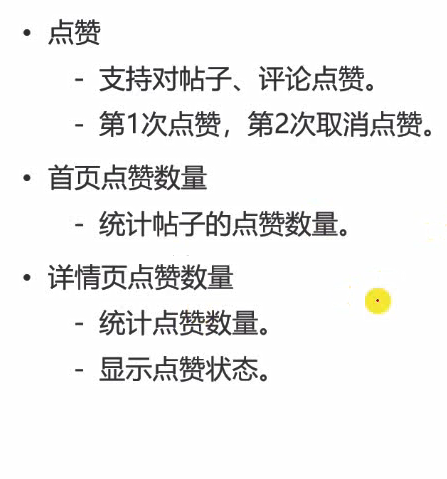
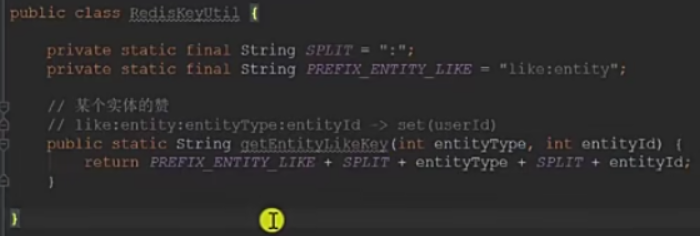
因为可能**同时有很多人给同一个人点赞**，所以对性能要求较高，用redis来存储



先去写一个生成redis Key的工具



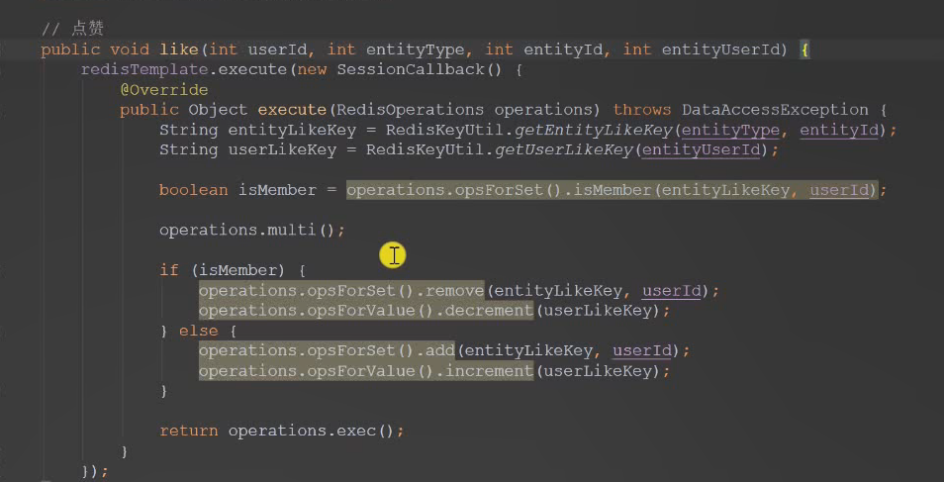
Service层

第三个方法是为了在帖子详情页里判断按钮的显示状态，**用RedisTemplate的isMember判断有没有点赞(set里有没有)**，若点赞1则显示已赞，若未赞0则显示赞，点赞是在帖子详情页里点击赞/已赞按钮的动作，传入给redis的值为set格式(里面有userId)

**新建一个redis的key，每点一次赞，这个key对应的value就加一**

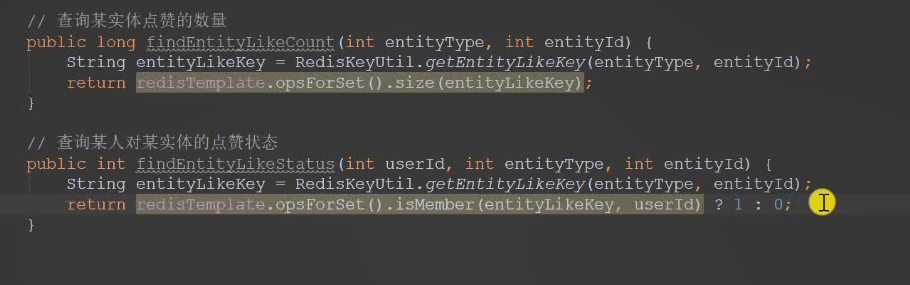
在**service**层，**给帖子点赞和用户获得的赞的数量加一应加在一个事务里**，

如果赞过就把这个用户id移出set，用户获得的赞减一；不在就将用户id add进set



还要确定**显示已赞/点赞, 和点赞数量**

因为可能后期我们需要查询都有哪些用户给此贴点过赞 ，查询数量在主页和帖子详情页里都有用到，在主页浏览时显示每一个帖子的已赞数量，在帖子详情页里的点赞按钮包括点赞数量

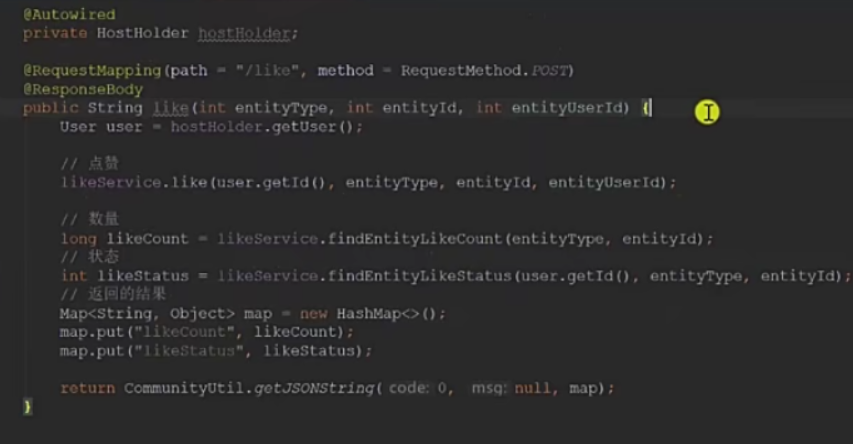


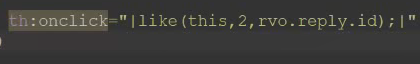
Controller:就是实施likeService里的点赞方法 然后把点赞数量和状态给model

如何获得到参数：

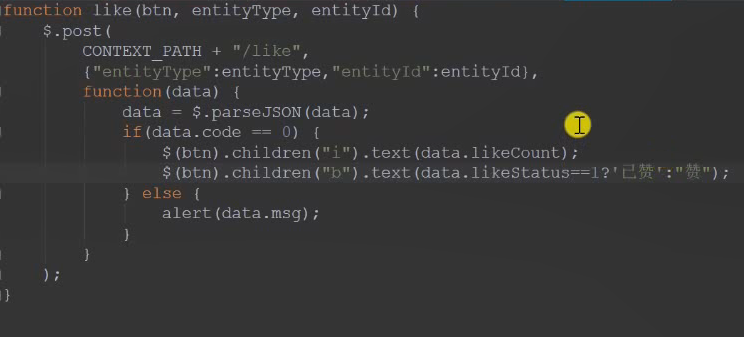
网页通过调用JavaScript方法

由于点赞的操作在帖子详情页里只是点赞按钮这块**局部刷新**，所以我们在controller层编写响应体时返回**json**格式的字符串，将点赞的数量和状态传入map中，将map转为json格式的字符串





在jQuery中编写JavaScript方法，JavaScript方法发送请求将参数传递，java返回json字符串，jQuery获得返回数据



由于在**主页显示每个帖子的点赞数量是直接在返回的网页上显示的，所以我们在主页的控制层调用点赞的服务层，获取点赞数量，传给model使其在网页上显示**，（注意：主页不能点赞，只能查看赞的数量）

**帖子的控制层**：帖子的**点赞数量**与上述同理，且要多给model传入一个**点赞状态，评论（评论是列表，列表通过map记录着每个评论的userID和内容）的点赞数量**，同理，但是要将其**传入给map**中而不是model，**因为评论列表会传给model，回复列表同理**