1.如何实现一个评论 回复系统，@功能怎么实现

嵌套文档

```
"commentList": [
  {
    "commentId": 1,
    "userId": 8,
    "userName": "hobo",
    "headPortrait": "/home/hobo/blog/head-pic/8.jpg",
    "creatTime": "2019年12月13日 20:21",
    "commentMsg": "第一层",
    "replyVO": [
      {
        "commentId": 2,
        "userId": 5,
        "userName": "sunshine",
        "headPortrait": "/home/hobo/blog/head-pic/defult.jpeg",
        "replyUserId": 8,
        "replyUserName": "hobo",
        "creatTime": "2019年12月13日 20:22",
        "commentMsg": "第二层"
      },
      {
        "commentId": 3,
        "userId": 8,
        "userName": "hobo",
        "headPortrait": "/home/hobo/blog/head-pic/8.jpg",
        "replyUserId": 5,
        "replyUserName": "sunshine",
        "creatTime": "2019年12月13日 20:25",
        "commentMsg": "回复评论"
      }
    ]
  },
```

基于父子文档

```
DELETE my_index

PUT /my_index?pretty
{
  "mappings": {
    "properties": {
      "my_join_field": {
        "type": "join",
        "relations": {
          "question": "answer"
        }
      }
    }
  }
}


# 插入父
PUT /my_index/_doc/8?refresh&pretty
{
  "text": "This is a question",
  "my_join_field": {
```

```
      "name": "question"
    }
}

PUT /my_index/_doc/10?refresh&pretty
{
  "text": "This is a new question",
  "my_join_field": {
    "name": "question"
  }
}

PUT /my_index/_doc/12?refresh&pretty
{
  "text": "This is a new question",
  "my_join_field": {
    "name": "question"
  }
}

# 插入子
PUT /my_index/_doc/3?routing=8&refresh&pretty
{
  "text": "This is an answer",
  "my_join_field": {
    "name": "answer",
    "parent": "8"
  }
}


PUT /my_index/_doc/4?routing=10&refresh&pretty
{
  "text": "This is another answer",
  "my_join_field": {
    "name": "answer",
    "parent": "10"
  }
}

# 通过parent_id查询子文档
GET my_index/_search
{
  "query": {
    "parent_id": {
      "type": "answer",
      "id": "8"
    }
  }
}

# 查询relation
POST my_index/_search
{
 "script_fields": {
    "parent": {
      "script": {
        "source": "doc['my_join_field#question']"
```

```
        }
      }
    }
  }
```

2.最新的spring-data-elasticsearch 也支持7.x的操作了，实际项目中能不能用呢

可以　但一般也要考虑团队中技术是否统一　现在主流依然是restClient

3.使用ElasticsearchRestTemplate 怎么获取一条数据的版本号

```
UpdateQueryBuilder updateQueryBuilder = new UpdateQueryBuilder();
updateQueryBuilder.withId(docBean.getId()+"");
updateQueryBuilder.withUpdateRequest(updateRequest);
updateQueryBuilder.withClass(DocBean.class);
UpdateQuery build = updateQueryBuilder.build();
UpdateResponse update1 = elasticsearchTemplate.update(build);
```