

به نام خدا

برای انجام این تکلیف ابتدا مانند تمرین قبل سطر به سطر کلمه ها را جدا کرده (token by token) و در یک فایل قرار داده سپس متن آنرا را تمیز کرده (حروف انگلیسی ، اعداد و علامت ها را حذف نموده) و وارد مرحله بعد می شویم.

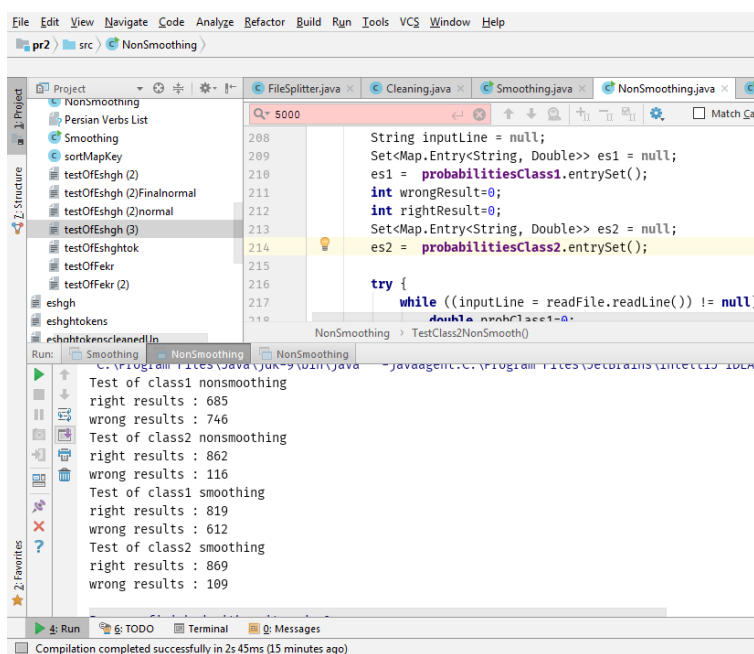
نکته: یکی از کارهای لازم برای تمیز کردن یکسان کردن بعضی حروف است مثلا:

```
temp[m] = temp[m].replace("ی", "ی");  
temp[m] = temp[m].replaceAll("ي", "ی");
```

اکنون در کلاس اصلی لازم است تعداد هر کلمه را شمرده و در یک map بریزیم .(برای روشن شدن چگونه انجام شدن این کار این map را در یک فایل هم چاپ کردیم که قابل ملاحظه است. سپس با استفاده از این فایل حالا با جمع کردن تعداد value های هر map تعداد کل لغات را به دست آورده و ازین طریق احتمال هر لغت را حساب کرده و در map نهایی احتمال میریزیم.

حالا طبق روش نیو بیز فایل های تست را خط به خط ابتدا با روش nonesmoothing و سپس با روش smoothing تست می کنیم. احتمال هر خط تست در هر دو کلاس محاسبه شده هر کدام بیشتر بود در آن کلاس قرار میگیرد.

با ران کردن کلاس NoneSmoothing همه این موارد اجرا می شوند.



The screenshot shows an IDE with the following components:

- Project Structure:** A tree view on the left showing a project named 'NonSmoothing' with sub-projects like 'Smoothing', 'testOfeshgh (2)', 'testOfeshgh (2)Finalnormal', 'testOfeshgh (2)normal', 'testOfeshgh (3)', 'testOfeshghtok', 'testOffekr', 'testOffekr (2)', 'eshgh', 'eshghtokens', and 'eshghtokscleanedIn'.
- Code Editor:** The main window displays Java code for 'NonSmoothing.java'. It includes imports for 'String', 'Set', 'Map', 'Entry', and 'Double'. It defines two sets, 'es1' and 'es2', and a 'try' block with a 'while' loop that reads lines from a file and processes them.
- Run Console:** The bottom panel shows the output of running the tests. It displays results for 'Test of class1 nonsmoothing', 'Test of class2 nonsmoothing', 'Test of class1 smoothing', and 'Test of class2 smoothing'. Each test shows 'right results' and 'wrong results' counts.
- Status Bar:** At the bottom, it indicates 'Compilation completed successfully in 2s 45ms (15 minutes ago)'.

در حالت noneSmoothing:

Class1:

Pre: 47%

Rec: 85%

در این حالت چون تعداد کلمات جدید در فایل تست زیاد است prec کم است.

Class2:

Pre : 88%

Rec : 53%

در حالت smoothing:

Class1:

Prec: 57%

Rec : 88%

Class2 :

Prec : 89%

Rec: 58%