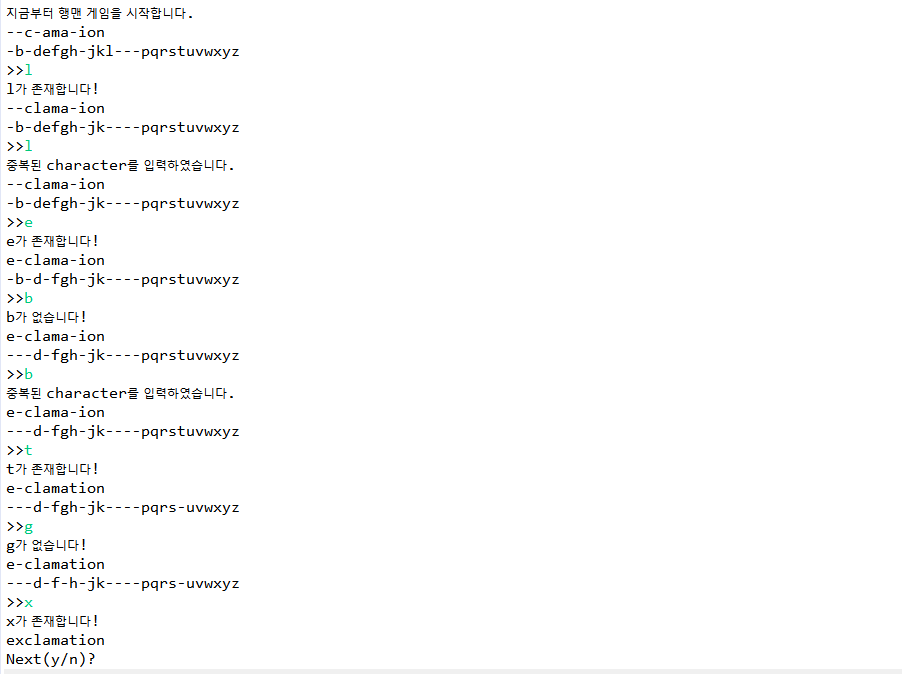
개인 프로젝트1\_김지원\_1711412

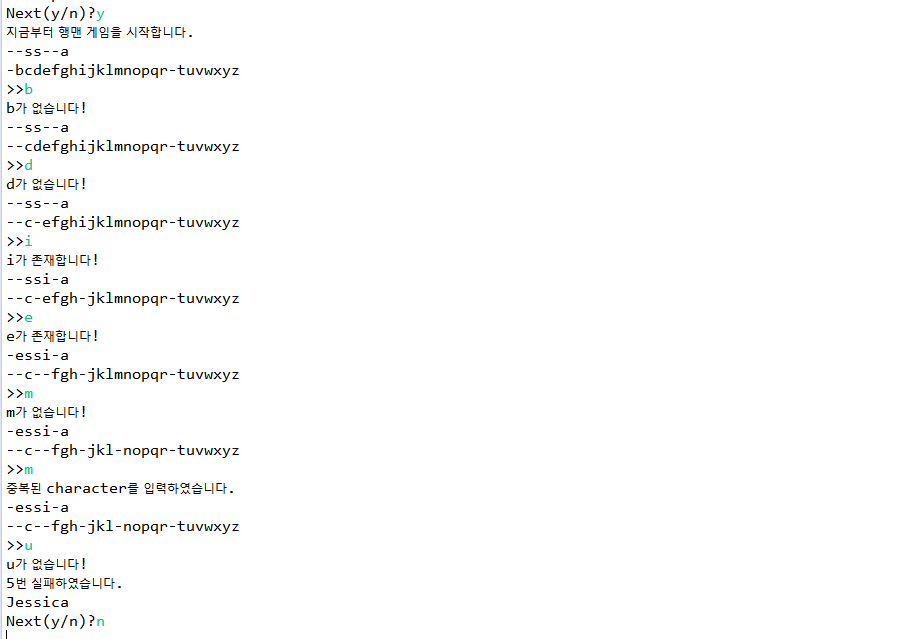
1. **클래스 설계 – 사용한 클래스, 메소드, 변수명 모두 기술하세요.**
2. **클래스1(CreateWords): word.txt.에 있는 단어들 중 한 단어를 랜덤으로 골라준다.**
   1. **메소드1(randomWord()): word.txt.를 읽어와 랜덤한 n번째 라인에 있는 단어를 str에 저장해 리턴 한다.**
      1. **BufferedReader br: file을 읽어와 버퍼에 보관 후 버퍼에서 읽어오는 BufferedReader 생성**
      2. **File file: 파일 정보에 따른 File 생성**
      3. **Ramdom random: 랜덤 발생기 생성**
      4. **String str: 라인 단위로 읽어온 데이터를 저장하는 변수**
      5. **Int n: word.txt.에 있는 1줄~25143줄 중에서 랜덤하게 설정될 ‘라인 수 -1’을 나타내는 변수**
3. **클래스2(Project\_김지원\_1711412): CreateWords class에서 생성된 랜덤의 word를 받아 가릴 알파벳 4개를 정하고 종료 조건 전 까지 게임을 실행시킨다.**
   * 1. **StringBuffer hiddenWord: hiddenWord StringBuffer에 랜덤으로 설정된 단어 저장**
     2. **String newRandomWord: 랜덤으로 설정된 단어**
     3. **char [] alpabat = ‘a’-‘z’까지 들어있는 알파벳 배열**
     4. **boolean check[]=입력한 알파벳이 중복인지 확인하기 위한 배열**
     5. **int fail: 실패한 횟수**
   1. **메소드1(gameStart()): 게임을 시작하고 한 게임이 끝나면 계속할지 여부를 물어본다.**
      1. **String answer : 다음을 게임을 할지 말지 y/n로 입력한 사용자의 입력을 받는 변수**
   2. **메소드2(hideWord()): 숨길 알파벳 4개 설정 후 단어에서 해당하는 알파벳을 ‘-‘로 바꿔준다. 알파벳 배열은 단어에서 숨길 알파벳을 제외한 나머지 알파벳을 ‘-‘로 바꿔준다.**
      1. **Int index[]: 0~(단어 길이-1) 사이의 랜덤 수 저장하는 배열**
      2. **Char c: 단어의 랜덤 인덱스의 문자를 c에 저장**
   3. **메소드3(play()): 종료조건이 되기 전까지 게임을 진행한다.**
      1. **Char f\_input: 사용자가 입력한 알파벳 한 글자**
   4. **메소드4(result(char f\_input)): 사용자가 입력한 알파벳을 받아 단어에 해당하는지 아닌지를 확인후 게임의 결과를 리턴 해준다. 성공 시 true 리턴 아니면 false 리턴 한다.** 
      1. **Boolean success: 성공여부를 나타내는 변수**
   5. **Main 메소드: Project\_김지원\_1711412의 game객체를 생성한 후**

**gameStart()메소드를 실행시켜 게임을 시작하게 한다.**

1. **실행화면 - 5번 안에 성공한 경우, 실패한 경우 각각 캡처**

**성공:**

****

**실패:**