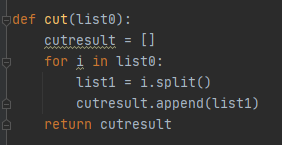
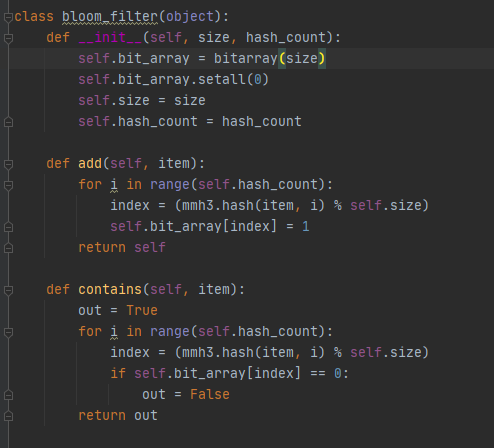
1、布隆过滤器算法实现关键词的查询功能

算法核心：将文本切片为单词后，将每个文件的分词add进bloom过滤器生成该文章对应的布隆过滤器.然后通过if A in B 来判断是否在该文件对应的布隆过滤器中。

切片函数，主要是将句子拆分成单词。



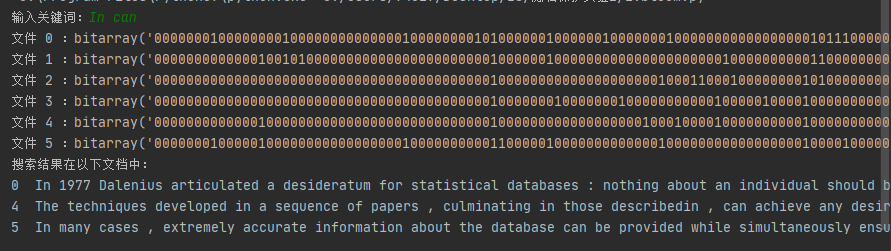
布隆过滤器算法主体，add建立hash表，contains为将查询单词和hash表的hash值进行对比。



布隆过滤器调用，对比两个hash表判断结果。



输入In can的输出结果



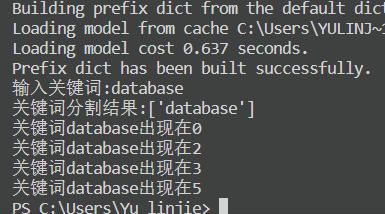
2、布隆过滤器库使用

算法核心：首先通过split和jieba对文档进行切片,然后生成n个bloom实例(n为文件数量),然后将每个文件的分词add进bloom过滤器生成该文章对应的布隆过滤器.

然后通过if A in B 来判断是否在该文件对应的布隆过滤器中

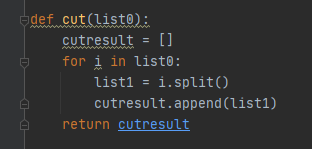


运行结果

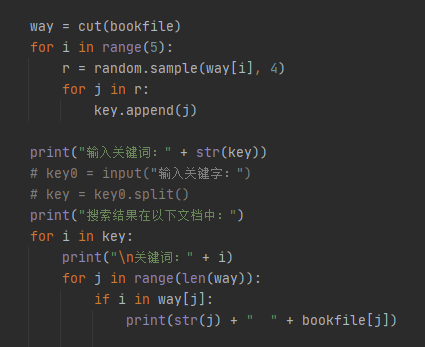


3、倒排索引算法实现

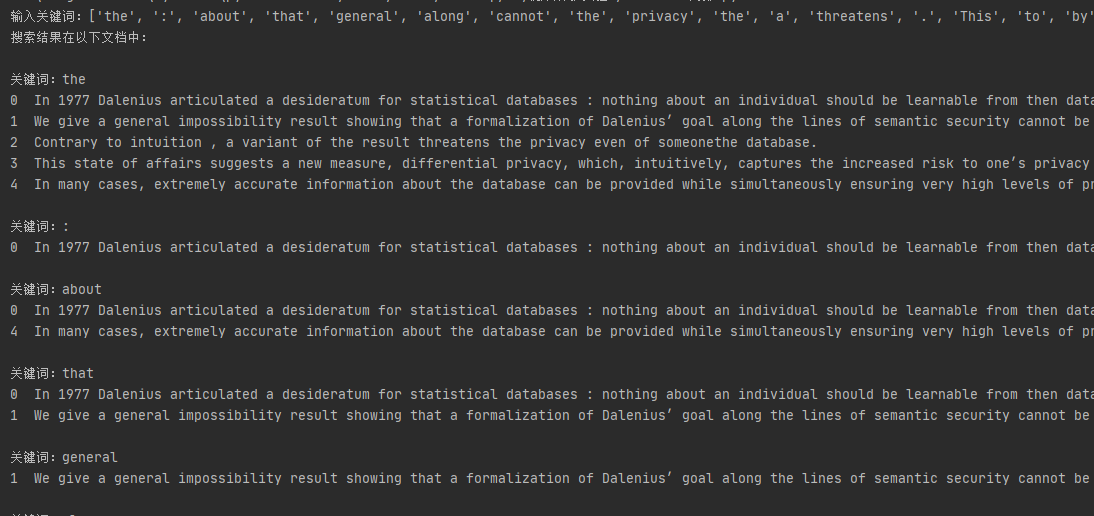
切片函数，主要是将句子拆分成单词



倒排索引算法，本次实验中为直接随机生成需要查询的单词



输出结果：

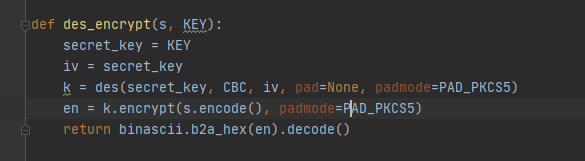


4、密文检索

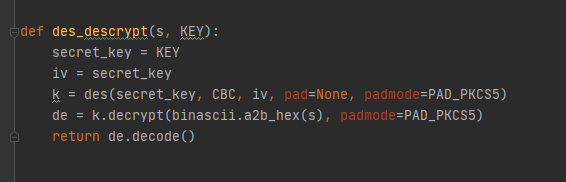
算法核心：

1. 服务端根据用户提供的密钥对文档中的关键词进行DES加密
2. 加密的同时将加密后密文放入bloom\_filter进行多次hash运算后生成该文件对应的Bloom bitarray。此时需要保存文件索引与文件bloom\_filter的键值对
3. 用户输入DES加密的关键词数据到服务器
4. 服务器生成该关 键词对应的Bloom bitarray与已有的文件Bloom bitarray进行按位与操作,如果与原来的一致则可认定该关键在此文件中
5. 将所有关键词对应的文件索引取交集后返回给用户

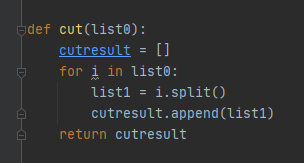
Des算法加密



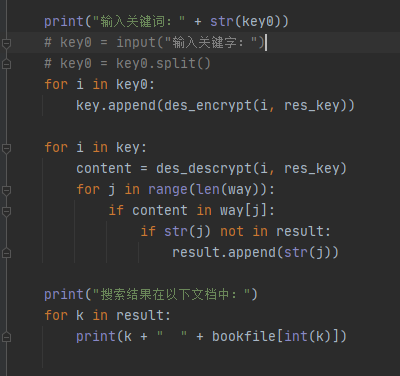
Des算法解密



切片函数



主函数部分



输出结果：

