**Html**

<!-- 按钮 -->

<div class="row">

<div class="col-md-2 col-md-offset-1">

<button class="btn btn-danger" id="delAll">删除</button>

</div>

</div>

<!-- 显示表格数据 -->

<div class="row">

<div class="col-md-10 col-md-offset-1">

<br/>

<table class="table table-bordered table-hover" border="0">

<tbody>

<tr>

<th><input type="checkbox" id="delAllEmps" ></th>

<th>#</th>

<th>empName</th>

<th>gender</th>

<th>email</th>

<th>deptName</th>

<th>操作</th>

</tr>

<c:if test="${!empty pageInfo }">

<%-- <c:forEach items="${employeeList}" var="employee"> --%>

<c:forEach items="${pageInfo.list}" var="employee">

<tr>

<td><input type="checkbox" id="delOneEmp" class="delOneEmp"></td>

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

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

<td>${employee.gender.equals("M")?"男":"女"}</td>

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

<td>${employee.department.deptName }</td>

<td>

</td>

</tr>

</c:forEach>

</c:if>

</tbody>

</table>

</div>

</div>

**JQuery:**

//点击顶端多选框，对底下的多选框全选

$("#delAllEmps").click(function(){

//attr获取checked是undefined;

//我们这些dom原生的属性；attr获取自定义属性的值；

//prop修改和读取dom原生属性的值

$(".delOneEmp").prop("checked",$(this).prop("checked"));//这里只能用class也就是"."，不能用“#”设置，原因不详

});

//点击全部删除，就批量删除

$("#delAll").click(function(){

//

var empNames = "";

var del\_idstr = "";

$.each($(".delOneEmp:checked"),function(){

//this

empNames += $(this).parents("tr").find("td:eq(2)").text()+",";

//组装员工id字符串

del\_idstr += $(this).parents("tr").find("td:eq(1)").text()+"-";

});

//去除empNames多余的,

empNames = empNames.substring(0, empNames.length-1);

//去除删除的id多余的-

del\_idstr = del\_idstr.substring(0, del\_idstr.length-1);

if(confirm("确认删除【"+empNames+"】吗？")){

//发送ajax请求删除

$.ajax({

url:"<%=basePath%>employee/deleteAllEmps/"+del\_idstr,

type:"DELETE",

success:function(result){

alert(result.msg);

//回到当前页面

//to\_page(currentPage);

window.location.href = "<%=basePath%>employee/getEmps";

}

});

}

});

**Controller:**

**@ResponseBody**

**@RequestMapping(value="/deleteAllEmps/{ids}",method=RequestMethod.DELETE)**

**public Msg deleteEmp(@PathVariable("ids")String ids){**

**//批量删除**

**if(ids.contains("-")){**

**List<Integer> del\_ids = new ArrayList<>();**

**String[] str\_ids = ids.split("-");**

**//组装id的集合**

**for (String string : str\_ids) {**

**int a=Integer.parseInt(string);**

**employeeService.delete(a);**

**}**

**}else{**

**Integer id = Integer.parseInt(ids);**

**employeeService.delete(id);**

**}**

**return Msg.success();**

**}**

**Msg类\*（参考）**

**package com.cn.hi.pojo;**

**import java.util.HashMap;**

**import java.util.Map;**

**/\*\***

**\* 通用的返回的类**

**\***

**\* @author lfy**

**\***

**\*/**

**public class Msg {**

**//状态码 100-成功 200-失败**

**private int code;**

**//提示信息**

**private String msg;**

**//用户要返回给浏览器的数据**

**private Map<String, Object> extend = new HashMap<String, Object>();**

**public static Msg success(){**

**Msg result = new Msg();**

**result.setCode(100);**

**result.setMsg("处理成功！");**

**return result;**

**}**

**public static Msg fail(){**

**Msg result = new Msg();**

**result.setCode(200);**

**result.setMsg("处理失败！");**

**return result;**

**}**

**public Msg add(String key,Object value){**

**this.getExtend().put(key, value);**

**return this;**

**}**

**public int getCode() {**

**return code;**

**}**

**public void setCode(int code) {**

**this.code = code;**

**}**

**public String getMsg() {**

**return msg;**

**}**

**public void setMsg(String msg) {**

**this.msg = msg;**

**}**

**public Map<String, Object> getExtend() {**

**return extend;**

**}**

**public void setExtend(Map<String, Object> extend) {**

**this.extend = extend;**

**}**

**}**