1. 思路

（1）读取源文件，用bufferedreader一行一行的读取，返回String[]供词法分析使用（方便报错）

（2）词法分析，在词法分析中，将读取源文件返回的string数组一行一行进行分析，依次读取字符，进行判断，如果遇到-或者数字，则进入数字DFA的函数解析数字；如果遇到t或者f或者n，则可能分别对应true，false或者null，跳入相应的DFA来解析特殊字符；如果遇到”，则跳入字符串DFA进行处理，最终结果产生一个token序列。

token新建一个类，存放token的value（string），type（一个枚举类型）

错误新建一个类，用来显示是词法分析的类，输出对应行列号

（3）语法分析，对词法分析得到的token序列进行解析，依次读入token，输出的为object或者array对象。

首先建立json抽象类，扩展出jsonobject类，jsonarray类，jsonvalue类，用来存取解析出来的object，array和其他类型，其中jsonobject内部是一个hashmap，由string对象，json对象键值对构成；jsonarray由arraylist构成，由json对象构成；jsonvalue由string对象（值），objecttype对象（一个枚举类，存放类型）。

然后建立迭代器类，用来迭代词法分析得到的token序列。

建立语法分析错误类，用来显示语法分析过程中的错误。

然后建立语法分析类，作为语法分析的入口，如果第一个token为{，则解析object，如果为[，则解析array，否则报错。解析object时，依次读入三个token，如果符合string：xxx的格式，则存入jsonobject对象中；解析array时，依次读入token，存入jsonarray对象中，另外对逗号进行判断。如果在解析object时遇到[，则解析array，并将返回的jsonarray添加到object对象中，反过来相同。

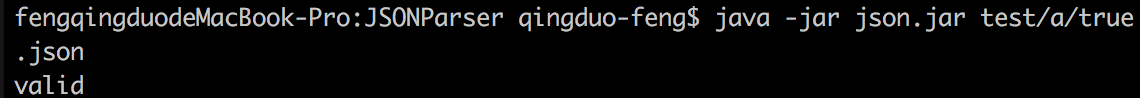
（4）格式化输出

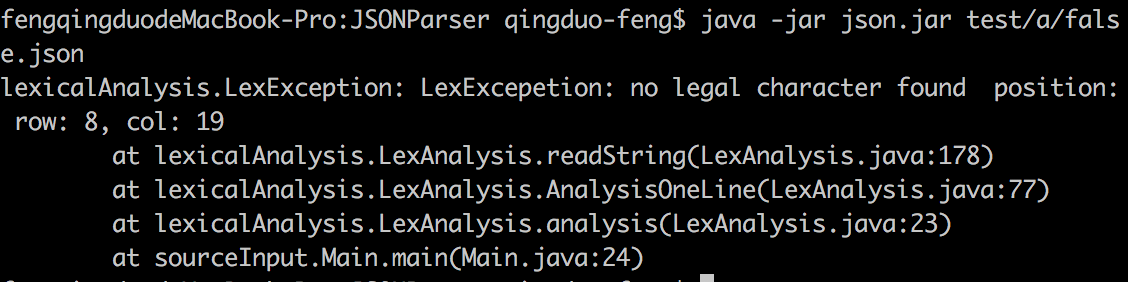
将语法分析得到的object或者array转换为字符串，依次读入单个字符，如果{或者[，则增加缩进，并且输出移至下一行；如果]或者}，则减少缩进，并且输出移至西一行；如果：，则空格；如果，则移至下一行

（5）查询

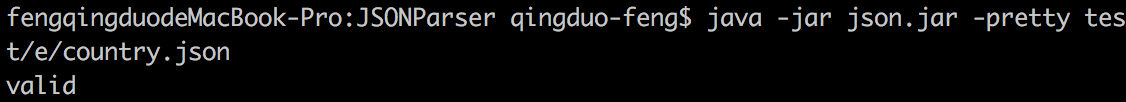
将查询条件分解为字符与数字，字符则查询jsonobject中map中对应的对象；数字则查询jsonarray，找到对应的元素进行返回，直到所有条件查询结束，输出最后的结果。

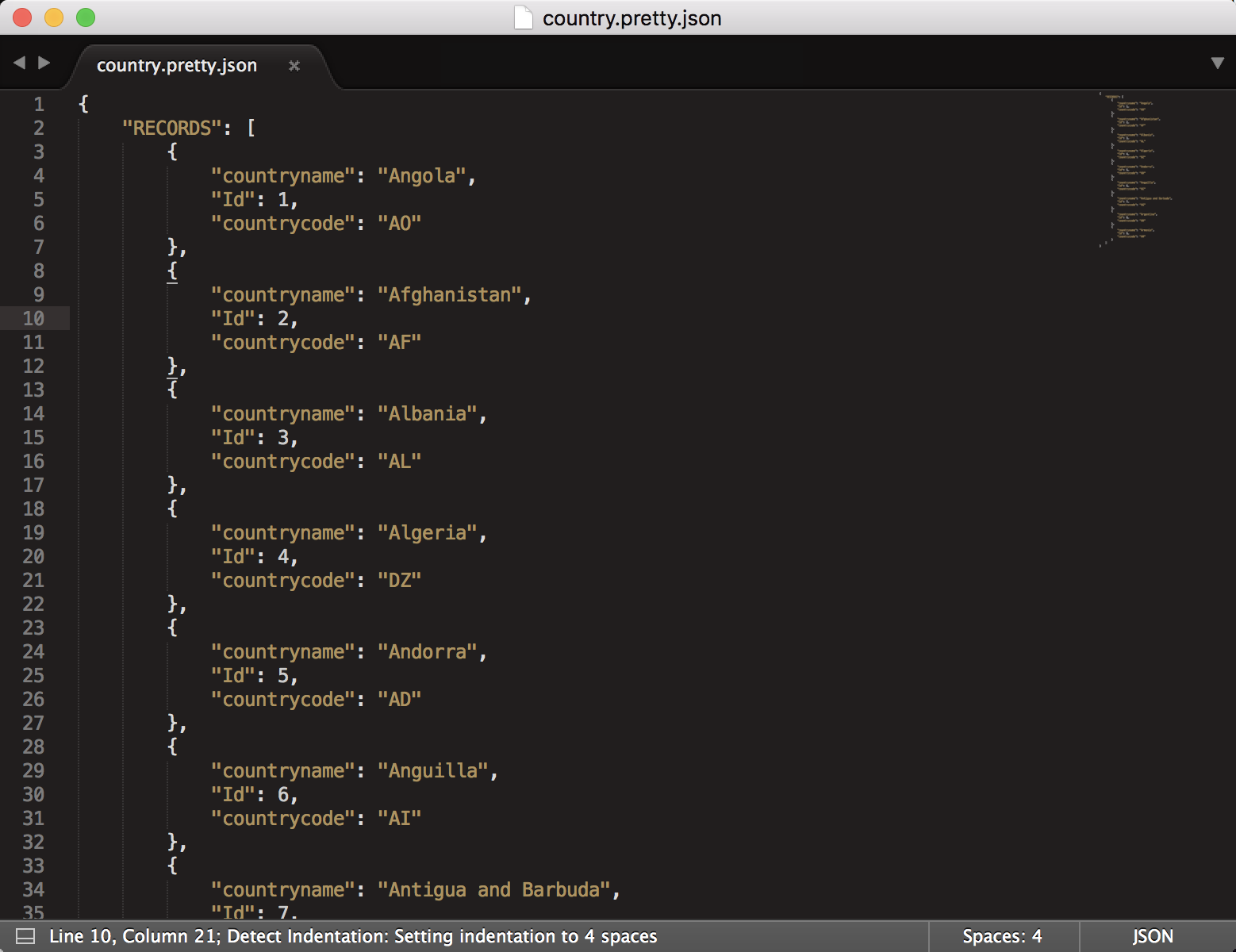
1. 使用说明
2. 解析





1. 格式化输出





（3） 查询

