

08 租客信息的管理

笔记本： 课堂笔记-房屋租赁系统

创建时间： 2020/9/19 20:34

更新时间： 2020/9/19 21:41

作者： yishuihan1104@163.com

1、租客信息的查询功能

1.1 后台功能处理

1.1.1 dao层接口和实现

```
@Component("customerMapper")
public interface CustomerMapper extends BaseMapper<Customer> {

    List<Customer> queryAll(Customer customer);

}
```

实现

```
<select parameterType="com.yanzhen.model.Customer"
resultType="com.yanzhen.model.Customer">
    select * from customer
    <where>
        <if test="identity!=null and identity!=''">
            and identity like '%${identity}%'
        </if>
        <if test="custname!=null and custname!=''">
            and custname like '%${custname}%'
        </if>
    </where>
</select>
```

1.1.2 service层接口和实现

```

/**
 * 高级查询 支持分页效果
 */
PageInfo<Customer> queryAll(int page, int limit, Customer customer);

```

实现

```

@Autowired
private CustomerMapper customerMapper;

@Override
public PageInfo<Customer> queryAll(int page, int limit, Customer
customer) {
    PageHelper.startPage(page, limit);
    //查询租客信息列表
    List<Customer> list=customerMapper.queryAll(customer);
    PageInfo<Customer> pageInfo=new PageInfo<>(list);
    return pageInfo;
}

```

1.1.3 controller功能实现

```

@RequestMapping("/queryCoutomerAll")
public Object findCoutomer(Customer customer, @RequestParam(defaultValue =
"1") Integer page ,
                                @RequestParam(defaultValue = "10") Integer
limit) {
    JsonObject object=new JsonObject();
    //获取记录信息
    PageInfo<Customer> pageInfo=
customerService.queryAll(page, limit, customer);
    object.setMsg("ok");
    object.setData(pageInfo.getList());
    object.setCode(0);
    object.setCount(pageInfo.getTotal());
    return object;
}

```

1.2 页面的处理

1.2.1 点击菜单页面转发

```
{
  "title": "租客信息管理",
  "href": "",
  "icon": "fa fa-calendar",
  "target": "_self",
  "child": [
    {
      "title": "租客信息管理",
      "href": "page/tenant/TenantIndex.html",
      "icon": "fa fa-gears",
      "target": "_self"
    }
  ]
}
```

1.2.2 创建主页信息

```
<div >
  <div >

    <div >
      租客姓名:
      <div >
        <input name="custname" autocomplete="off">
      </div>
      身份证号码:
      <div >
        <input name="identity" autocomplete="off">
      </div>
      <button data-type="reload">高级查询</button>
    </div>

    <script type="text/html" >
      <div >
        <button lay-event="add"> 添加 </button>
        <button lay-event="delete"> 删除 </button>
      </div>
    </script>

    <table lay-filter="currentTableFilter"></table>
```

```

        <script type="text/html" >
            <a lay-event="edit">编辑</a>
            <a lay-event="delete">删除</a>

        </script>

    </div>
</div>

<script>
    layui.use(['form', 'table', 'miniPage', 'element'], function () {
        var $ = layui.jquery,
            form = layui.form,
            table = layui.table,
            miniPage = layui.miniPage;

        table.render({
            elem: '#currentTableId',
            url: 'http://localhost:8888/customer/queryCutomerAll',
            toolbar: '#toolbarDemo',
            defaultToolbar: ['filter', 'exports', 'print', {
                title: '提示',
                layEvent: 'LAYTABLE_TIPS',
                icon: 'layui-icon-tips'
            }
        ]],
            cols: [[
                {type: "checkbox", width: 50},
                {field: 'id', width: 80, title: 'ID', sort: true},
                {field: 'custname', width: 80, title: '租客姓名'},
                {field: 'identity', width: 80, title: '身份证号码'},
                {field: 'sex', width: 80, title: '性别'},
                {field: 'address', width: 80, title: '地址'},
                {field: 'phone', title: '电话', minWidth: 80},
                {field: 'career', width: 80, title: '职业'},
                {title: '操作', minWidth: 150, toolbar: '#currentTableBar',
            align: "center"}
        ]],
            limits: [10, 15, 20, 25, 50, 100],
            limit: 15,
            page: true,
            id: 'testReload',
            skin: 'line'
        });
    });

```

```

var $ = layui.$, active = {
  reload: function() {
    var custname = $('#custname');
    var identity = $('#identity');
    //执行重载
    table.reload('testReload', {
      page: {
        curr: 1 //重新从第 1 页开始
      },
      where: {
        identity: identity.val(),
        custname: custname.val()
      }
    }, 'data');
  }
};

$('#.demoTable .layui-btn').on('click', function() {
  var type = $(this).data('type');
  active[type] ? active[type].call(this) : '';
});

});
</script>

```

1.2.3 展示效果

The screenshot displays the '租赁管理系统' (Leasing Management System) interface. The sidebar on the left contains navigation links for '户主信息管理', '租客信息管理', '房屋信息管理', '管理员工信息管理', '公告管理', '异常页面', and '其它界面'. The main content area is titled '租客信息管理' (Tenant Information Management) and includes search filters for '租客姓名' (Tenant Name) and '身份证号码' (ID Number), along with a '高级查询' (Advanced Search) button. Below the filters is a table with columns: ID, 租客... (Tenant Name), 身份... (ID Number), 性别 (Gender), 地址 (Address), 电话 (Phone), 职业 (Occupation), and 操作 (Action). The table contains two rows of data: one for '刘晓' (Liu Xiao) with ID 1002, and one for '王小明' (Wang Xiaoming) with ID 1001. Each row has '编辑' (Edit) and '删除' (Delete) buttons. A red box highlights the table and its controls. The page also features a pagination bar at the bottom showing '共 2 条' (Total 2 items) and '25 条/页' (25 items per page).

2、租客信息的添加

2.1 后台controller层实现

```
@ApiOperation(value = "新增租户信息表")
@PostMapping("/addCustomer")
public R add(@RequestBody Customer customer) {
    int num= customerService.add(customer);
    if(num>0) {
        return R.ok();
    }
    return R.fail("租客信息添加失败,请重新添加");
}
```

2.2 前台页面处理

2.2.1 页面的设计

```
<div >

    <div >
        <div >
            <label >身份证号</label>
            <div >
                <input type="text" name="identity" lay-verify="required" lay-reqtext="身份证号不能为空" placeholder="请输入用户名" value="" >
                <tip>填写自己管理账号的名称。</tip>
            </div>
        </div>
        <div >
            <label >租客姓名</label>
            <div >
                <input type="text" name="custname" lay-verify="required" lay-reqtext="姓名不能为空" placeholder="请输入用户名" value="" >
            </div>
        </div>
    </div>

    <div >
        <label >性别</label>
        <div >
            <input type="radio" name="sex" value="男" title="男">
```

```

        <input type="radio" name="sex" value="女" title="女">
    </div>
</div>
<div >
    <label >联系方式</label>
    <div >
        <input type="number" name="phone" lay-verify="required" lay-reqtext="手机不能为空" placeholder="请输入手机" value="" >
    </div>
</div>
<div >
    <label >联系地址</label>
    <div >
        <input type="text" name="address" >
    </div>
</div>
<div >
    <label >职业</label>
    <div >
        <input type="email" name="career" >
    </div>
</div>

<div >
    <label >备注信息</label>
    <div >
        <textarea name="remark" placeholder="请输入备注信息">
    </textarea>
    </div>
</div>

<div >
    <div >
        <button lay-submit lay-filter="saveBtn">确认保存</button>
    </div>
</div>
</div>
</div>

```

2.2.2 监听添加事件

```

* toolbar事件监听
*/
table.on('toolbar(currentTableFilter)', function (obj) {
    if (obj.event === 'add') { // 监听添加操作
        var content = miniPage.getHrefContent('page/tenant/add.html');
        var openWH = miniPage.getOpenWidthHeight();

        var index = layer.open({
            title: '添加租客信息',
            type: 1,
            shade: 0.2,
            maxmin: true,
            shadeClose: true,
            area: [openWH[0] + 'px', openWH[1] + 'px'],
            offset: [openWH[2] + 'px', openWH[3] + 'px'],
            content: content,
        });
    }
});

```

2.2.3 展示效果

租赁管理系统

常规管理 租赁管理 统计分析

首页 / 常规管理 / 租客信息管理 / 租客信息管理

添加租客信息

身份证号* 请输入用户名
填写自己管理账号的名称。

租客姓名* 请输入用户名

性别 ☐ 男 ☐ 女

联系方式* 请输入手机

联系地址*

职业

备注信息 请输入备注信息

2.2.4 提交实现添加功能



3、租客信息的删除

3.1 controller层代码

```
@ApiOperation(value = "删除租户信息表")
@RequestMapping("/deleteByIds")
public R delete(String ids) {
    //获取前端传过来的ids集合 "1,2,3,4,5"
    List<String> list= Arrays.asList(ids.split(","));
    //遍历list集合进行删除操作
    for(String id:list){
        Long idLong=Long.parseLong(id);
        customerService.delete(idLong);
    }
    return R.ok();
}
```

3.1.2前端监听事件

```
/**
 * toolbar事件监听
 */
table.on('toolbar(currentTableFilter)', function (obj) {
    if (obj.event === 'add') { // 监听添加操作

    } else if (obj.event === 'delete') { // 监听删除操作
```

```

        var checkStatus = table.checkStatus(obj.config.id)
        , data = checkStatus.data;
    /**
     * 首先判断是否选择信息，如果没有提示
     * 如果有
     *     1、获取选中的id集合
     *     2、通过ajax请求发送id集合，进行业务处理
     */
    if(data.length==0){
        layer.msg("请选要删除的记录信息");
    }else{
        //获取删除id的集合
        var ids=getCheckId(data);
        layer.confirm("你真的要删除了吗？",function(index){
            deleteByIds(ids,index);
        })
    }
    // layer.alert(JSON.stringify(data));
});

/**
 * 获取批量删除选中的id集合
 */
function getCheckId(data){
    var arr=new Array();
    for(var i=0;i<data.length;i++){
        arr.push(data[i].id);
    }
    return arr.join(",");
};

/**
 * 删除功能的实现
 */
function deleteByIds(ids,index){
    $.ajax({
        url:"http://localhost:8888/customer/deleteByIds",
        type:"POST",
        // contentType:"application/json",
        // data:JSON.stringify(data.field),
        data:{"ids":ids},
        success:function(result){
            if(result.code==200){

```

```

        layer.msg("删除成功",{
            icon:6,
            time:500
        },function(){
            parent.window.location.reload();//重新页面
            // var iframeIndex =
parent.layer.getFrameIndex(window.name);
            // parent.layer.close(iframeIndex);
        });

    }else{
        layer.msg("删除失败");
    }
}
})
return false;
};

```

3.1.3效果图展示



4、租客信息编辑

4.1 后台controller处理

```

@ApiOperation(value = "更新租户信息表")
@RequestMapping("/updateCustomer")
public R update(@RequestBody Customer customer){
    int num= customerService.updateData(customer);
    if(num>0){
        return R.ok();
    }
}

```

```
}  
    return R.fail("失败");  
}
```

4.2 前台页面处理

4.2.1 页面设计

```
<div >  
  
    <div lay-filter="updateSubmit">  
        <input type="hidden" name="id" >  
        <div >  
            <label >租客姓名</label>  
            <div >  
                <input type="text" name="custname" lay-verify="required" >  
            </div>  
        </div>  
  
        <div >  
            <label >性别</label>  
            <div >  
                <input type="radio" name="sex" value="男" title="男"  
checked="">  
                <input type="radio" name="sex" value="女" title="女">  
            </div>  
        </div>  
  
        <div >  
            <label >身份证号</label>  
            <div >  
                <input type="text" name="identity" lay-verify="required" lay-  
reqtext="身份证号不能为空" placeholder="请输入用户名" value="" >  
                <tip>填写自己管理账号的名称。</tip>  
            </div>  
        </div>  
  
        <div >  
            <label >手机</label>  
            <div >  
                <input type="number" name="phone" lay-verify="required" lay-  
reqtext="手机不能为空" placeholder="请输入手机" value="" >  
            </div>  
        </div>  
    </div>  
</div>
```

```
        </div>
    </div>
    <div >
        <label >联系地址</label>
        <div >
            <input type="text" name="address" >
        </div>
    </div>
    <div >
        <label >职业</label>
        <div >
            <input type="text" name="career" >
        </div>
    </div>

    <div >
        <label >备注信息</label>
        <div >
            <textarea name="remarks" placeholder="请输入备注信息">
</textarea>
        </div>
    </div>

    <div >
        <div >
            <button lay-submit lay-filter="updateSubmit">确认修改
</button>
        </div>
    </div>
</div>
```

4.2.2 监听内容

```

table.on('tool(currentTableFilter)', function (obj) {
    var data = obj.data;
    if (obj.event === 'edit') {
        var content = miniPage.getHrefContent('page/tenant/edit.html');
        var openWH = miniPage.getOpenWidthHeight();

        var index = layer.open({
            title: '编辑租客信息',
            type: 1,
            shade: 0.2,
            maxmin: true,
            shadeClose: true,
            area: [openWH[0] + 'px', openWH[1] + 'px'],
            offset: [openWH[2] + 'px', openWH[3] + 'px'],
            content: content,

```

4.2.3 提交功能实现

```

layui.use(['form', 'table'], function () {
    var form = layui.form,
        layer = layui.layer,
        // table = layui.table,
        $ = layui.$;
    form.render();
    //监听提交
    form.on('submit(updateSubmit)', function (data) {
        console.log(data.field);
        //向后台发送数据并进行添加操作
        $.ajax({
            url: "http://localhost:8888/customer/updateCustomer",
            type: "POST",
            contentType: "application/json",
            data: JSON.stringify(data.field),
            success: function (result) {
                //把json对象转string
                // result=JSON.parse(result);
                console.log(result)
                if (result.code == 200) {
                    layer.msg("修改成功", {
                        icon: 6,
                        time: 500
                    }, function () {
                        parent.window.location.reload(); //重新页面

```

```

        var iframeIndex =
parent.layer.getFrameIndex(window.name);
        parent.layer.close(iframeIndex);
    });

    }else{
        layer.msg("修改失败");
    }
}
})
return false;
});
});

```

5、效果图

The screenshot displays the Rental Management System (租赁管理系统) interface. The sidebar on the left contains navigation options: 户主信息管理 (Landlord Information Management), 租客信息管理 (Tenant Information Management), 房屋信息管理 (House Information Management), 管理员信息管理 (Admin Information Management), 公告管理 (Announcement Management), and 异常页面 (Exception Pages). The main content area shows the Tenant Information Management (租客信息管理) page. It includes a search bar with fields for 租客姓名 (Tenant Name) and 身份证号码 (ID Card Number), and a 高级查询 (Advanced Search) button. Below the search bar is a table with columns: ID, 租客姓名 (Tenant Name), 身份证号码 (ID Card Number), 性别 (Gender), 地址 (Address), 电话 (Phone), 职业 (Occupation), and 操作 (Operations). The table contains two records: one for 刘晓1 (Liu Xiao1) with ID 1002, and another for 王小明 (Wang Xiaoming) with ID 1001. The 操作 column for each record contains 编辑 (Edit) and 删除 (Delete) buttons. At the bottom of the table, there is a pagination bar showing 1 of 2 pages, with a 确定 (Confirm) button and a 15条/页 (15 items per page) dropdown.