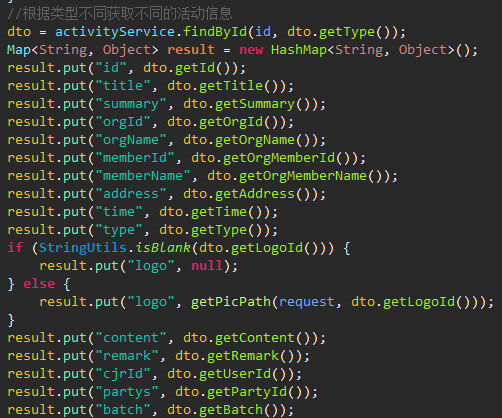
### Controller

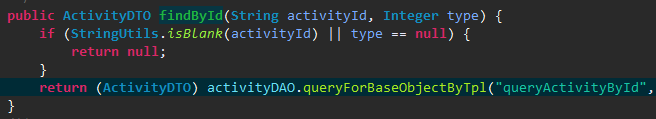
* 在controll er中，@RequestMapping(value = "/activitys/{id}", method = RequestMethod.GET)，用于映射url中红字部分http://localhost:8080/xspj/api/activitys/76acfc1c62ad4f89831788447b81fa63， 也就是说当url是这样的情况时，就会被controller捕获，并且method也是一致时，就会执行对应的方法getActivity（）。@PathVariable String id就是获得路径中的变量id，因此可以在方法中直接使用。
* 通过service获取数据（service通过dao获取数据），放在map对象result中。



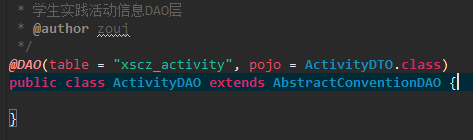
* 将获得的数据result以json的格式返回写入response中。

### Service

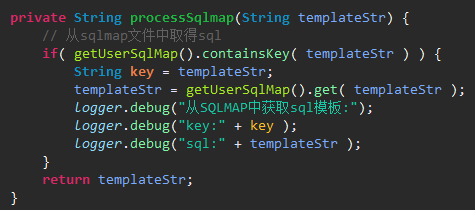
在controller中调用service中的方法，而在service中则是调用dao层进行数据操作，基本是直接返回dao的到的dto对象或者map对象，其中指定了sqlmap中定义的sql操作“queryActivityById“。



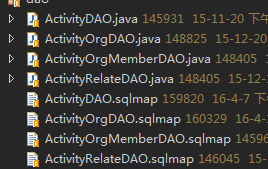
### Dao



在dao中，进行注解@DAO(table = "xscz\_activity", pojo = ActivityDTO.class)，指定操作的表是xscz\_activity表，操作的类是ActivityDTO。因为直接继承了 AbstractConventionDAO，所以里面有很多已经实现了的通用的方法了，比如activityDAO.queryForBaseObjectByTpl("queryActivityById", ConventionUtils.m("activityId", activityId, "type", type));这个方法指定了要执行的sql语句queryActivityById，该语句在sqlmap中已经提前写好了。通过下面的方法从sqlmap中读取sql语句。



### SqlMap

Sqlmap需要与dao放在一起，并且名字要一致，如下图：

在sqlmap中以如下格式写好要执行的sql语句：

Key=  
Sql语句  
Sql语句  
;

e.g：

