

测试输入数据://测试结束后请调用删除函数删除脏数据

{

“title”:”testTitle1”,

“content”:”testContent1”

“question”:{

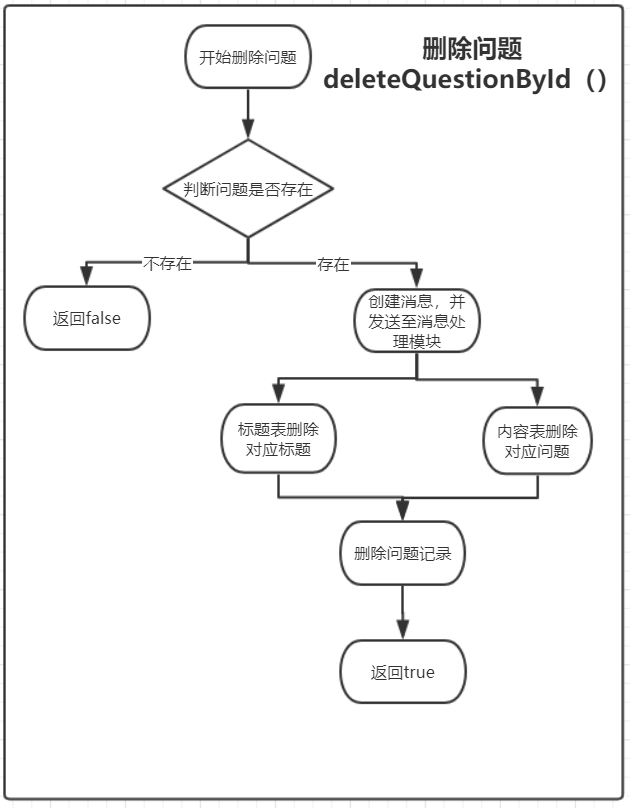
“authorId”://testMan1的id

“createTime”:date

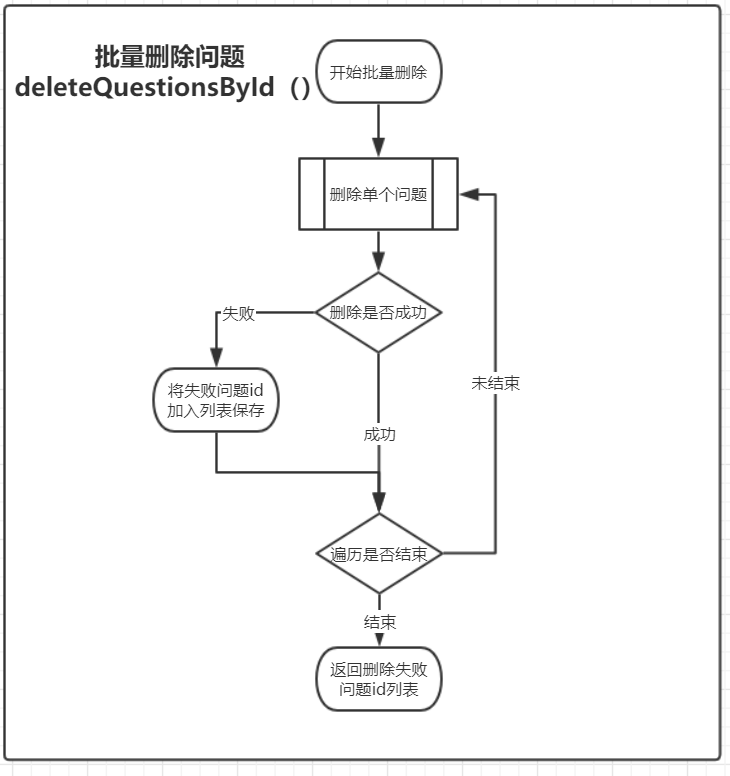
}

}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试函数 | 测试描述 | 预期结果 | 实际结果 | 测试评价 |
| addQuestion | 输入新增问题 | 数据库新增该问题的记录  创建作者提问数+1，积分和经验值获得增加  创作作者的消息数+1 |  |  |



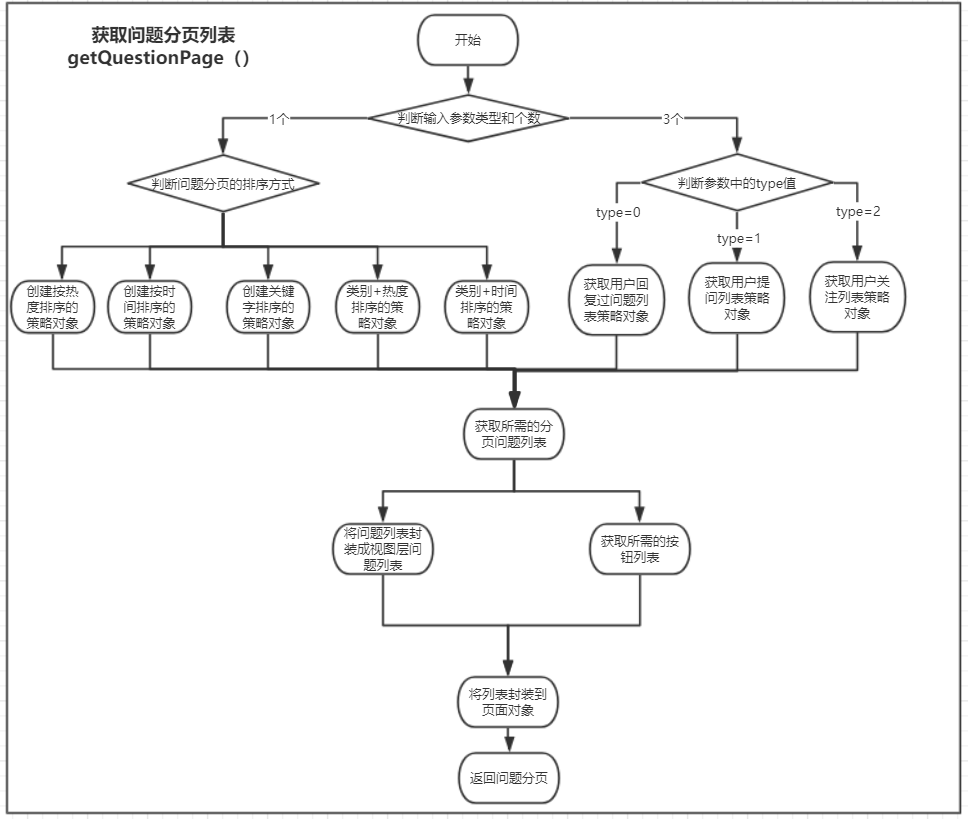
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试函数 | 测试描述 | 预期结果 | 实际结果 | 测试评价 |
| DeleteQuestionById | 删除一条已存在的问题记录 | 数据库中该条记录被删除  作者提问数-1  作者的消息数+1 |  |  |



测试前数据库填充数据：

调用addQuestion方法添加10条作者为test1用户的问题记录，并记录id序列

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试函数 | 测试描述 | 预期结果 | 实际结果 | 测试评价 |
| deleteQuestionById | 删除新增问题id序列的问题 | 数据库中id序列的问被删除;作者问题数-10;返回失败列表长度为0 |  |  |
| 在上述id序列中添加id为1、2，进行批量删除操作（包含部分不存在的问题） | 数据库中存在10个id对应问题被删除  作者问题数-10  返回序列包含1 /2两个数据  作者消息数+10 |  |  |



测试输入数据://测试结束后请调用删除函数删除脏数据

{

“title”:”testTitle1”,

“content”:”testContent1”

“question”:{

“authorId”:10,

“createTime”:”2020-01-01”

“kindId”:0,

“anoymous”:0,

“responseNum”:10

}

}，

{

“title”:”testTitle2”,

“content”:”testContent1”

“question”:{

“authorId”:10,

“createTime”:”2020-02-03”,

“kindId”:1,

“anoymous”:0,

“responseNum”:12

}

}，

{

“title”:”test3”,

“content”:”testContent1”

“question”:{

“authorId”:10,

“createTime”:”2020-03-02”,

“kindId”:0,

“anoymous”:1,

“responseNum”:8

}

}，

---------------------------------------------------------------------------------------------------------------------------------

//调用ResponseService.insertResponse(ResponseVO responseVO)添加回复

{

“response”:{

“questionId”://问题testTitle1的id

“authorId”://testMan1的id

“createTime”:”2020-10-02”

},

“content”:”responseContent1”

},

{

“response”:{

“questionId”://问题testTitle2的id

“authorId”://testMan1的id

“createTime”:”2020-10-03”,

},

“content”:”responseContent2”

}

---------------------------------------------------------------------------------------------------------------------------------

//调用AttentionService.insertAttention(Attention attention)插入关注记录

{

“userId”://testMan1的id,

“question”://testTiltle2的id,

“createTime”:”2020-03-12”,

“flag”:1

}

{

“userId”://testMan1的id,

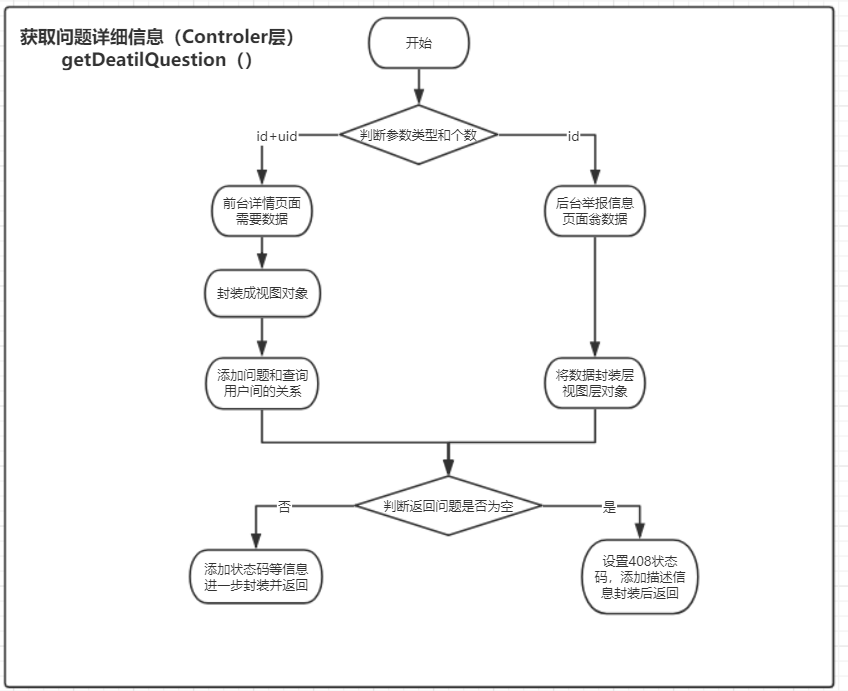
“question”://testTiltle3的id,

“createTime”:”2020-03-12”,

“flag”:1

}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试函数 | 测试描述 | 预期结果 | 实际结果 | 测试评价 |
| getQuestionPage  (QuestionPage) | 按热度排序 | 分页中的排序列表序列为test2-test1-test3 |  |  |
| 按时间排序 | 返回序列顺序为test3-test2-test1 |  |  |
| 按关键字Title排序 | 返回序列为test2-test1 |  |  |
| 按类别0+热度排序 | 返回序列test1-test3 |  |  |
| 按类别1+时间排序 | 返回序列test3-test1 |  |  |
| getQuestionPage  (Integer userId, QuestionPage page, int type) | 查看用户回复过的问题列表(type=0) | 返回序列test2-test1 |  |  |
| 查看用户提问列表(type=1) | 返回序列test3-test2-test1 |  |  |
| 获取用户关注问题列表(type=2) | 返回序列test3-test2 |  |  |



测试输入数据://测试结束后请调用删除函数删除脏数据

{

“title”:”testTitle1”,

“content”:”testContent1”

“question”:{

“authorId”:10,

“createTime”:”2020-01-01”

“kindId”:0,

“anoymous”:0,

“responseNum”:10

}

}，

---------------------------------------------------------------------------------------------------------------------------------//调用AttentionService.insertAttention(Attention attention)插入关注记录

{

“userId”://testMan1的id,

“question”://testTiltle2的id,

“createTime”:”2020-03-12”,

“flag”:1

}

---------------------------------------------------------------------------------------------------------------------------------

//调用AttentionService.insertAttention(Attention attention)插入关注记录

{

“reportorId”://举报人testMan1的id,

“questionId”://testTiltle1的id,

“createTime”:”2020-03-12”,

“flag”:1

}

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 测试函数 | 测试描述 | 预期结果 | 实际结果 | 测试评价 |
| getQuestionBById | 查询存在的问题 | 返回问题的数据库实体数据 |  |  |
| 查询不存在的问题 | 返回null |  |  |
| addRelationToUId | 添加查询问题与查者之间的关系 | questionVO.doesAttention=true  questionVO.doesReported=true |  |  |