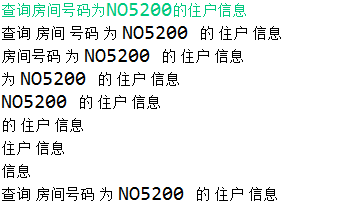
自然语言查询接口实现算法，是可独立于特定数据库的，这里采用企业进销存管理系统做示例展示。

算法分为3大部分：分词 词性匹配 SQL语句构造

**分词部分：**

①使用开源的分词工具，只需导入相应的jar包，学习分词方法，即可完成。但是为了确保分词的准确性，必须采取循环多次分词的方法（由于分词工具有局限性）。举个例子：查询语句“查询房间号码为NO5200的住户信息”。如果不采取循环多次分词，“房间”和“号码”会分开作为两个词，这显然是不符合语义的。

采取循环多次分词的意思是：每一次只取句子分词结果的第一个词，剩下的词在重构成新的句子继续执行分词算法，直到分完为止。截图展示：

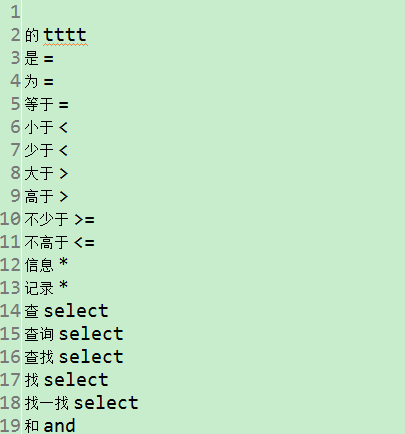


②为了使分词结果更准确，我们需要根据数据库系统构造相应的词典。词典应当包含查询语句所有可能出现的词。保存在dict.txt文件内

**词性匹配部分：**

①对上一步得到的词，需要匹配对应的属性（数据库表的列名，但是不全是）。

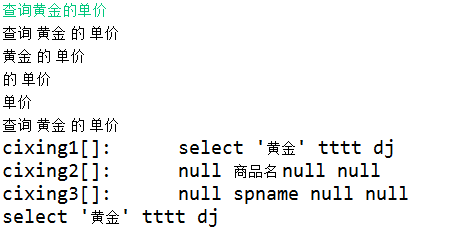
②词典截图dict.txt（包含词和词性）



**SQL语句构造部分：**

①算法模拟人构造SQL语句的思路，根据所有的属性，分别确定select、from、where关键词后的语句。

②对于每个正常的查询语句，我们要求在必要的地方包含结构助词“的”（对应的属性tttt）。因为根据它可以方便断句。举个例子：“查询黄金的单价”。分词和词性匹配后截图展示：



这样根据判断“select”和“tttt”前后的属性，就可以确定跟在select后的属性是dj（单价），跟在where后面的语句是tb\_spinfo.spname='黄金'。

③接下里需要根据所有获得的有效属性来确定表。这里包括dj、’黄金’两个，可确定tb\_kc（库存表）和sp\_info（商品信息表），则跟在from后的语句是tb\_kc natural join tb\_spinfo，采用自然连接可以避免很多问题。

④表之间的关联（主键与外键）。这里出现了两个表tb\_kc（库存表）和sp\_info（商品信息表），所以需要在where后面加上tb\_kc.id = tb\_spinfo.id 。

⑤为了方便观察查询结果，我还把'黄金'对应的属性tb\_spinfo.spname加入到了select关键词后。

⑥最后构造的SQL语句：select tb\_spinfo.spname,dj from tb\_kc natural join tb\_spinfo where tb\_kc.id = tb\_spinfo.id and tb\_spinfo.spname='黄金'

**作者总结：**

上面只展示了一个简单的示例，更多查询语句的其它细节不在展示。