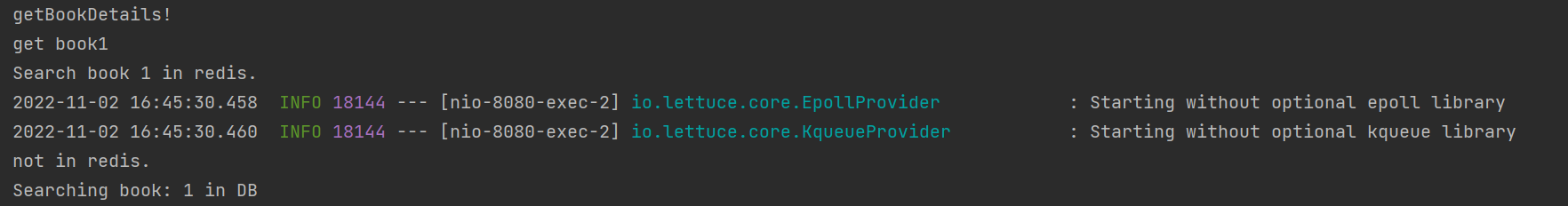
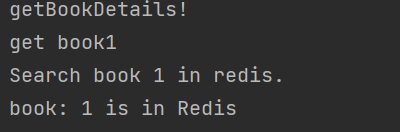
1.首次运行getbook 1时：



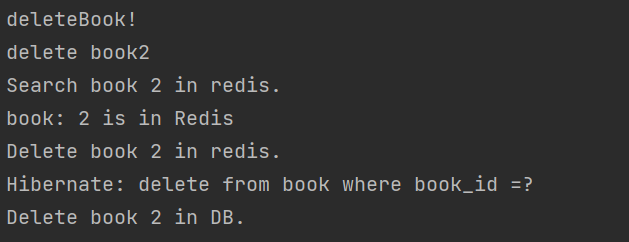
显然，此时redis为空，先对redis进行搜索，没有搜索到对应key的object，选择去数据库中获取，并把它写入redis

2.再次读取getbook 1时



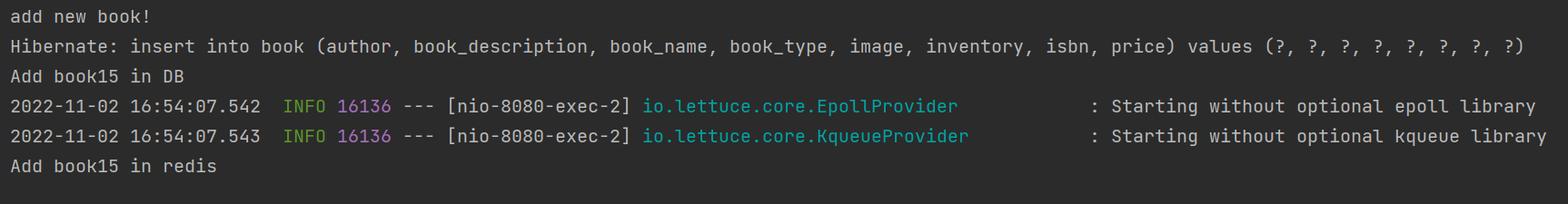
此时，redis中已经存了key为book1对应的value，在对redis进行搜索时发现存在对象，直接返回即可

3.在运行deleteBook2前已经事先将book2 get到redis中，运行得到：



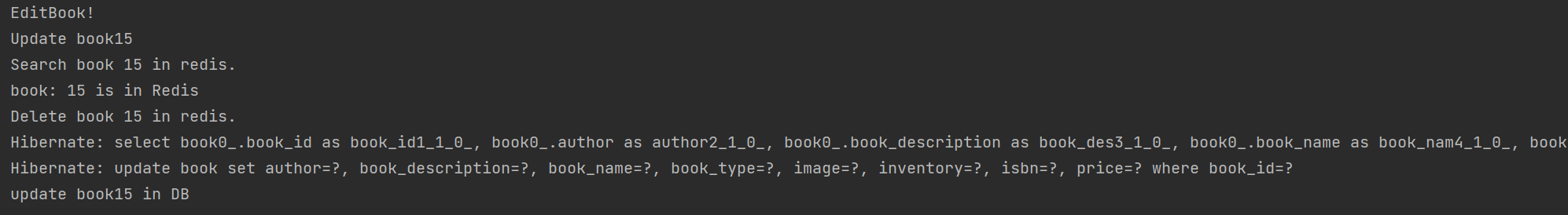
deleteBook时先搜索redis，如果有则删除对象，没有就不用操作，最后统一删除db中的book

4.在addBook命令时



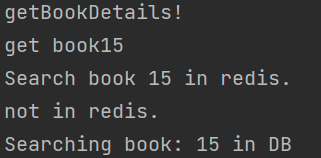
先生成book对象，加入数据库获得自增的bookid后加入redis中

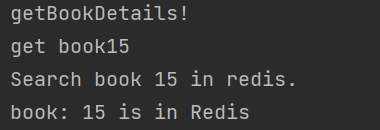
5.EditBook15



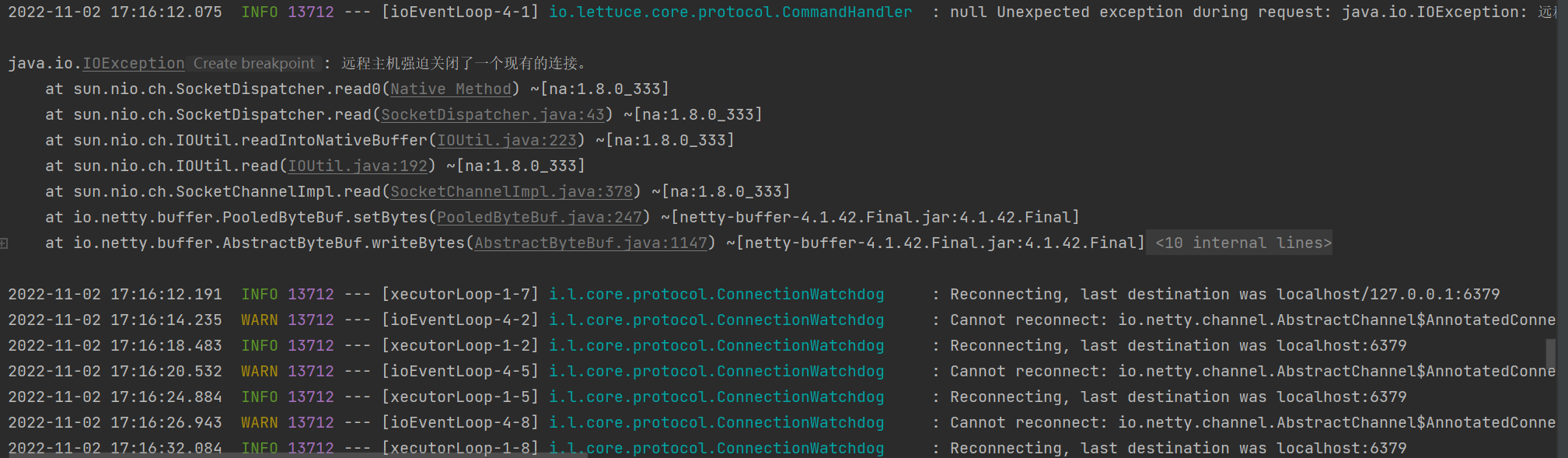
在修改时先在redis和db中查找并获得对应的book信息，如果redis中有就不用找db了。同时如果在redis中找到对应book之后直接删除Redis中该条book信息，在db中对对应的book信息进行更新。这样写的逻辑是为了使更新信息之后再次Get需要redis从db中读取信息，保证了redis和db中数据信息的一致性。如果需要同步更新的话，也可以先在redis中搜索，如果有对应的book信息，则先删除再加入更新后的信息，如果没有，则直接在redis中加入更新后的信息。

再次进行两次getBook15的输出如下





6.人为关闭redis后



由于redis通过socket连接方式和server端连接，redis关闭后，连接端开，server端会每隔一段时间继续向redis原先运行的端口发送连接请求。

在此时发送editbook15请求，后端由于连接不到redis，导致连接时间超过设定的300ms，报错。

