

# Report\_Chap07

Date: 2022. 05. 03

Name: 송예지

Student ID: 22100396

## Project #1.

Write a program that will read a line of text that **ends with a period**, which serves as a sentinel value. Display all the letters that occur in the text, one per line and **in alphabetical order**, along with the number of times each letter occurs in the text. Use an array of base type int

of length 26, so that the element at index 0 contains the number of a's, the element at index 1 contains the number of b's, and so forth. Allow both uppercase and lowercase letters as input, but treat upper case and lowercase versions of the same letter as being equal. Hint s: Use one of the methods toUpperCase or toLowerCase in the wrapper class Character, described in Chapter 6. You will find it helpful to

define a method that takes a character as an argument and returns an int value that is the correct index for that character. For example, the argument 'a' results in 0 as the return value, the argument 'b' gives 1 as the return value, and so on. Note that you can use a type cast, such as (int)letter, to change a char to an int. Of course, this will not get the number you want, but if you subtract (int)'a', you will then get the right index. Allow the user to repeat this task until the user says she or he is through.

# 센티넬 값 역할을 하는 마침표로 끝나는 텍스트 행을 읽는 프로그램을 작성한다. 텍스트에서 각 문자가 발생하는 횟수와 함께 한 줄당 하나씩 알파벳 순으로 텍스트에서 발생하는 모든 문자를 표시합니다. 기본 유형 배열 사용

인덱스 0의 요소가 a의 수를 포함하도록, 인덱스 1의 요소가 b의 수를 포함하도록, 길이 26. 대문자와 소문자를 모두 입력으로 허용하되, 동일한 문자의 대문자 및 소문자 버전은 동일한 것으로 간주합니다.

힌트: 6장에 설명된 클래스 문자 줄 바꿈에서 대문자나 소문자로 가는 방법 중 하나를 사용합니다. 하는 것이 도움이 될 것이다

문자를 인수로 사용하고 해당 문자에 대한 올바른 인덱스인 int 값을 반환하는 메서드를 정의합니다.

예를 들어 인수 'a'는 반환값으로 0이 되고 인수 'b'는 반환값으로 1이 주어지는 식입니다. (int) 글자와 같은 형식 캐스트를 사용하여 문자를 int로 변경할 수 있습니다. 물론, 이것은 여러분이 원하는 숫자를 얻을 수는 없지만, 만약 여러분이 (int) 'a'를 빼면, 여러분은 올바른 인덱스를 얻을 것입니다. 사용자가 완료했다고 말할 때까지 이 작업을 반복할 수 있습니다.

---

## 기본 문제 해결

- ➔ 26 개의 size 를 갖고 있는 int array 생성
  - ➔ 입력 받은 문자열 공백 제거 후 대문자로 변환 시켜주기
  - ➔ 문자열을 하나씩 anIndex(method)에 보내서 alphabet 의 위치 값 return 받기
  - ➔ A(0)~Z(25)
- 
- ➔ anIndex(char c) method 는 parameter 로 받아온 문자 아스키코드를 통해서 숫자로 변환한 값에서 65 빼주기 (ex) (int)c - 65
  - ➔ 문자열을 모두 대문자로 변환 해줬기 때문에 65 빼줌

---

# 실습 코드

```
import java.util.Scanner;

public class AlphabeticalOrder {

    public static void main(String[] args) {
        String sentence;
        String answer;
        Scanner s = new Scanner(System.in);
        // int [] index = new int[26];

        do {
            int [] index = new int[26];
            System.out.println("Enter a line of text");
            System.out.println("terminated by a period.");
            sentence = s.nextLine();

            sentence = sentence.replaceAll("\\s", "");
            sentence = sentence.toUpperCase();

            // System.out.println(sentence);    // 확인용

            for(int i = 0; i<sentence.length(); i++) {
                if(sentence.charAt(i)=='.') break;
                int n = anIndex(sentence.charAt(i));
                index[n]++;
            }
        }
    }
}
```

```

        for(int i = 0; i<26; i++) {
            if(index[i] == 0) continue;
            System.out.println((char)(i+65)+" "+index[i]);
        }

        System.out.println("Enter another line? (y for 'yes') ");
        answer = s.nextLine();

        }while(answer.equals("y")|| answer.equals("Y"));

    }

    public static int anIndex(char a) {
        return ((int)a)-65;
    }
}

```

---

# 실습 코드 설명

```
import java.util.Scanner;
```

```
public class AlphabeticalOrder {
```

```
    public static void main(String[] args) {
```

```
        String sentence;
```

→ 문장 입력 받을 변수

```
        String answer;
```

→ While문 종료를 위한 변수

```
        Scanner s = new Scanner(System.in);
```

```
do {
```

```
    int [] index = new int[26];
```

→ array 생성 ( alphabet 개수로 size 지정)

```
    System.out.println("Enter a line of text");
```

```
    System.out.println("terminated by a period.");
```

```
    sentence = s.nextLine();
```

→ 문장 입력 받기

```
    sentence = sentence.replaceAll("\\s", "");
```

→ 받은 문자열의 공백 제거

```
    sentence = sentence.toUpperCase();
```

→ 받은 문자열의 문자들 모두 대문자로 변환

```
//
```

```
    System.out.println(sentence);    // 확인용
```

```
    for(int i = 0; i<sentence.length(); i++) {
```

→ 입력받은 문자열의 길이만큼 반복문 돌리기

```
        if(sentence.charAt(i)=='.') break;
```

→ .이 나오면 반복문 종료

```
        int n = anIndex(sentence.charAt(i));
```

→ Method를 통해 문자의 alphabet 위치 return 받기

```
        index[n]++;
```

→ Return 값 위치의 index array 1증가

```
    }
```

```
    for(int i = 0; i<26; i++) {
```

→ Alphabet 갯수만큼 반복문 돌리기

```
        if(index[i] == 0) continue;
```

→ 1번째의 index array가 0이면 출력 없이 진행

→

```
System.out.println((char)(i+65)+" "+index[i]);
```

→ i번째 alphabet 출력하고 그 alphabet이 몇 개 입력 되었는지까지 출력

```
}
```

```
System.out.println("Enter another line? (y for 'yes') ");
```

→ while문 반복할지 물어보기

```
answer = s.nextLine();
```

→ while문 반복 여부에 대한 문자 입력받기

```
}while(answer.equals("y")|| answer.equals("Y"));
```

→ answer이 y(or Y)이면 계속 진행

```
}
```

```
public static int anIndex(char a) {
```

```
return ((int)a)-65;
```

```
}
```

→ Parameter 온 문자열 아스키 코드로 변환 후 65를 빼 줌

→ Alphabet 위치 return (ex) a이면 0 return )

```
}
```

---