客户端首页

管理需求

1.查询启示列表信息（支持高级查询），按照时间倒叙排列。

2.评论记录数据的统计

3.根据状态统计，进行可视化渲染

4.查询分类并统计数量

5.公告信息的查询（最新）

6.日历插件添加。

1. 后台接口提供

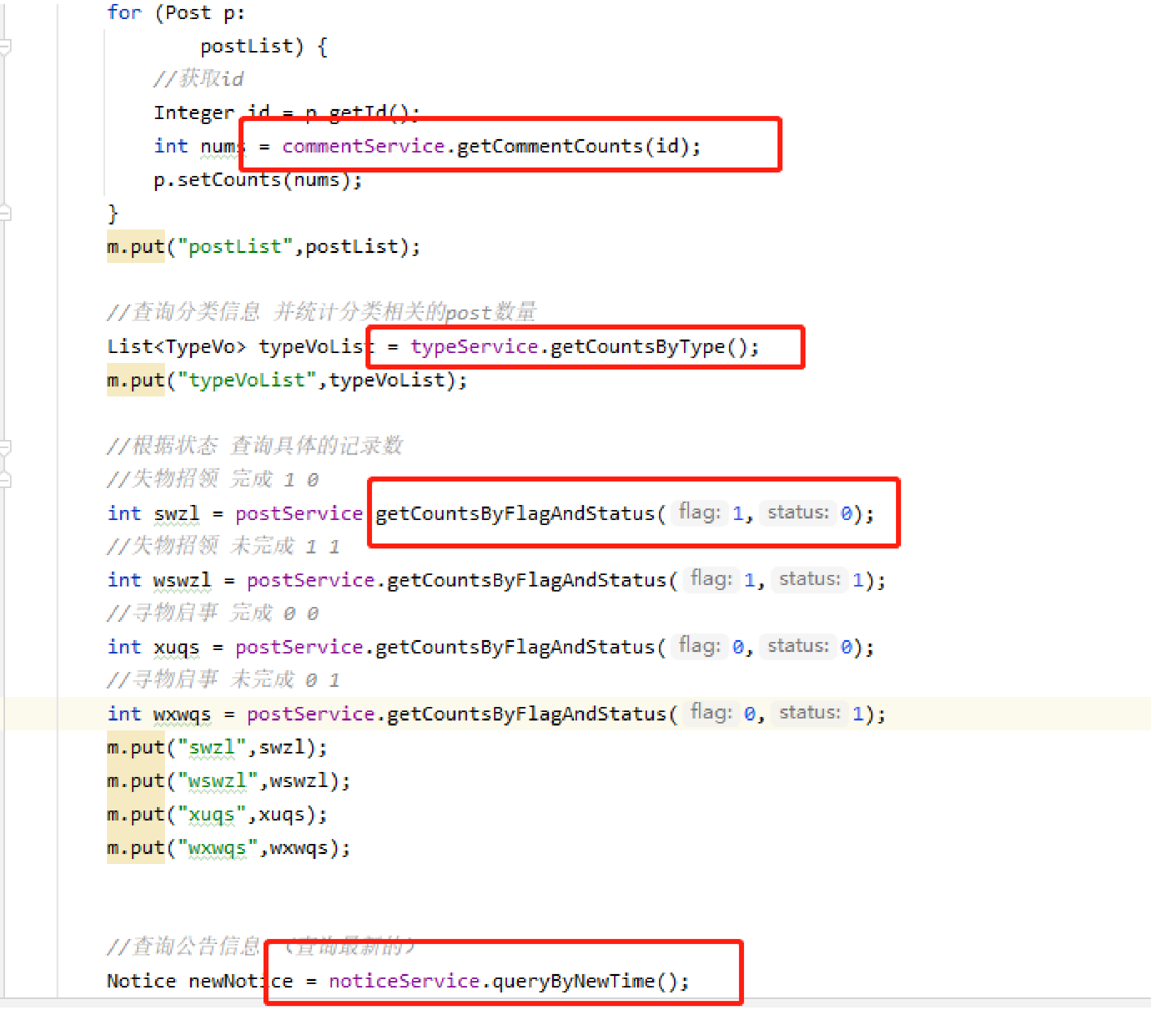
Controller 层

传参：Map 对象，作为高级查询的条件。

返回值：Map 对象。

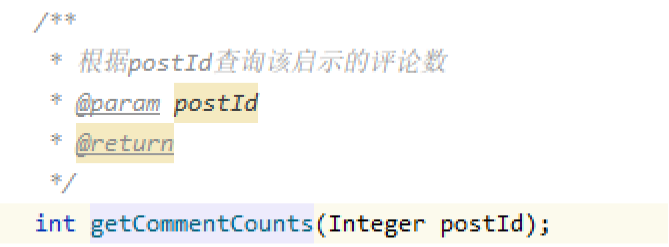
```
/**
     * 用来渲染主页的功能
    */
    @RequestMapping("/queryIndexList")
    public Map queryIndexList(@RequestBody Map<String,String> map){
//    public Map queryIndexList(){

       //创建 map 对象 用以返回前端
       Map m = new HashMap();

        /**
         * 1 查询 post 列表信息（支持高级查询）
        *       1.1 支持高级查询（标签、分类、标题）
        *       1.2 查询过程中需要统计每条记录的评论数（根据 id 查询评论信息）
        * 2 查询分类信息 并统计分类相关的 post 数量
        * 3 查询公告信息 （查询最新的）
        * 4 根据标签（状态）统计
        */

        Post post = new Post();

        if(map.containsKey("flag")){
            String flag = map.get("flag");
            post.setFlag(Integer.parseInt(flag));
        }
        if(map.containsKey("status")){
            String status = map.get("status");
            post.setStatus(Integer.parseInt(status));
        }
        if(map.containsKey("typeName")){
            String typeName = map.get("typeName");
            post.setTypeName(typeName);
        }
        if(map.containsKey("title")){
            String title = map.get("title");
            post.setTitle(title);
        }

        //查询post 列表信息（支持高级查询）
       PageInfo<Post> postPageInfo = postService.findAll(1,100,post);
        List<Post> postList = postPageInfo.getList();
        //遍历列表 放入评论数
       for (Post p:
                postList) {
            //获取id
            Integer id = p.getId();
            int nums = commentService.getCommentCounts(id);
            p.setCounts(nums);
        }
        m.put("postList",postList);

        //查询分类信息 并统计分类相关的post 数量
       List<TypeVo> typeVoList = typeService.getCountsByType();
        m.put("typeVoList",typeVoList);

        //根据状态 查询具体的记录数
       //失物招领 完成 1 0
        int swzl = postService.getCountsByFlagAndStatus(1,0);
        //失物招领 未完成 1 1
        int wswzl = postService.getCountsByFlagAndStatus(1,1);
        //寻物启事 完成 0 0
        int xuqs = postService.getCountsByFlagAndStatus(0,0);
        //寻物启事 未完成 0 1
        int wxwqs = postService.getCountsByFlagAndStatus(0,1);
        m.put("swzl",swzl);
        m.put("wswzl",wswzl);
        m.put("xuqs",xuqs);
        m.put("wxwqs",wxwqs);


        //查询公告信息 （查询最新的）
       Notice newNotice = noticeService.queryByNewTime();
        m.put("newNotice",newNotice);
        return m;
    }
```

Service 层 Dao 层的实现

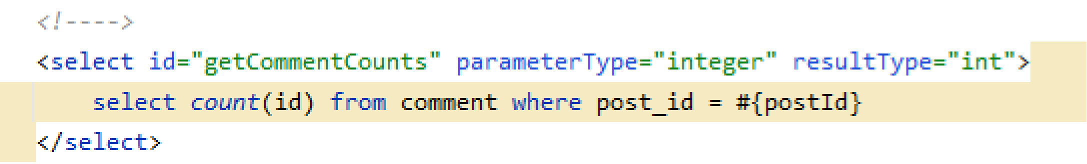


```
for (Post p:
        postList) {
    //获取id
    Integer id = p.getId();
    int nums = commentService.getCommentCounts(id);
    p.setCounts(nums);
}
m.put("postList",postList);

//查询分类信息 并统计分类相关的post数量
List<TypeVo> typeVoList = typeService.getCountsByType();
m.put("typeVoList",typeVoList);

//根据状态 查询具体的记录数
//失物招领 完成 1 0
int swzl = postService.getCountsByFlagAndStatus( flag: 1, status: 0);
//失物招领 未完成 1 1
int wswzl = postService.getCountsByFlagAndStatus( flag: 1, status: 1);
//寻物启事 完成 0 0
int xuqs = postService.getCountsByFlagAndStatus( flag: 0, status: 0);
//寻物启事 未完成 0 1
int wxwqs = postService.getCountsByFlagAndStatus( flag: 0, status: 1);
m.put("swzl",swzl);
m.put("wswzl",wswzl);
m.put("xuqs",xuqs);
m.put("wxwqs",wxwqs);


//查询公告信息 （查询最新的）
Notice newNotice = noticeService.queryByNewTime();
```

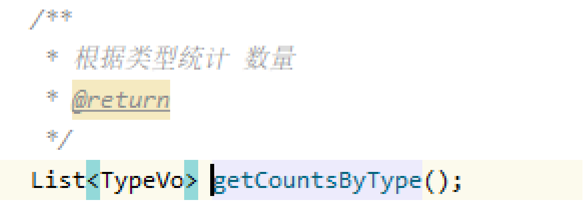
主要看上面圈住的四个接口



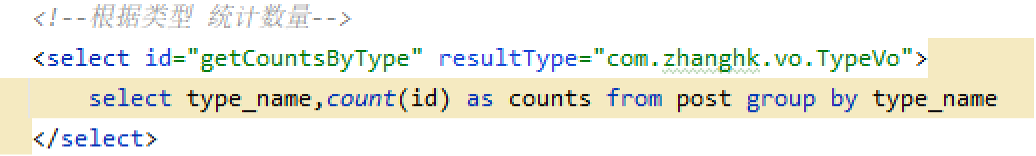
```
/**
 * 根据postId查询该启示的评论数
 * @param postId
 * @return
 */
int getCommentCounts(Integer postId);
```



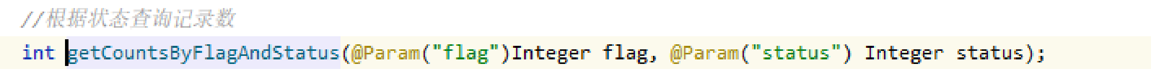
```
<!---->
<select id="getCommentCounts" parameterType="integer" resultType="int">
    select count(id) from comment where post_id = #{postId}
</select>
```



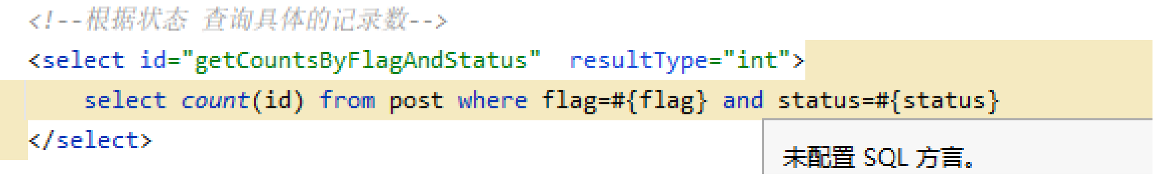
```java
/**
 * 根据类型统计 数量
 * @return
 */
List<TypeVo> getCountsByType();
```



```xml
<!--根据类型 统计数量-->
<select id="getCountsByType" resultType="com.zhanghk.vo.TypeVo">
    select type_name,count(id) as counts from post group by type_name
</select>
```

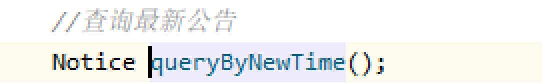


```java
//根据状态查询记录数
int getCountsByFlagAndStatus(@Param("flag")Integer flag, @Param("status") Integer status);
```

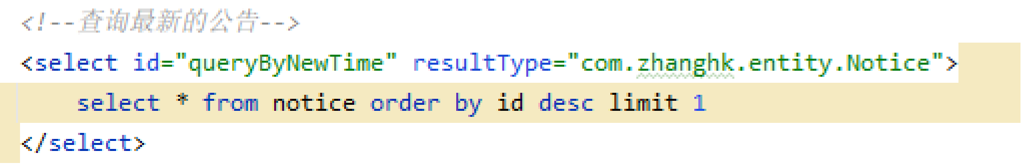


```xml
<!--根据状态 查询具体的记录数-->
<select id="getCountsByFlagAndStatus"  resultType="int">
    select count(id) from post where flag=#{flag} and status=#{status}
</select>
```

未配置 SQL 方言。



```java
//查询最新公告
Notice queryByNewTime();
```



```xml
<!--查询最新的公告-->
<select id="queryByNewTime" resultType="com.zhanghk.entity.Notice">
    select * from notice order by id desc limit 1
</select>
```

2. 前端页面处理

```javascript
//刚开始进入网页就调用
        getPost();

         //高级查询
        function selectPost(){
          var title=$('#selectTitle').val();
              //var content=$('#selectTitle').val();
          getPost(null,null,null,title);
        }

         //高级带条件的查询
        function getPost(flag,status,type,title){
            var values = {}
            if(flag!=null){
          values.flag=flag;
          values.status=status;
          }
            if(type!=null){
          values.typeName=type;
          }
          if(title!=null){
          values.title=title;
          }

            //向后台发送请求

axios.post('http://127.0.0.1:8888/post/queryIndexList',values).then(function
(response) {
            var data = response.data;
            //console.log(data);
            //获取postlist 集合
           var postList = data.postList;
            //console.log(postList)
            // 渲染数据
           let content = '';
            for(let obj of postList){
                content += `
                <div class="ui padded vertical segment m-padded-tb-large">
                        <div class="ui mobile reversed stackable grid">
                            <div class="eleven wide column">

                                <div class="item">
                                    <div ${obj.status==1 && obj.flag==0 ?'' :
'style="display:none;"'}  class="ui orange label horizontal" data-tooltip="暂未找到
">寻物启事</div>
                                    <div ${obj.status==0 && obj.flag==0 ?'' :
'style="display:none;"'}    class="ui green label horizontal" data-tooltip="已经找
到">寻物启事</div>
                                    <div ${obj.status==0 && obj.flag==1 ?'' :
'style="display:none;"' }  class="ui yellow label horizontal" data-tooltip="已经认
领">失物招领</div>
                                    <div ${obj.status==1 && obj.flag==1 ?'' :
'style="display:none;"'}  class="ui blue label horizontal" data-tooltip="正在寻找">
失物招领</div>
                                    <i class="tags icon"></i>${obj.typeName}
                                    <i class="map marker icon"></i>${obj.address}
                                </div>

                                <h3 class="ui header">${obj.title}</h3>

                                <p class="m-text">${obj.content}</p>
                                <div class="ui grid">
                                    <div class="twelve wide column">
                                        <div class="ui mini horizontal link list">

                                            <div class="item">
                                                <img
src="https://unsplash.it/100/100?image=1005" alt="" class="ui avatar image">
```

```
                                                <div class="content"><a href="#"
class="header">${obj.uname}</a></div>
                                            </div>
                                            <div class="item">
                                                <i class="calendar
icon"></i>${obj.ctime}
                                            </div>
                                            <div class="item">
                                                <i class="eye
icon"></i>${obj.viewCount}
                                            </div>
                                            <div class="item">
                                                <i class="comment
icon"></i>${obj.counts}
                                            </div>
                                        </div>
                                    </div>
                                    <div class="right aligned four wide column">
                                        <a href="detail.html?postId=${obj.id}"
class="ui teal basic label m-padded-tiny m-text-thin">详情</a>
                                    </div>
                                </div>
                            </div>

                            <div class="five wide column">
                                <a href="#" target="_blank">
                                    <img
src="https://unsplash.it/800/400?image=1005"  alt="" class="ui rounded image"> <!-
-用网站给的图片填充-->
                                </a>
                            </div>

                        </div>
                    </div>
                </div>

`
            }//for 循环结束标记

           //获取外围div 标签
           var mainDiv = document.getElementById("mainDiv");
            mainDiv.innerHTML = content;

            //标签的渲染--已经完成
           var tagsDiv = document.getElementById("tagsDiv");
            var  tagsContent = `
            <a href="javascript:void(0)" onclick="getPost(1,1)" class="ui teal
basic left pointing label m-margin-tb-large">
           正在寻找 <div class="detail">${data.wxwqs}</div>
            </a>
            <a href="javascript:void(0)" onclick="getPost(1,0)" class="ui teal
basic left pointing label m-margin-tb-large">
            已经找到 <div class="detail">${data.xuqs}</div>
            </a>
            <a href="javascript:void(0)" onclick="getPost(0,1)" class="ui teal
basic left pointing label m-margin-tb-large">
             等待认领 <div class="detail">${data.wswzl}</div>
             </a>
            <a href="javascript:void(0)" onclick="getPost(0,0)" class="ui teal
basic left pointing label m-margin-tb-large">
            已被认领 <div class="detail">${data.swzl}</div>
            </a>
            `;
            tagsDiv.innerHTML = tagsContent;

            // //分类的渲染
           // var typeDiv = document.getElementById("typeDiv");
            // var typeContent = '';
            // var typeVoList = data.typeVoList;
            // for(let type of typeVoList){

            //      typeContent +=  `
            //      <a href="javascript:void(0)"
onclick="getPost(null,null,'${type.typeName}')" class="ui teal basic left pointing
label m-margin-tb-tiny">
            //                 ${type.typeName} <div
class="detail">${type.counts}</div>
            //             </a>`;
            // }
            // typeDiv.innerHTML = typeContent;


            //渲染最新公告
           var noticeDiv = document.getElementById("noticeDiv");
            var noticeTitleDiv = document.getElementById("noticeTitle");
            var noticeTimeDiv = document.getElementById("noticeTime");
            var noticeTitle = data.newNotice.title;
            var noticeContent = data.newNotice.content;
            var noticeTime = data.newNotice.ctime;
            noticeTitleDiv.innerHTML = noticeTitle;
            noticeDiv.innerHTML = noticeContent;
            noticeTimeDiv.innerHTML = noticeTime;


            //渲染启示篇数
           $("#postNum").html(postList.length);

            //渲染统计图
           var myChart = echarts.init(document.getElementById("chartDiv"));
            //console.log(myChart);
            //统计图的配置项和数据
           myChart.setOption({
                tooltip: {},
                series: [{
                    radius: 35,
                    type: 'pie',
                    data: [{name: '已经找到',value: data.wxwqs},
                           {name: '暂未找到',value: data.xuqs},
                           {name: '等待认领',value: data.wswzl},
                           {name: '已经认领',value: data.swzl}]
                }]
            });

        }).catch(function (error) {})
```

```
    }
```