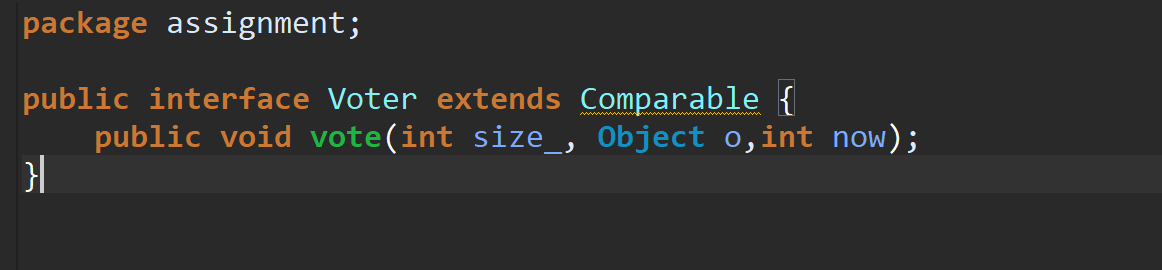
**객체지향언어 Assignmnet\_2 document**

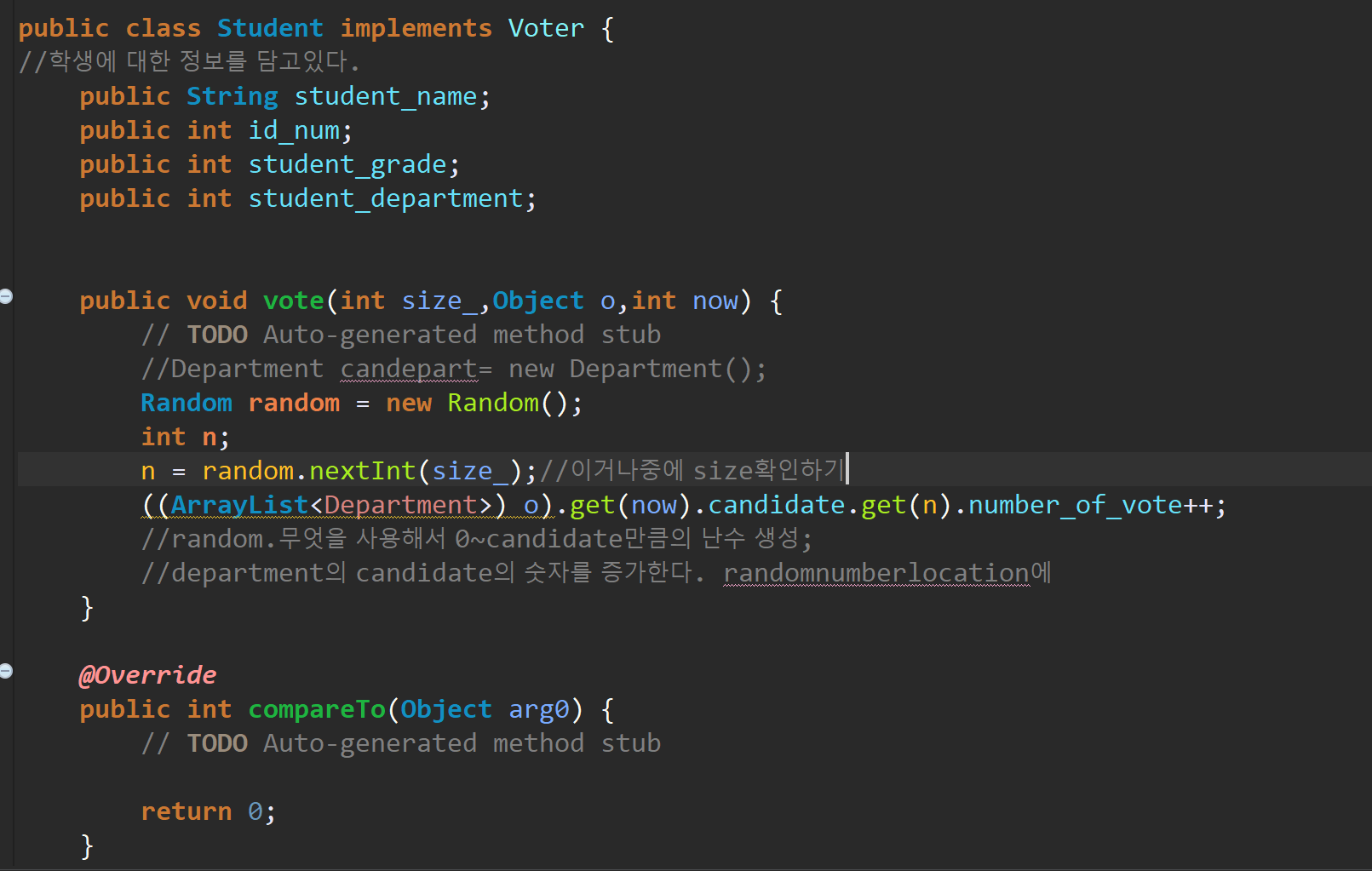
**2017030519 홍유진**

**1. voter interface**



1) Comparable을 상속한 Voter interface를 만들고 vote method를 선언했다.

**2. Student class**



**1) vote method**

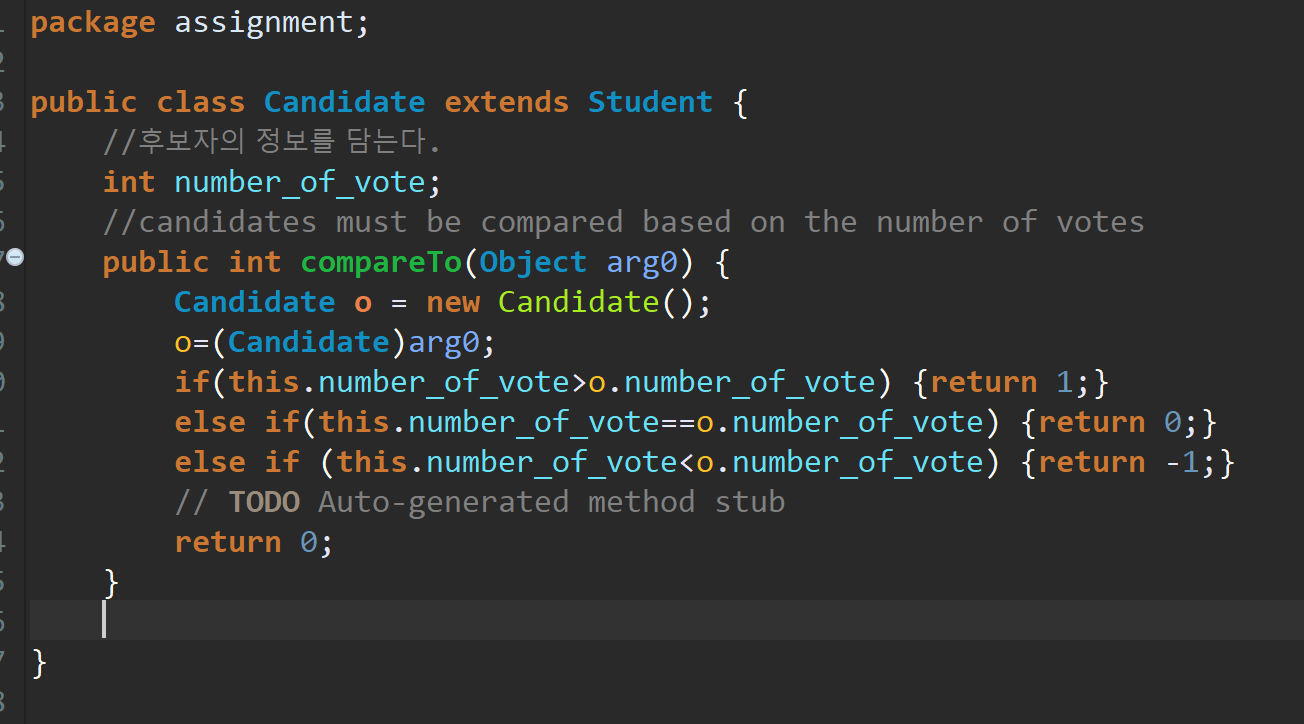
a) department의 candidate의 size와 department 객체와 현재 department값이 어떤 department를 가리키고있는지에 대한 인덱스 값인 now를 함수의 parameter로 받는다.

b) 함수 내에선 0~size\_ 사이의 난수를 생성해 now의 department값에 있는 candidate의 원래있던 number\_of\_vote값에 투표수를 +1해준다.

**2) compareTo method**

a) voter interface에서 구현한 함수를 오버라이드 한다. 단, 함수엔 아무내용을 적지않는다.

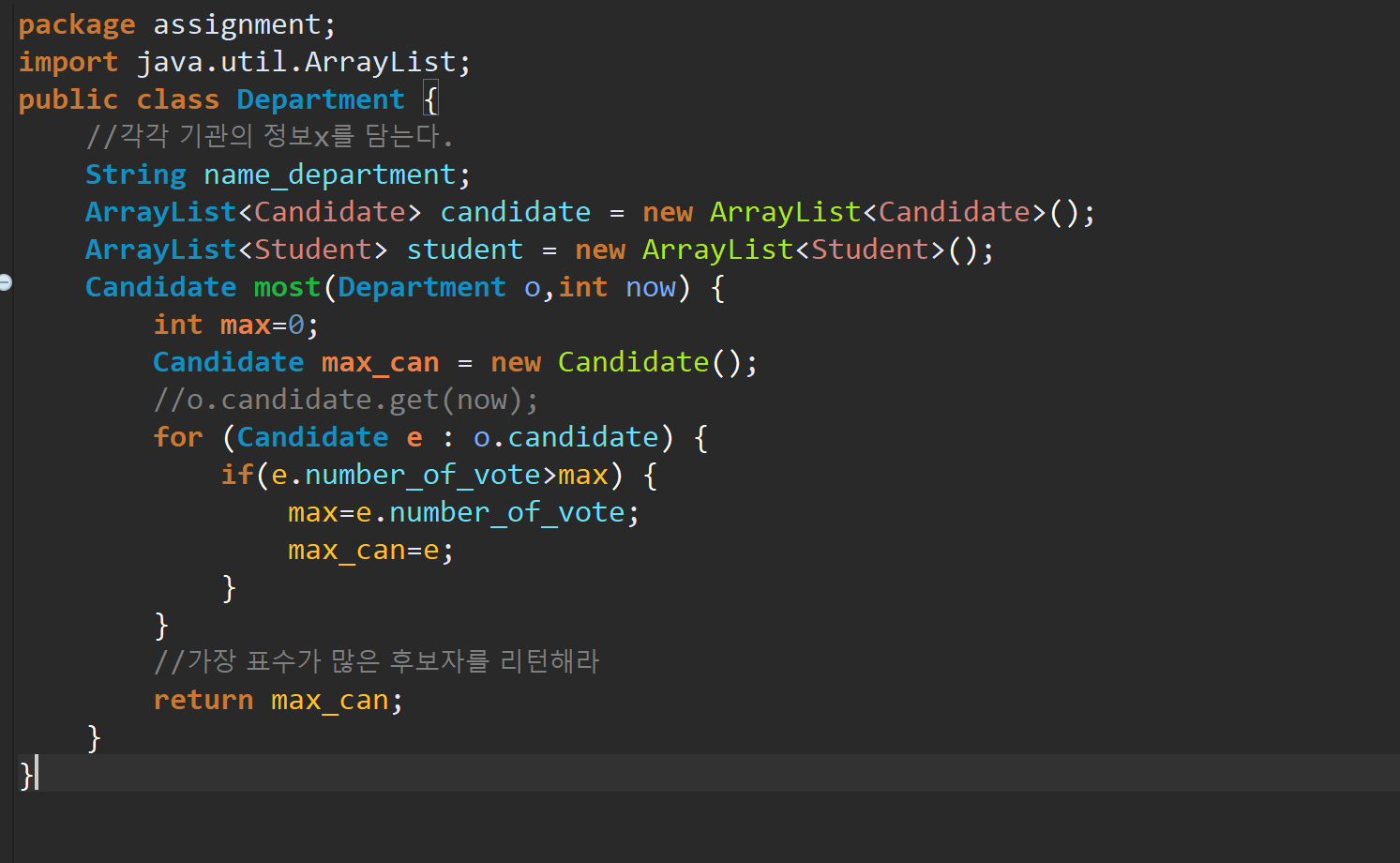
**2. Candidate class**



**1) compareTo method**

: sort를 하기 위해 구현해줬다. Object를 받아 Candidate로 형변환 시켜준 후 this(Candidate)의 number\_of\_vote값과 Object arg0를 형변환한 o의 number\_of\_vote를 비교해 각각 -1,1,0을 return 했다.

**3. Department class**



**1) most method**

: department의 Candidate 중 가장 투표수가 높은 Candidate를 return하는 함수이다.

max알고리즘과 for each문을 통해 Department o의 candidate의 투표수를 비교해줬다.

**4. ElectionSim Class**

**1) ElectionSim의 생성자**



a) 맨 처음 생성자로 받은 departmentInputFilePath의 이름을 가진 파일을 Scanner와 FileInputStream을 통해 읽어온다. FileNotFoundExeption 이라는 예외를 catch해주는 문을 만들었다.

b) file1이 다음문장을 가지고 있을 때까지 stringTokenizer를 이용해 department의 name을 department arraylist에 add한다.

c) String outfilepath 인스턴스변수를 만들어준 이유는 saveData method에서 생성자에서 받은 파일명을 저장하기 위함이다.



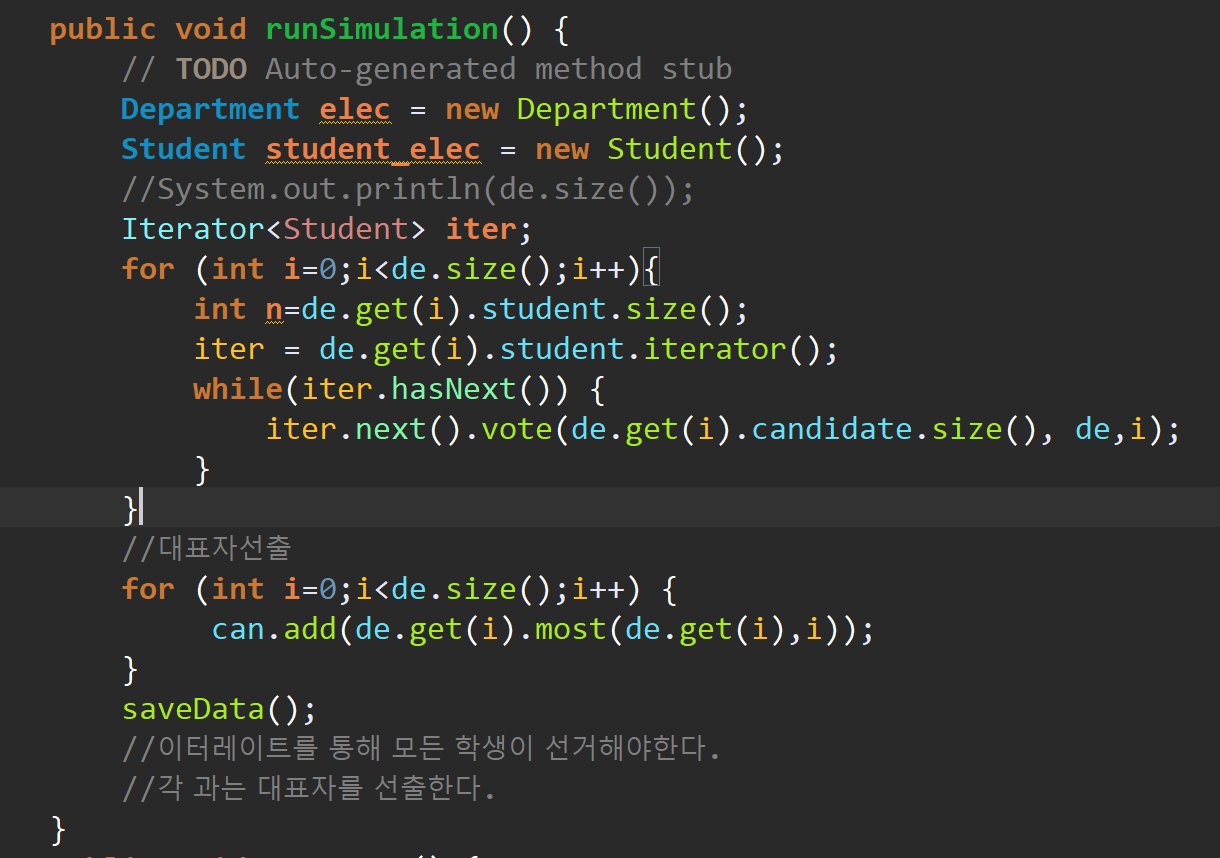
c) STUDENT\_INPUT\_FILE\_PATH의 이름을 가진 파일을 Scanner와 FileInputStream을 통해 읽어온다. FileNotFoundExeption 이라는 예외를 catch해주는 문을 만들었다.

d) Student의 객체 A를 만들고 StringTokenizer를 이용해 student의 인스턴스에 넣어준 후 department의 어레이리스트인de의 student의 department위치에 student어레이리스트에 student객체인 A를를 add해준다. 이때 student\_department의 값에 1을 빼준이유는 arraylist의 인덱스가 1부터 시작하기 때문이다.

e) 만약 candidate가 TRUE이면 (String elected를 통해 확인) Candidate의 새 객체인 canStu를 만들어 정보를 입력한후 de에 student department위치에 candidate어레이리스트에 add해준다.

f) 읽은 파일들을 close()해준다.

**2) runSimulation method**

 a) student의 이터레이터를 생성해준만큼 de의 사이즈만큼 반복하고 각 department의 student 어레이리스트가 존재할동안 반복하며 vote를 실행해준다.

b) Department의 most함수를 통해 대표자를 선출하고 ElectionSim class에 있는 candidate 어레이리스트인 can에 선출된 후보를 add한다.

c) saveData() method를 호출한다.

**3) saveData method**



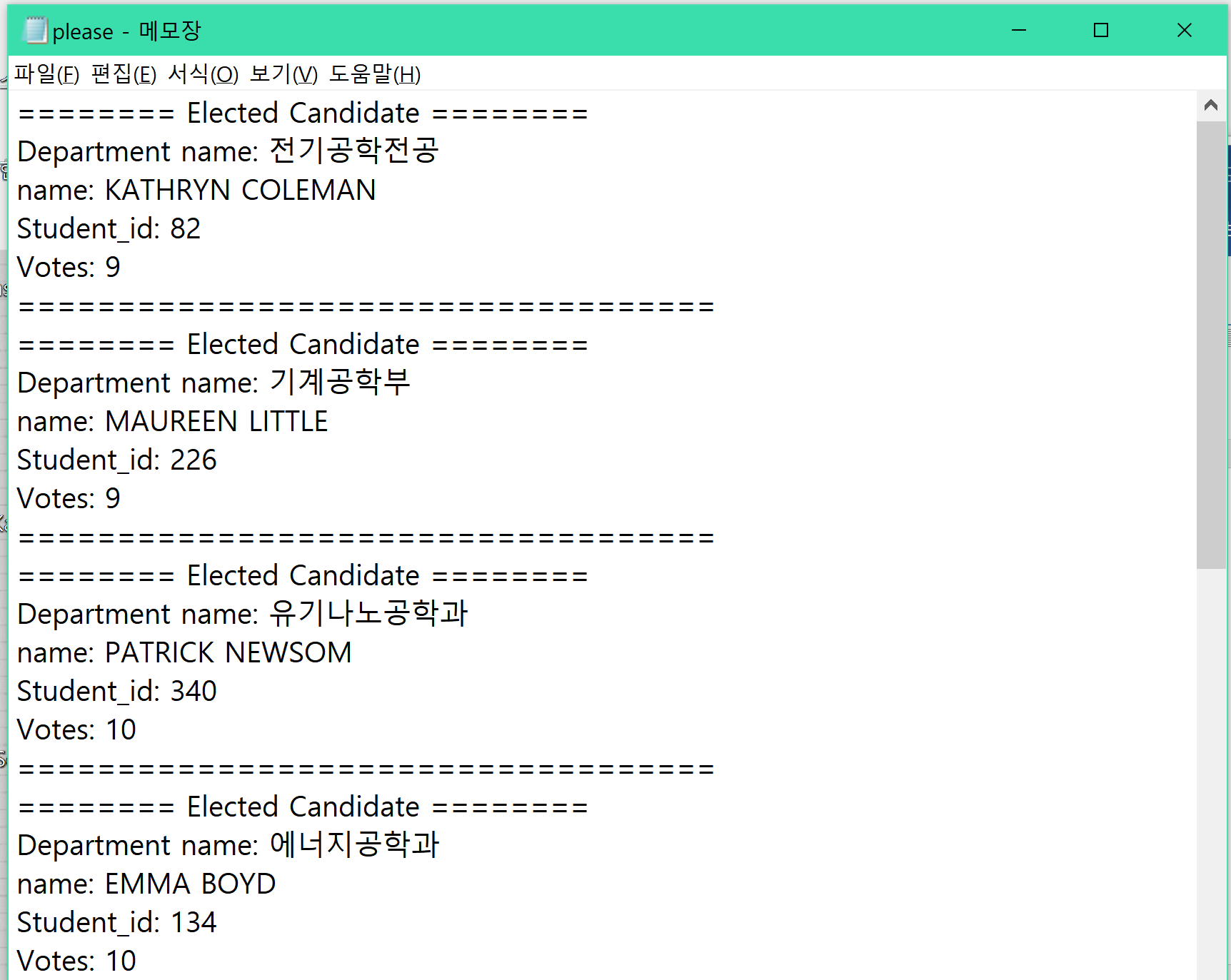
a) PrintWriter와 FileOutputStream을 이용해 쓰기 동작을 한다. FileNotFoundException의 예외를 잡을 catch문을 만든다.

b) Collections.sort(can)을 사용해 어레이리스트 can을 득표수만큼 오름차순 정렬한다.

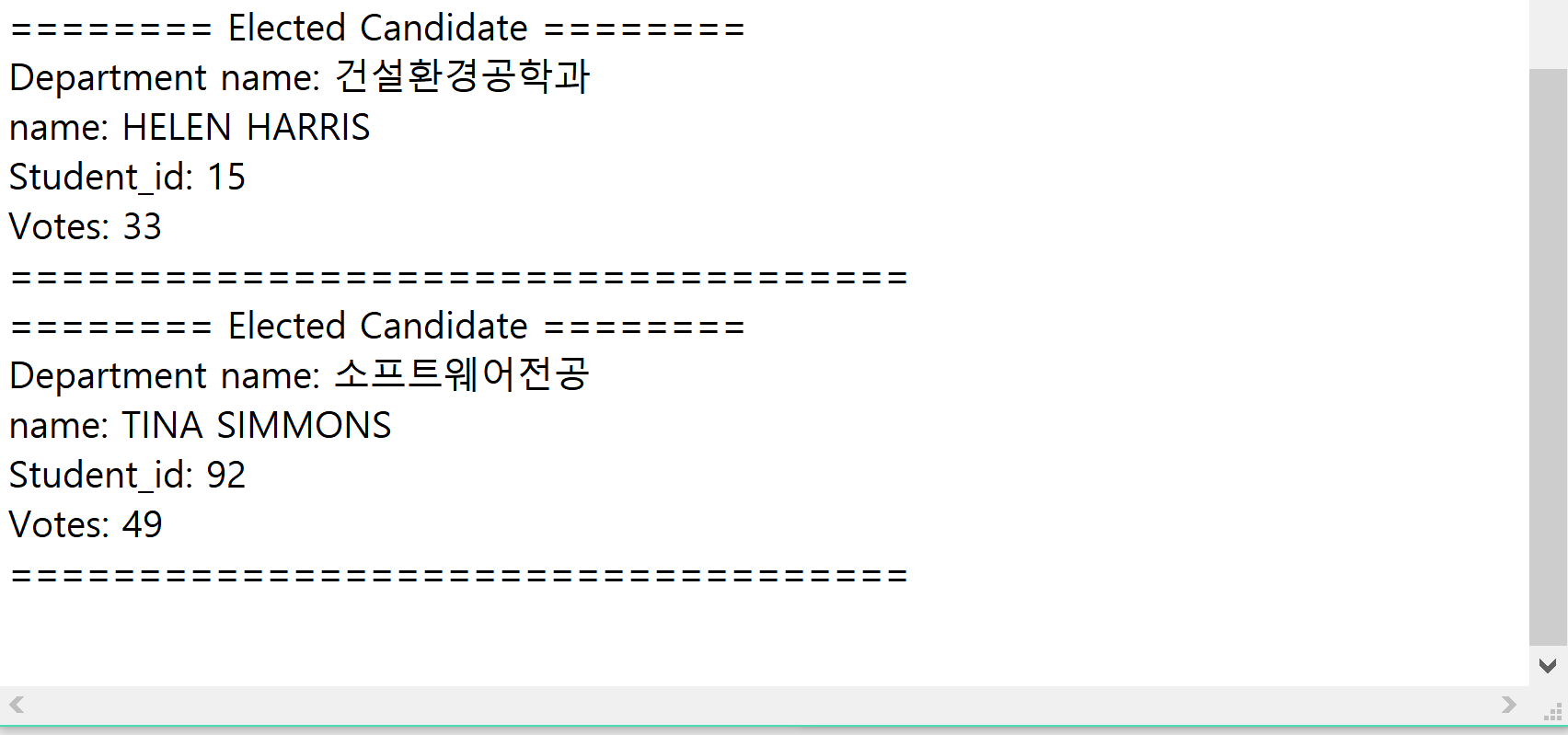
c) PrintWriter의 함수를 이용해 OUPUT\_FILE\_PATH의 이름을 가진 파일에 원하는 정보를 쓴다.

d) 파일을 close한다.

**<program’s ouput>**







//output파일명을 잠시 변경하였습니다.

// =====부분의 크기가 다른이유는 글씨체가 달라서이기 때문입니다.