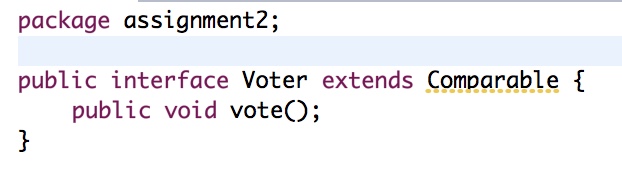
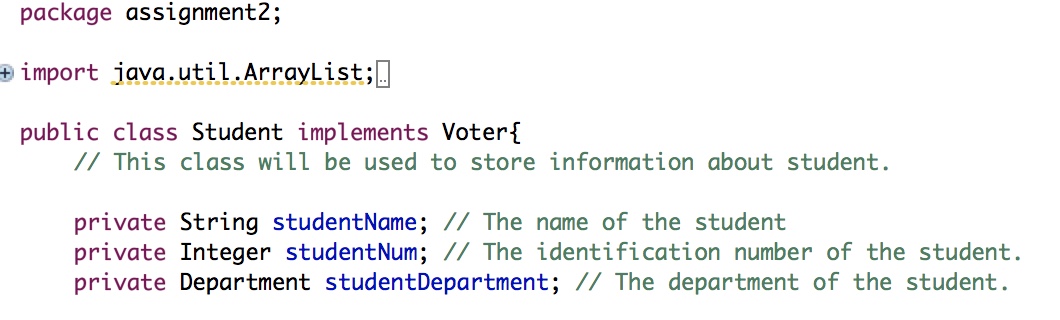
**2017029716 박혜정 Assignment2 Read me 파일**

1. **Voter interface**

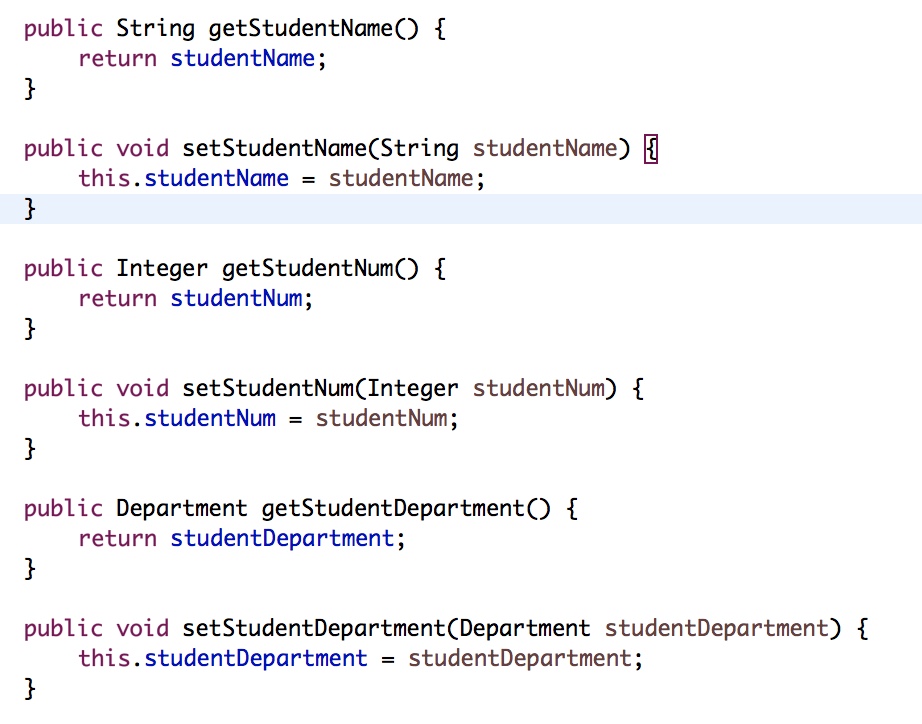


Voter interface를 만들어준다.

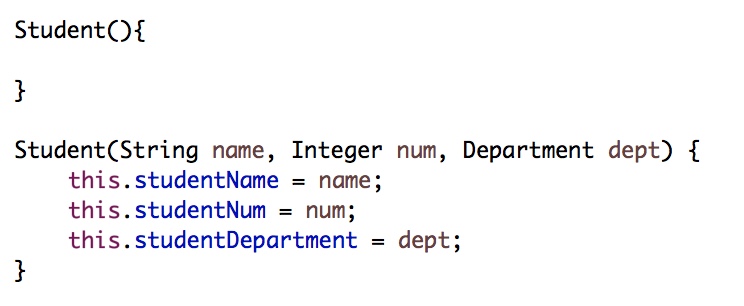
1. **Student Class**



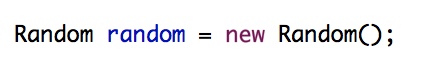
Voter interface를 implements한 Student class를 만들어준다.

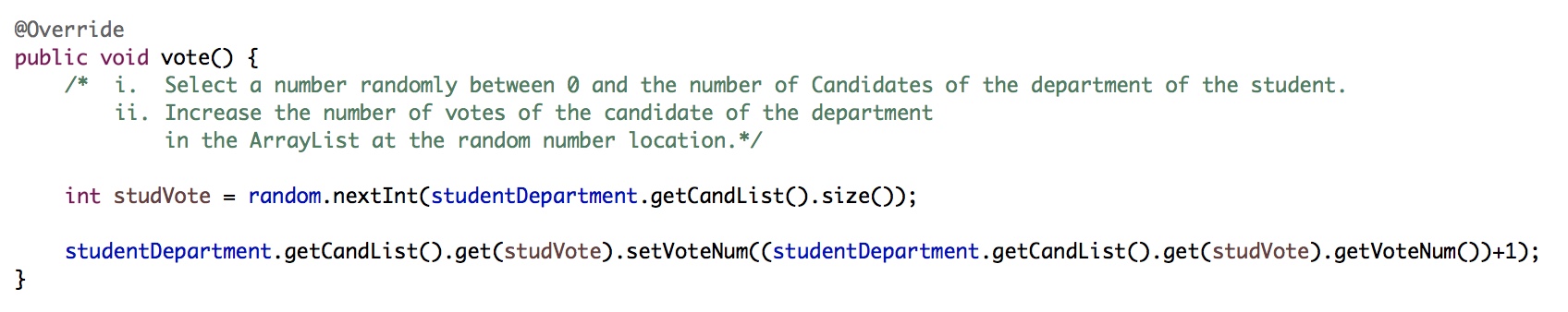


변수를 모두 private으로 선언하였기 때문에, getter, setter를 만들어준다.



생성자를 만들어 준다.

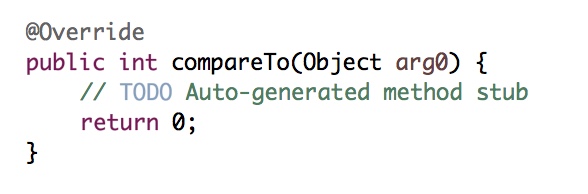




Random을 만들어주고,

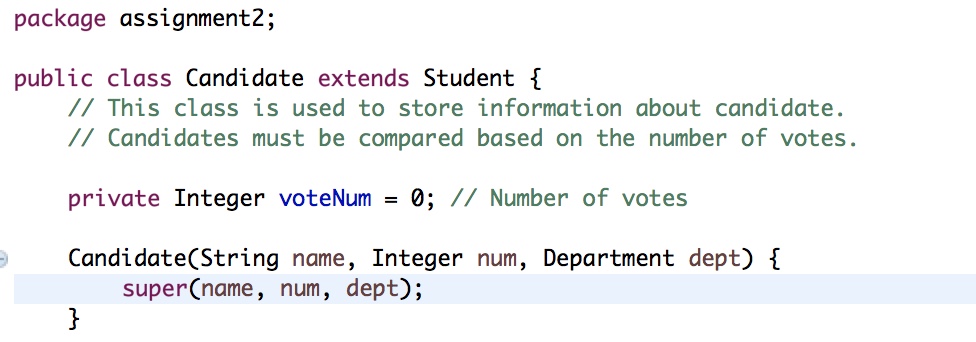
Candidate의 수만큼의 보기를 만들기 위해 Candidate List의 size를 가져와 random.nextInt()안에 넣어주었다.

그리고 표를 받은 Candidate를 불러와서 받은 표의 수를 1 올려주었다.

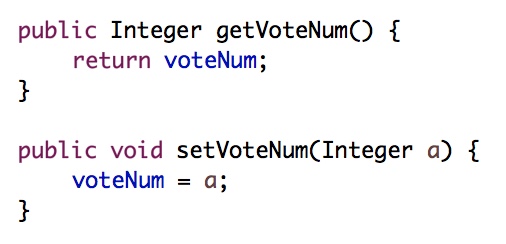


Implements한 Voter interface가 Caparable interface를 implements 했으므로, compareTo()를 정의해준다.

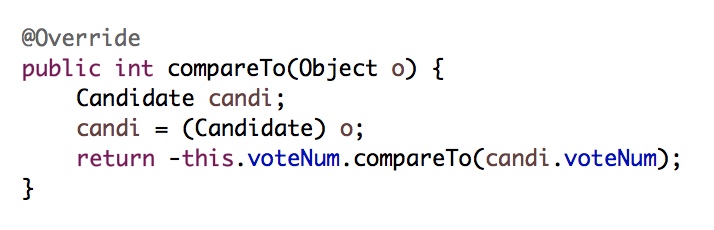
1. **Candidate class**



Candidate class와 instance variable, 생성자를 만들어준다.

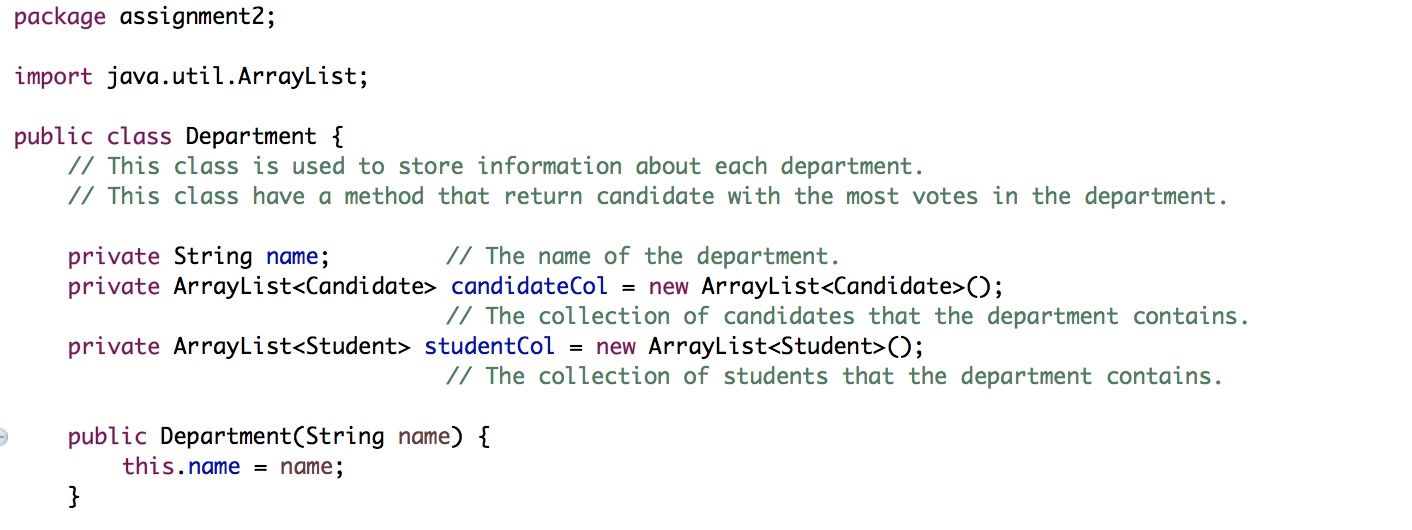


voteNum을 private으로 만들었으므로 getter와 setter를 만들어준다.

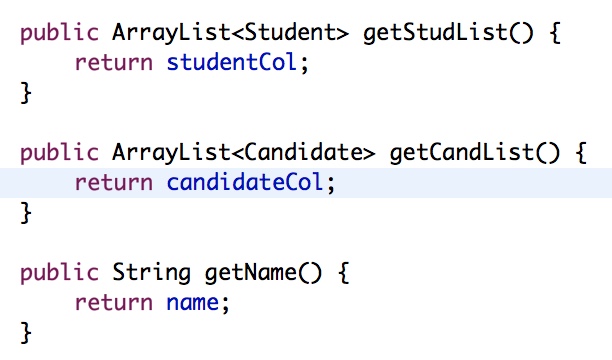


compareTo()를 득표수로 비교하여 내림차순으로 하도록 만들었습니다.

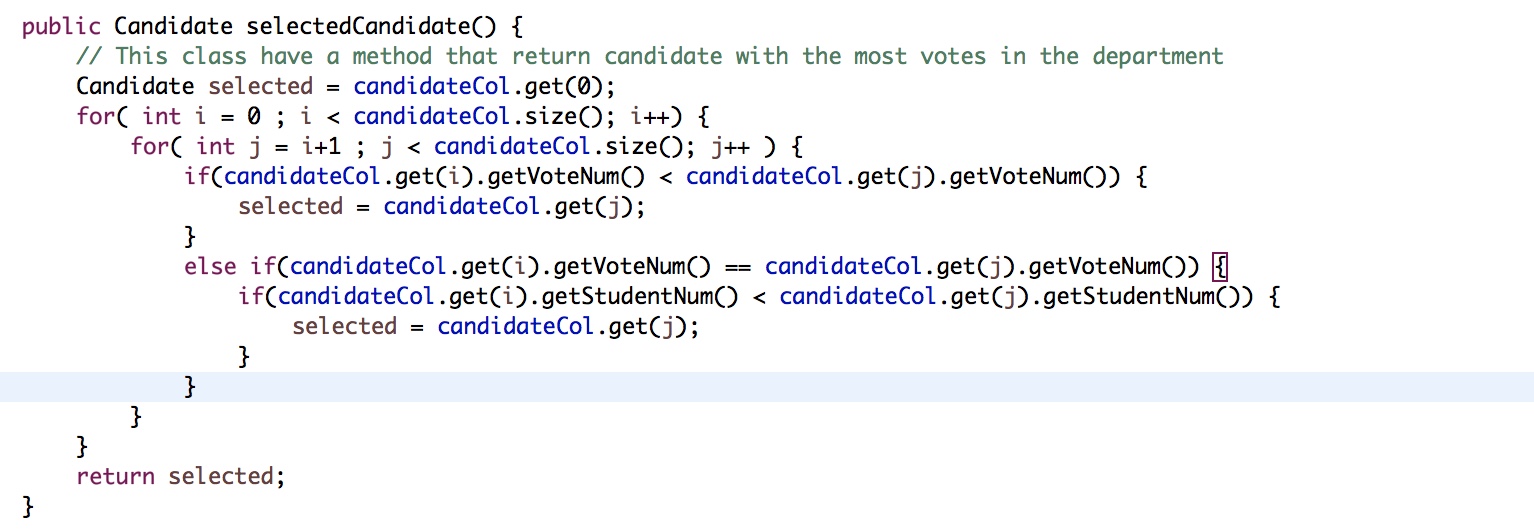
1. **Department class**



Department classd와 instance variable, 생성자를 만들었습니다.



Instance variable을 모두 private으로 만들었으므로 getter를 만들었습니다.

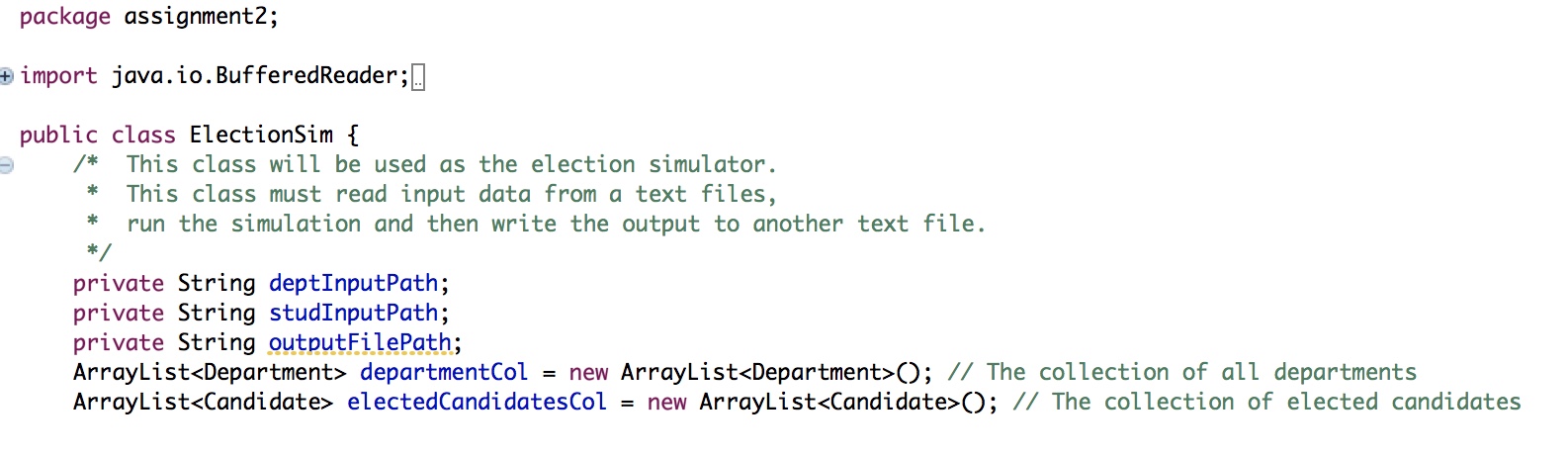


선출된 후보자를 반환하기 위해,

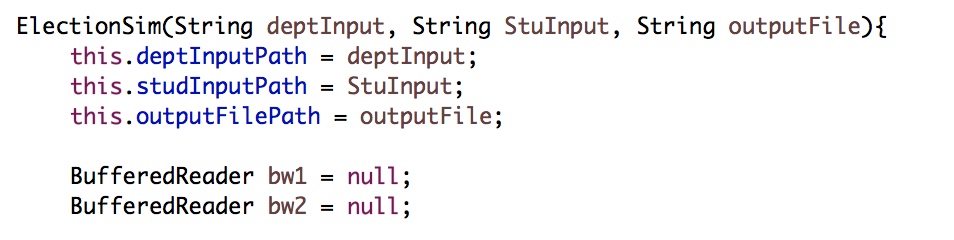
두 후보자의 득표수를 비교하여 더 많이 득표한 사람을 selected에 넣어, selected를 반환하는 방식으로 만들었습니다.

동점자의 경우 학번이 큰 사람이 선출됩니다.

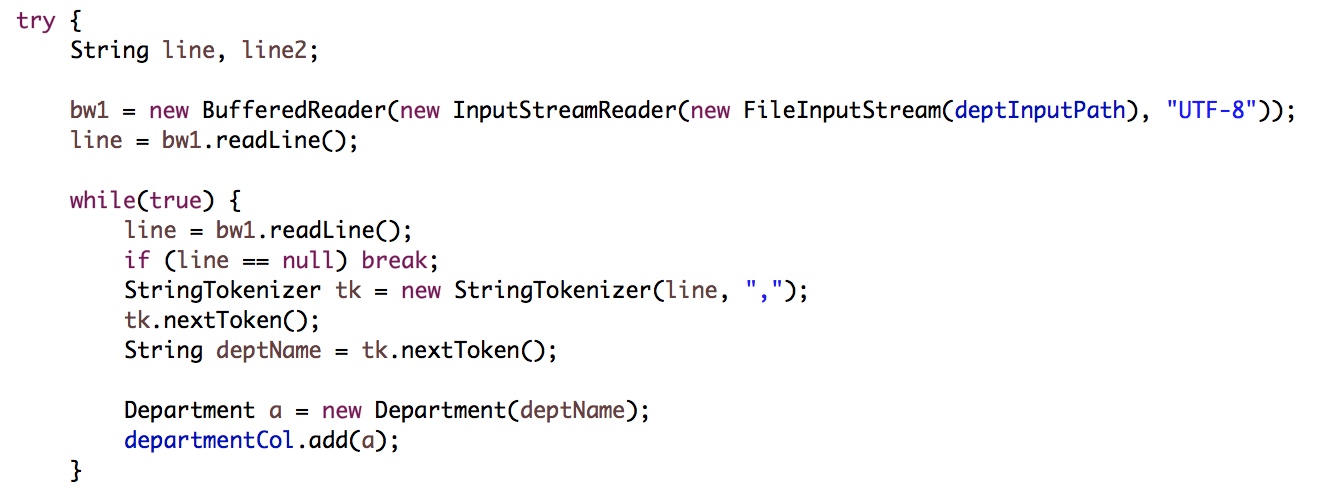
1. **ElectionSim class**



ElectionSim class를 만들어주고, instance variable들을 선언해주었습니다.



생성자를 만들어 줍니다.



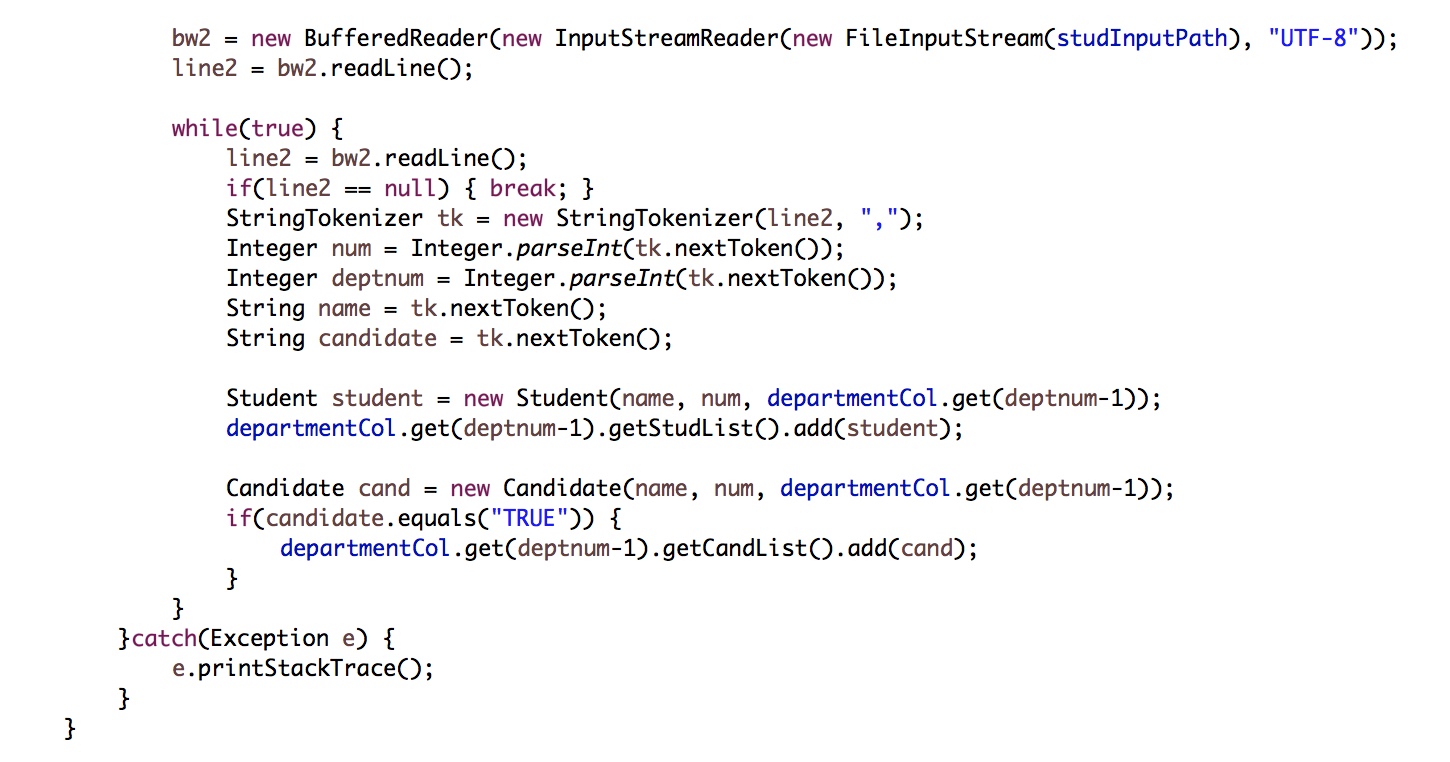
생성자 내에서 try-catch 문을 활용합니다.

try문 안에서 input1.csv파일을 읽어옵니다.

BufferedReader bw1을 이용해 파일을 한줄한줄 읽어오고,

StringTokenizer tk를 이용해 항목별로 잘라서 넣어줍니다.

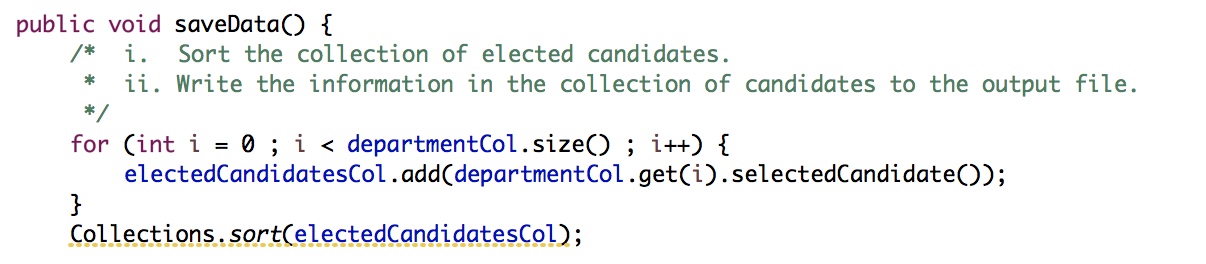
받아온 정보로 Department 객체를 만들어 학과 리스트에 넣어줍니다.



같은 방식으로 input2.csv 파일도 읽어오고,

Student 객체를 만들어 학생리스트에 넣어줍니다.

학생이 후보인지의 여부를 판단하여 Candidate 객체도 만들어 후보자리스트에 넣어줍니다.

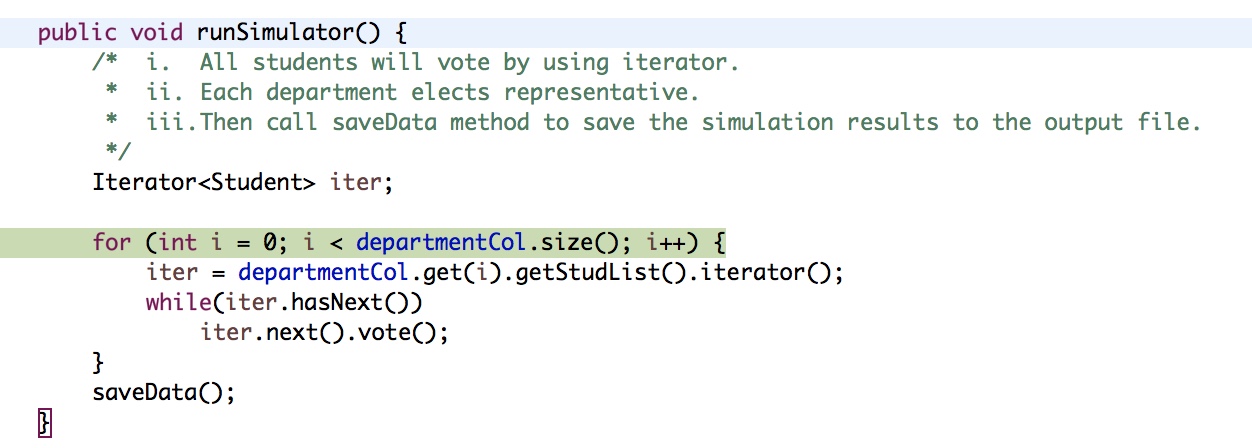


선출된 후보자 리스트에 선출된 사람들을 넣어줍니다.

그리고 득표수를 기준으로 내림차순으로 소팅합니다.

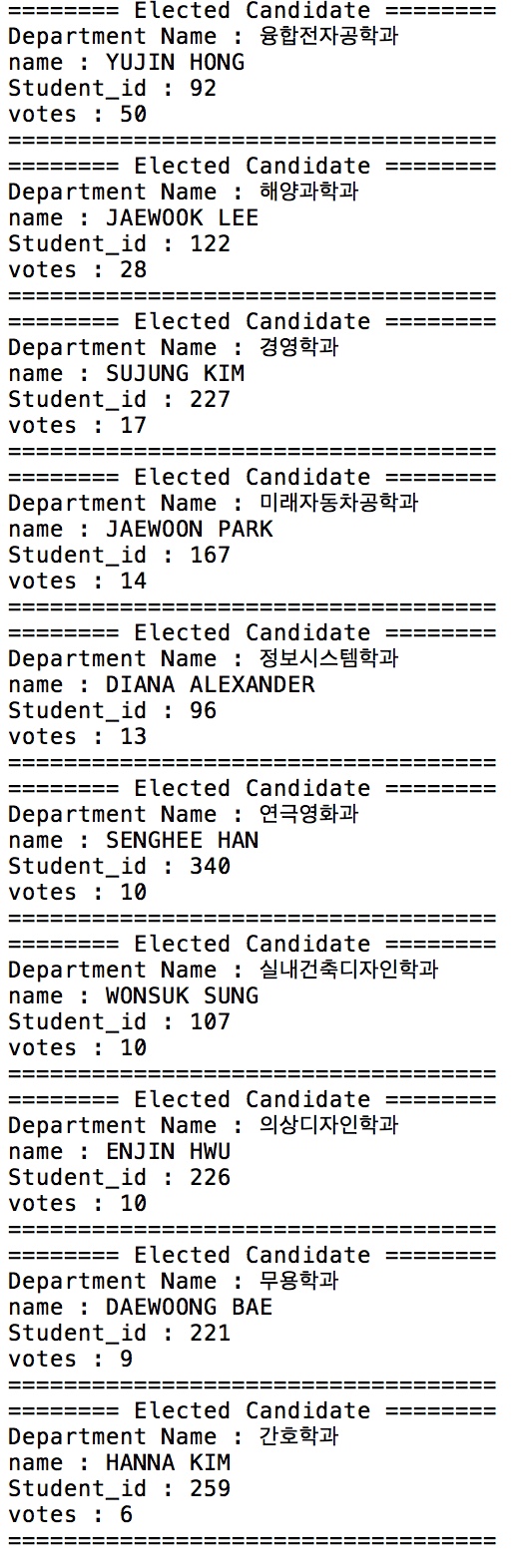
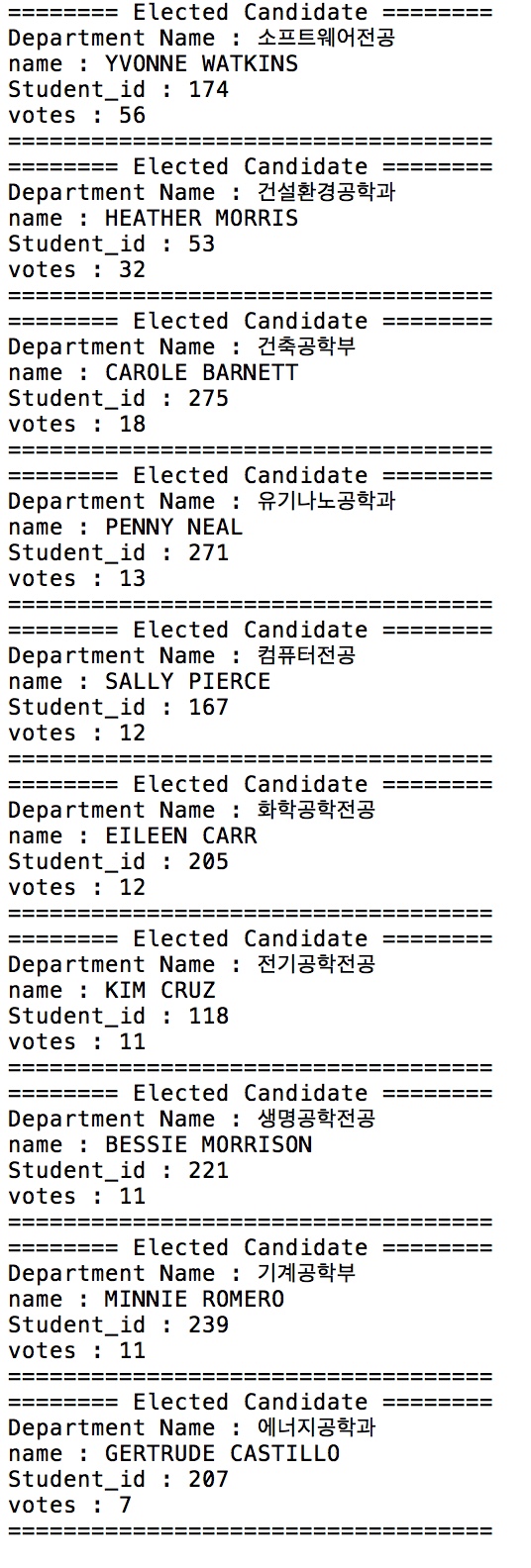


아웃풋 파일을 작성해줍니다.



Iterator을 이용해 학생들이 차례차례 투표하도록 만들어 줍니다.

1. **결과값**

****

1. 원래 인풋의 결과값 2. 내가 만든 인풋의 결과값