# Java项目结构

* pom.xml(项目的核心配置文件)
* main
  + java
    - com.register
      * entity(存放实体类，包括数据库的类)
      * controller（存放服务程序）
      * StartApplication（main）
  + resources
    - application.yml（配置web服务端口和数据库连接）

detail维护：

commentList:

List包含：

* id 评论独有的id
* name 评论人名称
* date 评论时间
* contentText 评论内容
* url 头像
* allReply 回复数目
* likeNum 点赞数目
* isLike 是否已经点赞
* replyList
  + name 回复人名字
  + contentStr 回复内容

点击评论，弹出评论框，输入内容，提交。

1. name, url 通过本账户用户信息得到
2. date通过系统时间得到
3. allReply, likeNum,
4. id自增长

处理方法：

加载评论时，通过评论对应文章id找到对应评论。每一个评论对应唯一一个评论者id，通过它得到评论人名称，头像进行显示。

需要后端传来的数据：评论数据，回复数据，评论人数据，回复人数据

ER图：

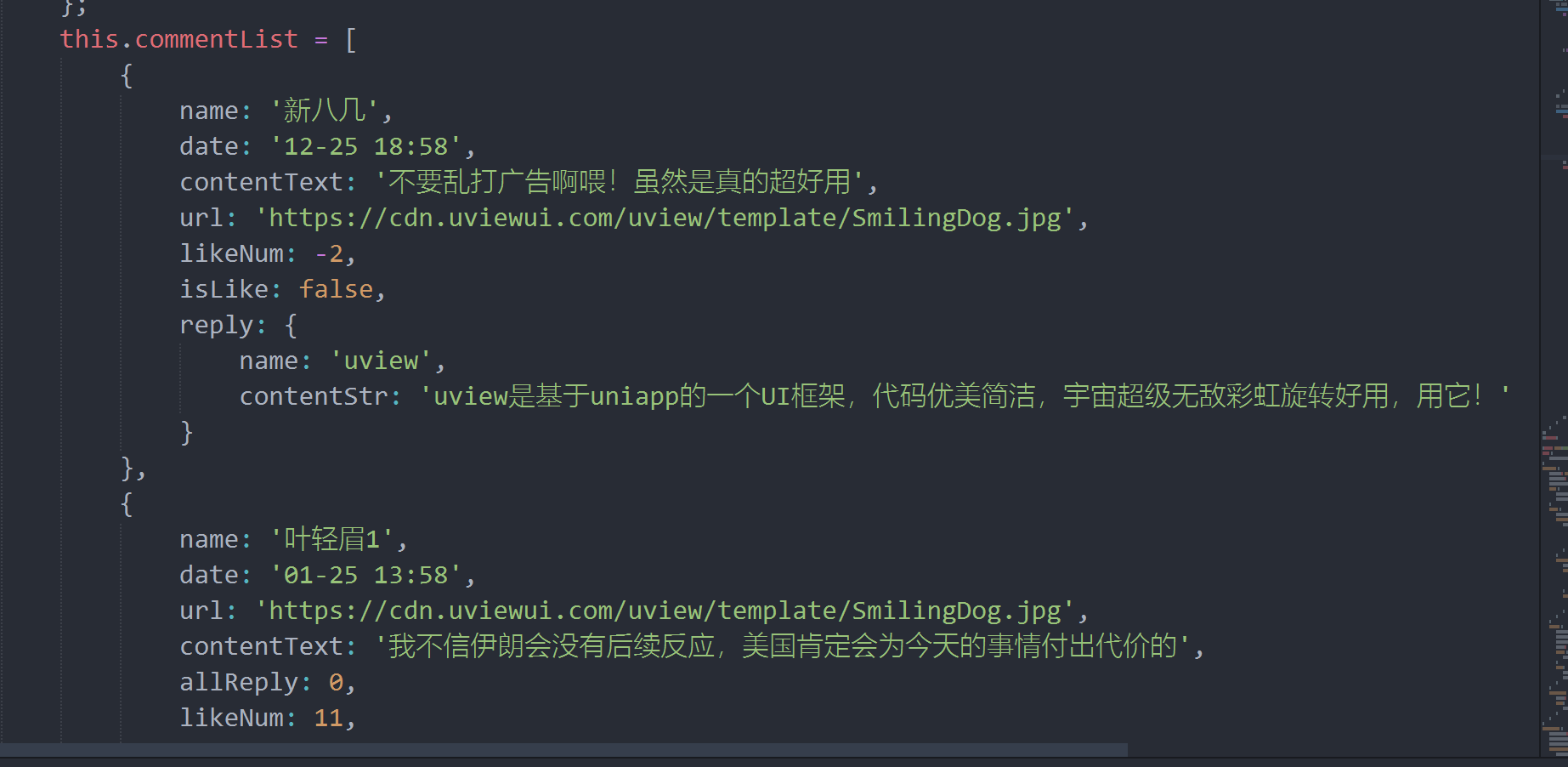
request：文章id

res：该文章的评论list，包含每个评论的id,name,…,replylist(回复者名字，内容；并且设置为0-2项)



request：评论id

res：该评论所有回复list，包括回复name，date，…，回复的回复（这个不是列表，只有一项）



er图 1.0：

回复

rply\_id <pk>

cmt\_id（对应评论id） <fk>

HFZ\_id（回复者id）<fk>

 contentStr

评论

cmt\_id <pk>

PLZ\_id（评论者id）<fk>

article\_id（文章） <fk>

date

 contentText

 allReply

 likeNum

文章包含：

|  |  |
| --- | --- |
| 英文名 | 中文名 |
| Eassyid | 文章id |
| contentText | 文章内容 |
| eassydate | 文章发布时间 |
| author | 作者名 |
| E\_photo | 图片 |
| Head | 标题 |