作业号：JW\_27 得分： of 100

题目要求(作业题干)

|  |
| --- |
| JW\_27 过滤器  1. 字符编码过滤器  1.1 urlPatterns设置为：适用于任何资源  1.2 login目录下的资源，不适用于该过滤器  （提示：如果uri中含有"/login"，则不设置字符编码）  1.3 如果是增加和修改功能，则设置请求和响应字符编码为UTF-8。如果是删除、查询功能，则仅设置响应字符编码为UTF-8  （提示：如果request.getMethod 是 "POST"或“PUT”，则请求和响应字符编码，否则仅设置响应编码）  2. 日志过滤器  打印被请求的资源名称和请求时间，格式如“school.ctl @ 2019年11月20日17：32”  第14周周六进行期末答辩，内容为本学期1-13周讲课内容，作业内容占99%。报告模板于下13周周四之前发放。 |

源程序(打印版本不得含有注释，不能修改字体字号)

|  |
| --- |
| //201802104029陈义乾  001 package bysj.Controller.Department;  002  003 import bysj.Daomain.Degree;  004 import bysj.Daomain.Department;  005 import bysj.Service.DegreeService;  006 import bysj.Service.DepartmentService;  007 import bysj.Util.JSONUtil;  008 import com.alibaba.fastjson.JSON;  009 import com.alibaba.fastjson.JSONObject;  010  011 import javax.servlet.ServletException;  012 import javax.servlet.annotation.WebServlet;  013 import javax.servlet.http.HttpServlet;  014 import javax.servlet.http.HttpServletRequest;  015 import javax.servlet.http.HttpServletResponse;  016 import java.io.IOException;  017 import java.sql.SQLException;  018 import java.util.Collection;  019  020 @WebServlet("/department.ctl")  021 public class DepartmentController extends HttpServlet {  022  023  024  025  026  027  028  029  030  031 protected void doGet(HttpServletRequest request, HttpServletResponse response)  032 throws ServletException, IOException {  033  034 String id\_str = request.getParameter("id");  035  036 if(id\_str == null){  037 try {  038  039 responseDepartments(response);  040 } catch (SQLException e) {  041 e.printStackTrace();  042 response.getWriter().println("列表失败，数据库操作异常");  043 } catch(Exception e){  044 e.printStackTrace();  045 response.getWriter().println("列表失败，网络异常");  046 }  047 }else{  048 int id = Integer.parseInt(id\_str);  049 try {  050  051 responseDepartment(id, response);  052 } catch (SQLException e) {  053 e.printStackTrace();  054 response.getWriter().println("查找失败，数据库操作异常");  055 } catch(Exception e){  056 e.printStackTrace();  057 response.getWriter().println("查找失败，网络异常");  058 }  059 }  060 }  061  062  063 protected void doPost(HttpServletRequest request, HttpServletResponse response)  064 throws ServletException, IOException {  065  066 String department\_json= JSONUtil.getJSON(request);  067  068 Department departmentToAdd= JSON.parseObject(department\_json, Department.class);  069  070 JSONObject resp = new JSONObject();  071 try{  072  073 boolean add=DepartmentService.getInstance().add(departmentToAdd);  074 if(add){  075  076 resp.put("message", "增加成功");  077 }else {  078 resp.put("message","增加失败");  079 }  080 } catch(SQLException e){  081 e.printStackTrace();  082 resp.put("message","添加失败，数据库操作异常");  083 } catch(Exception e){  084 e.printStackTrace();  085 resp.put("message", "添加失败，网络异常");  086 }  087  088 response.getWriter().println(resp);  089 }  090  091  092 protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  093  094 String id\_str=request.getParameter("id");  095  096 int id=Integer.parseInt(id\_str);  097  098 JSONObject resp = new JSONObject();  099 try{  100  101 boolean delete=DepartmentService.getInstance().delete(id);  102 if(delete){  103  104 resp.put("message","成功删除");  105 }else{  106 resp.put("message","删除失败");  107 }  108  109 }catch (SQLException e){  110 e.printStackTrace();  111 resp.put("message","删除失败，数据库操作异常");  112 } catch(Exception e){  113 e.printStackTrace();  114 resp.put("message","删除失败，网络异常");  115 }  116 response.getWriter().println(resp);  117 }  118  119  120 protected void doPut(HttpServletRequest request, HttpServletResponse response)  121 throws ServletException, IOException {  122 String department\_json = JSONUtil.getJSON(request);  123  124 Department departmentToAdd = JSON.parseObject(department\_json, Department.class);  125  126 response.setContentType("html/text;charset=UTF8");  127  128 JSONObject message = new JSONObject();  129  130 try {  131 DepartmentService.getInstance().update(departmentToAdd);  132  133 message.put("statusCode", "200");  134 message.put("message", "更新成功");  135 } catch (SQLException e) {  136 e.printStackTrace();  137 message.put("message", "更新失败，数据库操作异常");  138 } catch(Exception e){  139 message.put("message", "更新失败，网络异常");  140 }  141  142 response.getWriter().println(message);  143 }  144  145  146 private void responseDepartment(int id, HttpServletResponse response)  147 throws ServletException, IOException, SQLException {  148  149 Department department = DepartmentService.getInstance().find(id);  150 String department\_json = JSON.toJSONString(department);  151  152 response.getWriter().println(department\_json);  153 }  154  155  156 private void responseDepartments(HttpServletResponse response)  157 throws ServletException, IOException, SQLException {  158  159 Collection<Department> departments = DepartmentService.getInstance().findAll();  160 String departments\_json = JSON.toJSONString(departments);  161  162 response.getWriter().println(departments\_json);  163 }  164 } |
| 001 package bysj.Controller.Degree;  002  003 import bysj.Daomain.Degree;  004 import bysj.Service.DegreeService;  005 import bysj.Util.JSONUtil;  006 import com.alibaba.fastjson.JSON;  007 import com.alibaba.fastjson.JSONObject;  008  009 import javax.servlet.ServletException;  010 import javax.servlet.annotation.WebServlet;  011 import javax.servlet.http.HttpServlet;  012 import javax.servlet.http.HttpServletRequest;  013 import javax.servlet.http.HttpServletResponse;  014 import java.io.IOException;  015 import java.sql.SQLException;  016 import java.util.Collection;  017  018 @WebServlet("/degree.ctl")  019 public class DegreeController extends HttpServlet {  020  021  022  023  024  025  026  027  028  029 protected void doGet(HttpServletRequest request, HttpServletResponse response)  030 throws ServletException, IOException {  031  032 String id\_str = request.getParameter("id");  033  034 if(id\_str == null){  035 try {  036  037 responseDegrees(response);  038 } catch (SQLException e) {  039 e.printStackTrace();  040 response.getWriter().println("列表失败，数据库操作异常");  041 } catch(Exception e){  042 response.getWriter().println("列表失败，网络异常");  043 }  044 }else{  045 int id = Integer.parseInt(id\_str);  046 try {  047  048 responseDegree(id, response);  049 } catch (SQLException e) {  050 e.printStackTrace();  051 response.getWriter().println("查找失败，数据库操作异常");  052 } catch(Exception e){  053 e.printStackTrace();  054 response.getWriter().println("查找失败，网络异常");  055 }  056 }  057 }  058  059  060 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  061  062 String degree\_json= JSONUtil.getJSON(request);  063  064 Degree degreeToAdd= JSON.parseObject(degree\_json,Degree.class);  065  066 JSONObject resp = new JSONObject();  067 try{  068  069 boolean add=DegreeService.getInstance().add(degreeToAdd);  070 if(add){  071  072 resp.put("message", "添加成功");  073 }else {  074 resp.put("message","添加失败");  075 }  076 }catch(SQLException e){  077 e.printStackTrace();  078 resp.put("message","添加失败，数据库操作异常");  079 } catch(Exception e){  080 resp.put("message", "添加失败，网络异常");  081 }  082  083 response.getWriter().println(resp);  084 }  085  086  087 protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  088  089 String id\_str=request.getParameter("id");  090  091 int id=Integer.parseInt(id\_str);  092  093 JSONObject resp = new JSONObject();  094 try{  095  096 boolean delete=DegreeService.getInstance().delete(id);  097 if(delete){  098  099 resp.put("message", "成功删除");  100 }else{  101 resp.put("message","删除失败");  102 }  103 }catch (SQLException e){  104 e.printStackTrace();  105 resp.put("message", "删除失败，数据库操作异常");  106 } catch(Exception e){  107 e.printStackTrace();  108 resp.put("message", "删除失败，网络异常");  109 }  110 response.getWriter().println(resp);  111 }  112  113  114 protected void doPut(HttpServletRequest request, HttpServletResponse response)  115 throws ServletException, IOException {  116 String degree\_json = JSONUtil.getJSON(request);  117  118 Degree degreeToAdd = JSON.parseObject(degree\_json, Degree.class);  119  120 JSONObject message = new JSONObject();  121  122 try {  123 boolean update=DegreeService.getInstance().update(degreeToAdd);  124 if(update){  125  126 message.put("message", "更新成功");  127 }else{  128 message.put("message","更新失败");  129 }  130 } catch (SQLException e) {  131 e.printStackTrace();  132 message.put("message", "更新失败，数据库操作异常");  133 } catch(Exception e){  134 e.printStackTrace();  135 message.put("message", "更新失败，网络异常");  136 }  137  138 response.getWriter().println(message);  139 }  140  141  142 private void responseDegree(int id, HttpServletResponse response)  143 throws ServletException, IOException, SQLException {  144  145 Degree degree = DegreeService.getInstance().find(id);  146 String degree\_json = JSON.toJSONString(degree);  147  148 response.getWriter().println(degree\_json);  149 }  150  151 private void responseDegrees(HttpServletResponse response)  152 throws ServletException, IOException, SQLException {  153  154 Collection<Degree> degrees = DegreeService.getInstance().findAll();  155 String degrees\_json = JSON.toJSONString(degrees);  156  157 response.getWriter().println(degrees\_json);  158 }  159 } |
| 001 package bysj.Controller.Proftitle;  002  003 import bysj.Daomain.Degree;  004 import bysj.Daomain.Department;  005 import bysj.Daomain.ProfTitle;  006 import bysj.Service.DegreeService;  007 import bysj.Service.DepartmentService;  008 import bysj.Service.ProfTitleService;  009 import bysj.Util.JSONUtil;  010 import com.alibaba.fastjson.JSON;  011 import com.alibaba.fastjson.JSONObject;  012  013 import javax.servlet.ServletException;  014 import javax.servlet.annotation.WebServlet;  015 import javax.servlet.http.HttpServlet;  016 import javax.servlet.http.HttpServletRequest;  017 import javax.servlet.http.HttpServletResponse;  018 import java.io.IOException;  019 import java.sql.SQLException;  020 import java.util.Collection;  021  022 @WebServlet("/proftitle.ctl")  023 public class ProftitleController extends HttpServlet {  024  025  026  027  028  029  030  031  032  033 protected void doGet(HttpServletRequest request, HttpServletResponse response)  034 throws ServletException, IOException {  035  036 String id\_str = request.getParameter("id");  037  038 if(id\_str == null){  039 try {  040  041 responseProftiles(response);  042 } catch (SQLException e) {  043 e.printStackTrace();  044 response.getWriter().println("列表失败，数据库操作异常");  045 } catch(Exception e){  046 e.printStackTrace();  047 response.getWriter().println("列表失败，网络异常");  048 }  049 }else{  050 int id = Integer.parseInt(id\_str);  051 try {  052  053 responseProftitle(id, response);  054 } catch (SQLException e) {  055 e.printStackTrace();  056 response.getWriter().println("查找失败，数据库操作异常");  057 } catch(Exception e){  058 response.getWriter().println("查找失败，网络异常");  059 }  060 }  061 }  062  063  064 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  065  066 String proftitle\_json= JSONUtil.getJSON(request);  067  068 ProfTitle profTitleToAdd= JSON.parseObject(proftitle\_json,ProfTitle.class);  069  070 JSONObject resp = new JSONObject();  071 try{  072  073 boolean add=ProfTitleService.getInstance().add(profTitleToAdd);  074 if(add){  075  076 resp.put("message", "添加成功");  077 }else {  078 resp.put("message","添加失败");  079 }  080 }catch(SQLException e){  081 e.printStackTrace();  082 resp.put("message","添加失败，数据库操作异常");  083 } catch(Exception e){  084 e.printStackTrace();  085 resp.put("message", "添加失败，网络异常");  086 }  087  088 response.getWriter().println(resp);  089 }  090  091  092 protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  093  094 String id\_str=request.getParameter("id");  095  096 int id=Integer.parseInt(id\_str);  097  098 JSONObject resp = new JSONObject();  099 try{  100  101 boolean delete=ProfTitleService.getInstance().delete(id);  102 if(delete){  103  104 resp.put("message","成功删除");  105 }else {  106 resp.put("message","删除失败");  107 }  108 }catch (SQLException e){  109 e.printStackTrace();  110 resp.put("message","删除失败，数据库操作异常");  111 } catch(Exception e){  112 e.printStackTrace();  113 resp.put("message","删除失败，网络异常");  114 }  115 response.getWriter().println(resp);  116 }  117  118  119 protected void doPut(HttpServletRequest request, HttpServletResponse response)  120 throws ServletException, IOException {  121 String proftitle\_json = JSONUtil.getJSON(request);  122  123 ProfTitle proftitleToAdd = JSON.parseObject(proftitle\_json, ProfTitle.class);  124  125 JSONObject message = new JSONObject();  126  127 try {  128 boolean update=ProfTitleService.getInstance().update(proftitleToAdd);  129 if(update){  130  131 message.put("statusCode", "200");  132 message.put("message", "更新成功");  133 }else {  134 message.put("message","更新失败");  135 }  136  137 } catch (SQLException e) {  138 e.printStackTrace();  139 message.put("message", "更新失败，数据库操作异常");  140 } catch(Exception e){  141 e.printStackTrace();  142 message.put("message", "更新失败，网络异常");  143 }  144  145 response.getWriter().println(message);  146 }  147  148  149 private void responseProftitle(int id, HttpServletResponse response)  150 throws ServletException, IOException, SQLException {  151  152 ProfTitle profTitle = ProfTitleService.getInstance().find(id);  153 String proftitle\_json = JSON.toJSONString(profTitle);  154  155 response.getWriter().println(proftitle\_json);  156 }  157  158  159 private void responseProftiles(HttpServletResponse response)  160 throws ServletException, IOException, SQLException {  161  162 Collection<ProfTitle> proftitles = ProfTitleService.getInstance().findAll();  163 String proftitles\_json = JSON.toJSONString(proftitles);  164  165 response.getWriter().println(proftitles\_json);  166 }  167 } |
| 001 package bysj.Controller.School;  002  003 import bysj.Daomain.ProfTitle;  004 import bysj.Daomain.School;  005 import bysj.Daomain.Teacher;  006 import bysj.Service.ProfTitleService;  007 import bysj.Service.SchoolService;  008 import bysj.Service.TeacherService;  009 import bysj.Util.JSONUtil;  010 import com.alibaba.fastjson.JSON;  011 import com.alibaba.fastjson.JSONObject;  012 import com.mysql.cj.protocol.x.XMessage;  013  014 import javax.servlet.ServletException;  015 import javax.servlet.annotation.WebServlet;  016 import javax.servlet.http.HttpServlet;  017 import javax.servlet.http.HttpServletRequest;  018 import javax.servlet.http.HttpServletResponse;  019 import java.io.IOException;  020 import java.sql.SQLException;  021 import java.util.Collection;  022  023 @WebServlet("/school.ctl")  024 public class SchoolController extends HttpServlet {  025  026  027  028  029  030  031  032  033  034 protected void doGet(HttpServletRequest request, HttpServletResponse response)  035 throws ServletException, IOException {  036  037 String id\_str = request.getParameter("id");  038  039 if(id\_str == null){  040 try {  041 responseSchools(response);  042 } catch (SQLException e) {  043 e.printStackTrace();  044 response.getWriter().println("列表失败，数据库操作异常");  045 } catch(Exception e){  046 response.getWriter().println("列表失败，网络异常");  047 }  048 }else{  049 int id = Integer.parseInt(id\_str);  050 try {  051 responseSchool(id, response);  052 } catch (SQLException e) {  053 e.printStackTrace();  054 response.getWriter().println("查找失败，数据库操作异常");  055 } catch(Exception e){  056 response.getWriter().println("查找失败，网络异常");  057 }  058 }  059 }  060  061  062 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  063  064 String school\_json= JSONUtil.getJSON(request);  065  066 School schoolToAdd= JSON.parseObject(school\_json,School.class);  067  068 JSONObject resp = new JSONObject();  069 try{  070  071 boolean add=SchoolService.getInstance().add(schoolToAdd);  072 if (add){  073  074 resp.put("message", "添加成功");  075 }else {  076 resp.put("message","添加失败");  077 }  078 }catch(SQLException e){  079 e.printStackTrace();  080 resp.put("message","添加失败，数据库操作异常");  081 } catch(Exception e){  082 e.printStackTrace();  083 resp.put("message", "添加失败，网络异常");  084 }  085  086 response.getWriter().println(resp);  087 }  088  089  090 protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  091  092 String id\_str=request.getParameter("id");  093  094 int id=Integer.parseInt(id\_str);  095  096 JSONObject resp = new JSONObject();  097 try{  098  099 boolean delete=SchoolService.getInstance().delete(id);  100 if(delete){  101  102 resp.put("message","成功删除");  103 }else {  104 resp.put("message","删除失败");  105 }  106 }catch (SQLException e){  107 e.printStackTrace();  108 resp.put("message","删除失败，数据库操作异常");  109 } catch(Exception e){  110 e.printStackTrace();  111 resp.put("message","删除失败，网络异常");  112 }  113 response.getWriter().println(resp);  114 }  115  116  117 protected void doPut(HttpServletRequest request, HttpServletResponse response)  118 throws ServletException, IOException {  119 String school\_json = JSONUtil.getJSON(request);  120  121 School schoolToAdd = JSON.parseObject(school\_json, School.class);  122  123 JSONObject message = new JSONObject();  124  125 try {  126 boolean update=SchoolService.getInstance().update(schoolToAdd);  127 if(update){  128  129 message.put("message", "更新成功");  130 }else {  131 message.put("message","更新失败");  132 }  133 } catch (SQLException e) {  134 e.printStackTrace();  135 message.put("message", "更新失败，数据库操作异常");  136 } catch(Exception e){  137 e.printStackTrace();  138 message.put("message", "更新失败，网络异常");  139 }  140  141 response.getWriter().println(message);  142 }  143  144  145 private void responseSchool(int id, HttpServletResponse response)  146 throws ServletException, IOException, SQLException {  147  148 School school = SchoolService.getInstance().find(id);  149 String school\_json = JSON.toJSONString(school);  150  151 response.getWriter().println(school\_json);  152 }  153  154  155 private void responseSchools(HttpServletResponse response)  156 throws ServletException, IOException, SQLException {  157  158 Collection<School> schools = SchoolService.getInstance().findAll();  159 String schools\_json = JSON.toJSONString(schools);  160  161 response.getWriter().println(schools\_json);  162 }  163 } |
| 001 package bysj.Controller.Teacher;  002  003 import bysj.Daomain.Teacher;  004 import bysj.Daomain.Teacher;  005 import bysj.Service.TeacherService;  006 import bysj.Service.TeacherService;  007 import bysj.Util.JSONUtil;  008 import com.alibaba.fastjson.JSON;  009 import com.alibaba.fastjson.JSONObject;  010  011 import javax.servlet.ServletException;  012 import javax.servlet.annotation.WebServlet;  013 import javax.servlet.http.HttpServlet;  014 import javax.servlet.http.HttpServletRequest;  015 import javax.servlet.http.HttpServletResponse;  016 import java.io.IOException;  017 import java.sql.SQLException;  018 import java.util.Collection;  019  020 @WebServlet("/teacher.ctl")  021 public class TeacherController extends HttpServlet {  022  023  024  025  026  027  028  029  030  031 protected void doGet(HttpServletRequest request, HttpServletResponse response)  032 throws ServletException, IOException {  033  034 String id\_str = request.getParameter("id");  035  036 if(id\_str == null){  037 try {  038  039 responseTeachers(response);  040 } catch (SQLException e) {  041 e.printStackTrace();  042 response.getWriter().println("列表失败，数据库操作异常");  043 } catch(Exception e){  044 e.printStackTrace();  045 response.getWriter().println("列表失败，网络异常");  046 }  047 }else{  048 int id = Integer.parseInt(id\_str);  049 try {  050  051 responseTeacher(id, response);  052 } catch (SQLException e) {  053 e.printStackTrace();  054 response.getWriter().println("查找失败，数据库操作异常");  055 } catch(Exception e){  056 e.printStackTrace();  057 response.getWriter().println("查找失败，网络异常");  058 }  059 }  060 }  061  062  063 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  064  065 String teacher\_json= JSONUtil.getJSON(request);  066  067 Teacher teacherToAdd=JSON.parseObject(teacher\_json,Teacher.class);  068  069 JSONObject resp = new JSONObject();  070 try{  071  072 boolean add=TeacherService.getInstance().add(teacherToAdd);  073 if(add){  074  075 resp.put("message", "添加成功");  076 }else {  077 resp.put("message","添加失败");  078 }  079 }catch(SQLException e){  080 e.printStackTrace();  081 resp.put("message","添加失败，数据库操作异常");  082 } catch(Exception e){  083 e.printStackTrace();  084 resp.put("message", "添加失败，网络异常");  085 }  086  087 response.getWriter().println(resp);  088 }  089  090  091 protected void doDelete(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {  092  093 String id\_str=request.getParameter("id");  094  095 int id=Integer.parseInt(id\_str);  096  097 JSONObject resp = new JSONObject();  098 try{  099 boolean delete=TeacherService.getInstance().delete(id);  100 if(delete){  101 resp.put("message","成功删除");  102 }else {  103 resp.put("message","删除失败");  104 }  105 }catch (SQLException e){  106 e.printStackTrace();  107 resp.put("message","删除失败，数据库操作异常");  108 } catch(Exception e){  109 e.printStackTrace();  110 resp.put("message","删除失败，网络异常");  111 }  112 response.getWriter().println(resp);  113 }  114  115  116 protected void doPut(HttpServletRequest request, HttpServletResponse response)  117 throws ServletException, IOException {  118 String teacher\_json = JSONUtil.getJSON(request);  119  120 Teacher teacherToAdd = JSON.parseObject(teacher\_json, Teacher.class);  121  122 JSONObject message = new JSONObject();  123  124 try {  125 boolean update=TeacherService.getInstance().update(teacherToAdd);  126 if(update){  127  128 message.put("statusCode", "200");  129 message.put("message", "更新成功");  130 }else {  131 message.put("message","更新失败");  132 }  133 } catch (SQLException e) {  134 e.printStackTrace();  135 message.put("message", "更新失败，数据库操作异常");  136 } catch(Exception e){  137 e.printStackTrace();  138 message.put("message", "更新失败，网络异常");  139 }  140  141 response.getWriter().println(message);  142 }  143  144  145 private void responseTeacher(int id, HttpServletResponse response)  146 throws ServletException, IOException, SQLException {  147  148 Teacher teacher = TeacherService.getInstance().find(id);  149 String teacher\_json = JSON.toJSONString(teacher);  150  151 response.getWriter().println(teacher\_json);  152 }  153  154  155 private void responseTeachers(HttpServletResponse response)  156 throws ServletException, IOException, SQLException {  157  158 Collection<Teacher> teachers = TeacherService.getInstance().findAll();  159 String teachers\_json = JSON.toJSONString(teachers);  160  161 response.getWriter().println(teachers\_json);  162 }  163 } |
| 01 import javax.servlet.\*;  02 import javax.servlet.annotation.WebFilter;  03 import javax.servlet.http.HttpServletRequest;  04 import javax.servlet.http.HttpServletResponse;  05 import java.io.IOException;  06  07  08 @WebFilter(filterName = "Filter0",urlPatterns = "/\*")  09 public class Filter0 implements Filter {  10 public void destroy() {  11 }  12  13 public void doFilter(ServletRequest req, ServletResponse resp, FilterChain chain) throws ServletException, IOException {  14 System.out.println("Filter0 - encodimg begins");  15  16 HttpServletRequest request=(HttpServletRequest)req;  17  18 HttpServletResponse response=(HttpServletResponse) resp;  19  20 String path=request.getRequestURI();  21  22 if(path.contains("/login")){  23 System.out.println("请求包含/login，故不处理编码");  24 }else {  25 String method=request.getMethod();  26 if(method.equals("PUT")||method.equals("POST")){  27 request.setCharacterEncoding("UTF-8");  28 }  29 response.setContentType("text/html;charset =UTF-8");  30 }  31  32 chain.doFilter(req, resp);  33 System.out.println("Filter0 - encoding ends");  34 }  35  36 public void init(FilterConfig config) throws ServletException {  37  38 }  39  40 } |
| 01 import javax.servlet.\*;  02 import javax.servlet.annotation.WebFilter;  03 import javax.servlet.http.HttpServletRequest;  04 import java.io.IOException;  05 import java.util.Calendar;  06  07 @WebFilter(filterName = "Filter1",urlPatterns = "/\*")  08 public class Filter1 implements Filter {  09 public void destroy() {  10 }  11  12 public void doFilter(ServletRequest req, ServletResponse resp, FilterChain chain) throws ServletException, IOException {  13  14 HttpServletRequest request=(HttpServletRequest)req;  15  16 String path=request.getRequestURI();  17 Calendar cal = Calendar.getInstance();  18 String time = cal.get(Calendar.YEAR) + " 年 " +  19 (cal.get(Calendar.MONTH) + 1) + " 月 " +  20 cal.get(Calendar.DATE) + "日" +  21 cal.get(Calendar.HOUR\_OF\_DAY) + ": " +  22 cal.get(Calendar.MINUTE);  23 System.out.println("Fi;ter1 begin");  24 System.out.println(path+" @ "+time);  25  26 chain.doFilter(req, resp);  27 System.out.println("Filter1 ends");  28 }  29  30 public void init(FilterConfig config) throws ServletException {  31  32 }  33  34 } |

运行结果

|  |
| --- |
| JW_201802104029_27陈义乾01  JW_201802104029_27陈义乾02 |
|  |
|  |
|  |
|  |
|  |