

객체지향 프로그래밍 실습 과제

14주차. Generic Classes and Methods

개요

파일 입출력과 함께 Generic을 이용해 프로그램을 구현한다.

조건

1. IO_Class (5%)

2개의 제네릭 파라미터 T, M을 가지는 제네릭 타입 클래스이다.

다음과 같은 멤버변수를 가진다.

```
T input;  
M output;
```

각 멤버변수에 대한 getter, setter가 존재한다.

2. ReadText 클래스(30%)

IO_Class를 상속하며 제네릭 파라미터로 T, M을 가진다.

다음과 같은 멤버변수를 가진다.

```
String textString  
int numberOfSpace
```

numberOfSpace의 초기값은 0으로 한다.

멤버변수에 대한 getter가 존재한다.

다음과 같은 메소드를 가진다.

```
void readData(T input, String from){...}
```

- 파라미터로 전달받은 input을 해당 타입의 멤버변수에 대입한다.
- input이 String이라면 반복문을 통해 다음과 같은 기능을 수행한다.
 - 입력 스트림으로 FileReader를 만들어 from로부터 10개의 character씩 읽어들인다.
 - 읽어 들인 10개의 character 중에서 알파벳 소문자만 textString에 문자열로 연결한다.
 - 알파벳 소문자가 아니면 numberOfSpace의 값을 1 증가한다.
- Input이 String이 아니면 “알 수 없는 데이터 타입입니다.”를 콘솔에 출력한다.
- 파일 입력이 끝났다면 입력 스트림을 닫는다.

3. WriteText 클래스(30%)

IO_Class를 상속하며 제네릭 파라미터로 T, M을 가진다.

다음과 같은 메소드를 가진다.

```
public void writeData(M output, String to, ReadText rt){...}
```

- 파라미터로 전달받은 output을 해당 타입의 멤버변수에 대입한다.
- output이 String이라면 다음과 같은 기능을 수행한다.
 - 출력 스트림으로 FileWriter를 만들어 to에 rt의 textString을 출력한다.
 - 출력이 완료되면 출력이 완료되었다는 문구를 콘솔에 출력한다.
- output이 String이 아니면 “알 수 없는 데이터 타입입니다.”를 콘솔에 출력한다.
- 파일 출력이 끝났다면 버퍼를 비우고 출력 스트림을 닫는다.

4. Main 클래스(30%)

과제와 함께 첨부한 input.txt의 경로를 String from 변수에 대입한다.

output.txt이 생성될 위치를 String to 변수에 대입한다.

제네릭 타입을 모두 String으로 가지는 ReadText 객체와 WriteText 객체를 생성한다.

파일 입출력을 수행하고 실행화면처럼 콘솔에 내용을 출력한다.

```
text 파일의 총 글자 수: 274
```

```
text 파일의 알파벳 수: 237
```

```
출력 완료
```

```
[실행 화면]
```

```
|hullfluvaccineinfluenzavaccineflushotvaccineflushotintramuscularinjectiondeltoidflushotdeltoidflushotintradermalinjectionintramuscularinjectiondeltoidnasalsprayflushotnasalsprayvirusvaccineflushotinactivatedvaccineinactivatedvirusvaccine
```

```
[output.txt]
```

프로젝트명, 압축파일명을 "학번_이름_HW14"로 한다. [5%]

- 과제의 끝 -