069 }catch(Exception e){  070 message.put("message", "网络异常");  071 e.printStackTrace();  072 }  073 response.getWriter().println(message);  074 }  075  076  077 /\*\*  078 \* PUT, http://49.234.109.117:8080/bysj/department.ctl, 修改专业  079 \*/  080 @Override  081 protected void doPut(HttpServletRequest request, HttpServletResponse response)  082 throws ServletException, IOException {  083  084 String department\_json = JSONUtil.getJSON(request);  085 Department departmentToAdd = JSON.parseObject(department\_json, Department.class);  086  087  088 JSONObject message = new JSONObject();  089 try {  090 DepartmentService.getInstance().update(departmentToAdd);  091 message.put("message", "修改成功");  092 }catch (SQLException e){  093 message.put("message", "数据库操作异常");  094 e.printStackTrace();  095 }catch(Exception e){  096 message.put("message", "网络异常");  097 e.printStackTrace();  098 }  099 response.getWriter().println(message);  100 }  101  102 /\*\*  103 \* GET, http://49.234.109.117:8080/bysj/department.ctl?id=1, 查询id=1的专业  104 \* GET, http://49.234.109.117:8080/bysj/department.ctl, 查询所有的专业  105 \*/  106 @Override  107 protected void doGet(HttpServletRequest request, HttpServletResponse response)  108 throws ServletException, IOException {  109  110 String id\_str = request.getParameter("id");  111  112 JSONObject message = new JSONObject();  113 try {  114 if (id\_str == null) {  115 responseDepartments(response);  116 } else {  117 int id = Integer.parseInt(id\_str);  118 responseDepartment(id, response);  119 }  120 }catch (SQLException e){  121 message.put("message", "数据库操作异常");  122 e.printStackTrace();  123 response.getWriter().println(message);  124 }catch(Exception e){  125 message.put("message", "网络异常");  126 e.printStackTrace();  127 response.getWriter().println(message);  128 }  129 }  130 private void responseDepartment(int id, HttpServletResponse response)  131 throws ServletException, IOException, SQLException {  132 Department department = DepartmentService.getInstance().find(id);  133 String department\_json = JSON.toJSONString(department);  134  135 response.getWriter().println(department\_json);  136 }  137 private void responseDepartments(HttpServletResponse response)  138 throws ServletException, IOException, SQLException {  139 Collection<Department> departments = DepartmentService.getInstance().findAll();  140 String departments\_json = JSON.toJSONString(departments, SerializerFeature.DisableCircularReferenceDetect);  141 response.getWriter().println(departments\_json);  142 }  143 } |

运行结果

|  |
| --- |
|  |