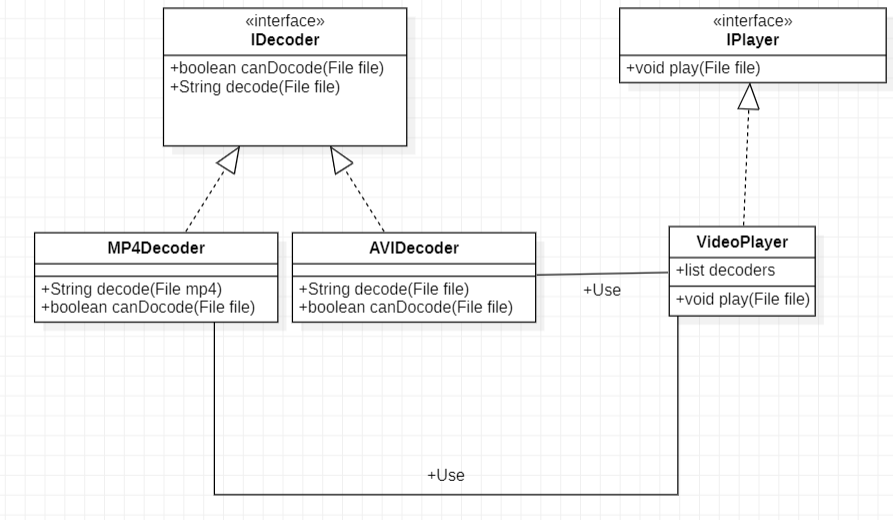
1.

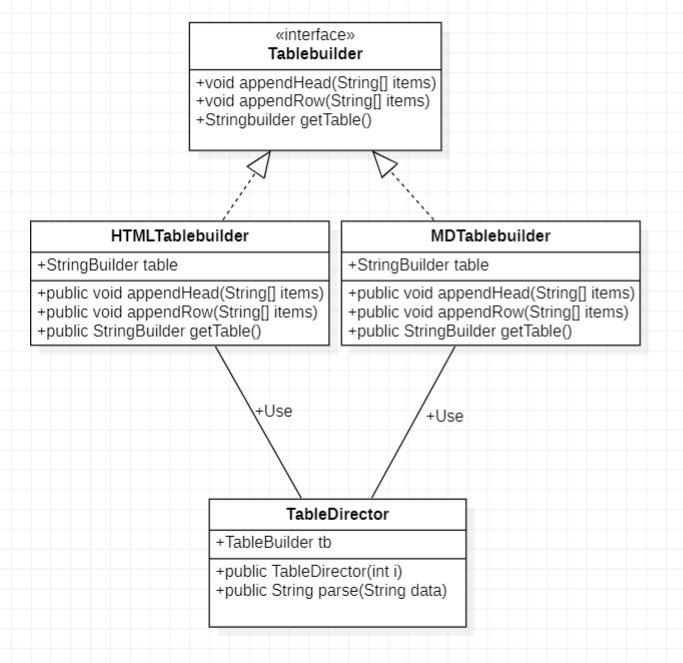
类图:  


代码截图: 

该播放器通过构造器，使得播放器实例化的时候自动加载解码器到播放器对象内部，然后再调用play方法的时候遍历解码器数组，当可以解码的时候进行解码并播放。

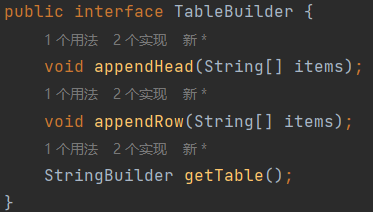
2.

类图：



代码截图：

Tablebuilder接口：



HTMLTablebuilder：

设计逻辑：根据传进来的String数组参数来建造StringBuilder类型的表格table。

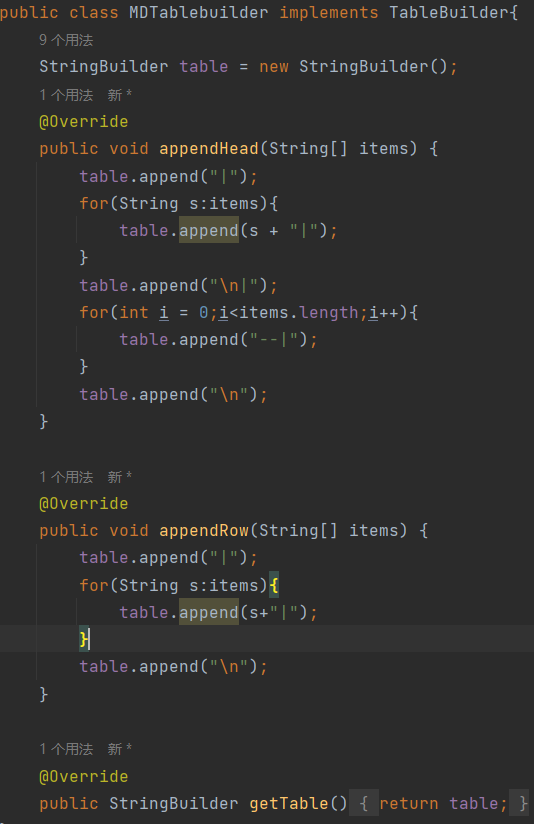
对于表头，现在table的开始位置append<tr>，然后把items中每一项首尾加入<th>和</th>之后插入到table中。

对于表格也是类似的操作方式。



MDTableBuilder类：

与HTMLTableBuilder类似，只是格式有所不同。



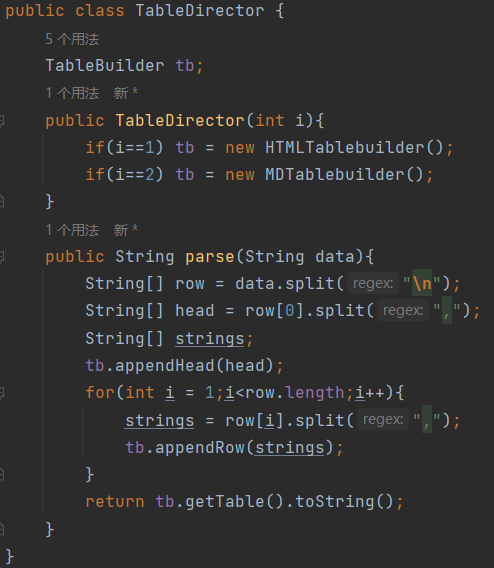
TableDirector类：

首先本类的内部有一个属性，作为创建类的builder，先不进行初始化，在TableDirector类的构造器中根据参数不同来对TableBuilder对象tb初始化，可以初始化为HTML表格建造者也可以是MarkDown的。

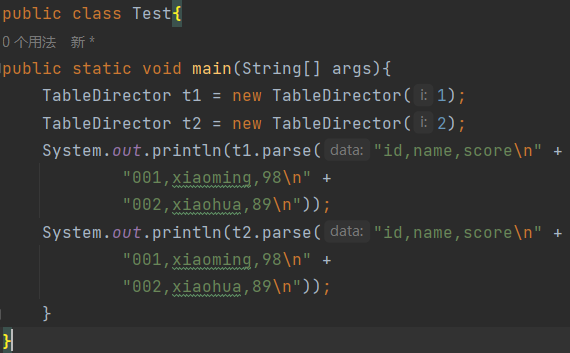
然后是对原始数据的分析函数，对于传进来的String，使用split方法按照换行符号切割，组成字符串数组，字符串数组的第一个元素作为表头，调用appendHead。之后的元素遍历调用appendRow。

需要注意的是每一个字符串数组的元素在调用append的时候需要再进行split切割，这次是按照逗号切割，才能作为新的字符串数组传入append方法中。

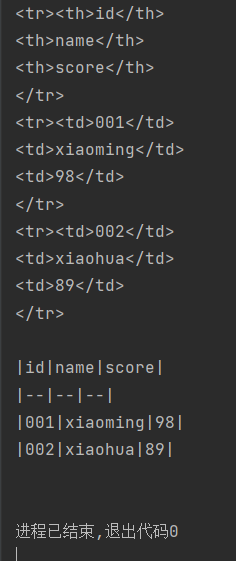
最后调用getTable并且将Stringbuilder使用toString方法转化为String输出。



测试类：

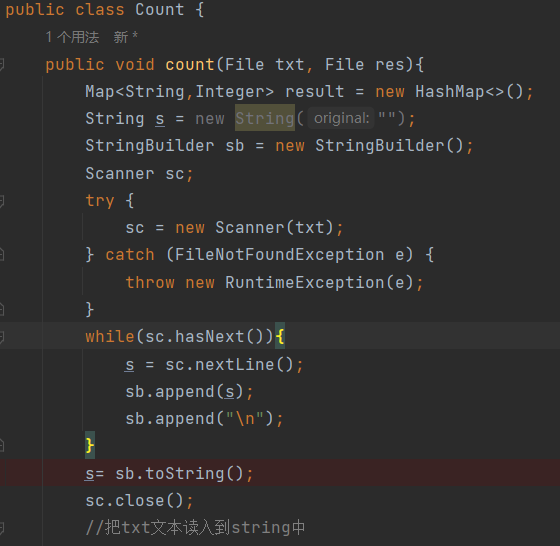


测试结果：



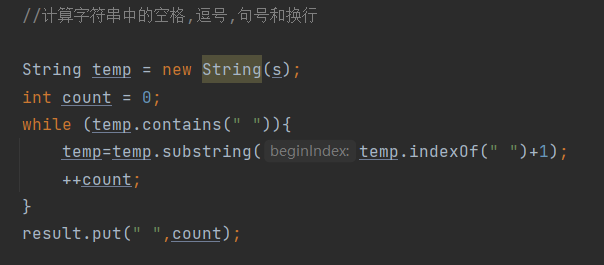
第三题

函数截图:用Map来存储统计的的结果. 用Scanner来读取文件流,每次读一行存到字符串s中,然后加到StringBuilder对象sb的末尾,并加入换行符,读完以后将sb转化为string赋值给s

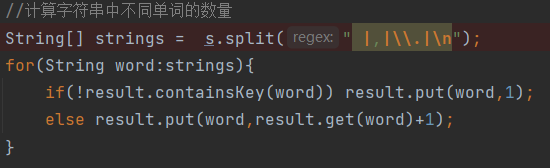


用一种比较笨的方法来计算空格,逗号句号和行数四种特殊的字符.

使用while循环,存在目标字符就执行,将当前字符串从第一个目标字符开始向前截取掉,然后计数+1循环到没有目标字符就停止.



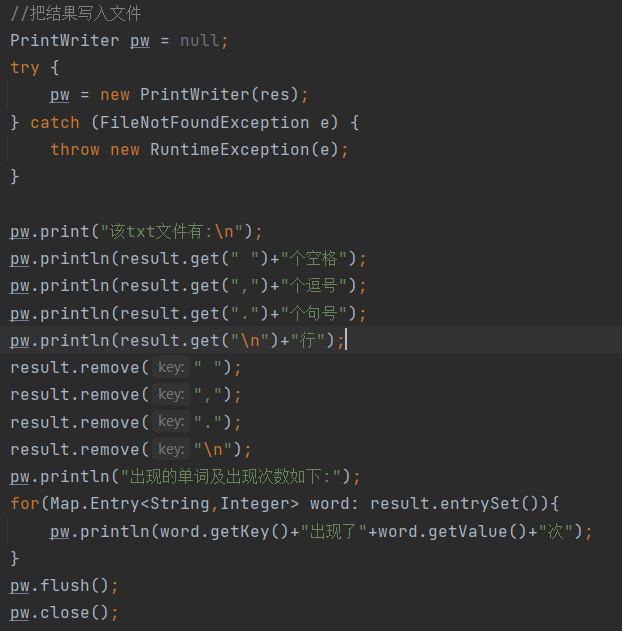
然后是计算各种单词的数量,使用split方法进行分词,把string分成string数组,正则表达式为四个特殊字符析取. 字符串数组提取完毕以后,对其进行遍历,每有新的单词就put进map中,若已存在该单词,就令其value+1



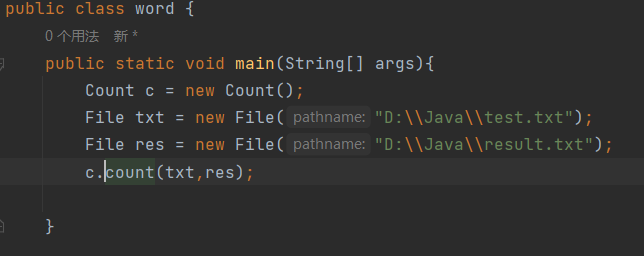
最后使用printwrite类来把result进行格式化输出到目标文件中.

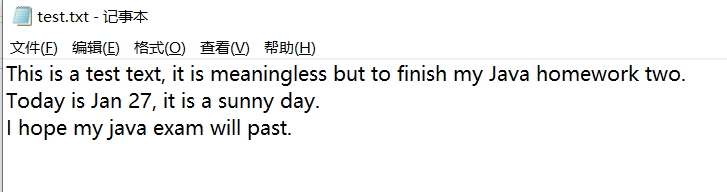
特殊地,这里因为for循环不支持map<String,Integer> 所以使用了map类中的子类Entry.

将result转换为Entry后进行遍历输出.



测试用例截图:





结果截图:

