
숙제4 -클래스, 상속 다형성, 파일

7.1 (`Rectangle` 클래스) 7.2절의 `Circle` 클래스 예제를 참조하여 사각형을 표현하는 `Rectangle`이라는 클래스를 설계하시오. `Rectangle` 클래스는 다음을 포함한다.

- 두 데이터 필드: `width`와 `height`
- 특정 폭(`width`)과 높이(`height`)를 갖는 사각형을 생성하는 생성자. 폭과 높이의 기본값은 각각 1과 2이다.
- 사각형의 넓이를 반환하는 `getArea()` 메소드
- 사각형의 둘레를 반환하는 `getPerimeter()` 메소드

이 클래스에 대한 UML 다이어그램을 작성한 후 클래스를 구현하시오. 하나는 폭 4, 높이 10, 다른 하나는 폭 3.5, 높이 35.7인 두 사각형을 생성하는 테스트 프로그램 작성하시오. 각 사각형의 폭, 높이, 넓이 및 둘레 순으로 출력하시오.

7.2 (Stock 클래스) 회사의 주식(stock)을 표현하는 `Stock`이라는 클래스를 설계하시오. `Stock` 클래스는 다음을 포함한다.

- 주식의 코드를 위한 private string 데이터 필드 `symbol`
- 주식의 종목명을 위한 private string 데이터 필드 `name`
- 전일 주식 마감 가격(전일종가)을 저장하는 private float 데이터 필드 `previousClosingPrice`
- 현재 시각의 주식 가격(현재가)을 저장하는 private float 데이터 필드 `currentPrice`
- 특정 코드, 종목명, 전일종가, 현재가의 주식을 생성하는 생성자
- 종목명을 반환하는 `get` 메소드
- 주식 코드를 반환하는 `get` 메소드
- 전일종가를 반환하고 설정하기 위한 `get`과 `set` 메소드
- 현재가를 반환하고 설정하기 위한 `get`과 `set` 메소드
- `previousClosingPrice`에서 `currentPrice`로 변화된 비율을 반환하는 `getChangePercent()` 메소드

`Stock` 클래스에 대한 UML 다이어그램을 작성한 후, `Stock` 클래스를 구현하시오. 주식 코드가 `INTC`, 종목명은 Intel Corporation, 전일종가가 `20500`이고 현재가가 `20350`인 `Stock` 객체를 생성하고 가격변동률을 출력하는 테스트 프로그램을 작성하시오.

7.4 (**Fan** 클래스) 팬을 표현하는 **Fan** 클래스를 설계하시오. **Fan** 클래스는 다음을 포함한다.

- 팬의 속도를 나타내는 각각의 값이 1, 2, 3인 세 상수 **SLOW**, **MEDIUM**과 **FAST**
- 팬의 속도를 명시하는 private **int** 데이터 필드 **speed**
- 팬의 전원이 켜져 있는지 나타내는 private **bool** 데이터 필드 **on**(기본값 **False**).
- 팬의 반지름(크기)을 나타내는 private **float** 데이터 필드 **radius**
- 팬의 색상을 나타내는 private **string** 데이터 필드 **color**
- 모든 데이터 필드에 대한 접근자와 변경자
- 특정 속도(기본값 **SLOW**), 반지름(기본값 5), 색상(기본값 **blue**), 그리고 전원(기본값 **False**)에 대한 선풍기를 생성하는 생성자

Fan 클래스에 대한 UML 다이어그램을 작성한 후 클래스를 구현하시오. 두 개의 **Fan** 객체를 생성하는 테스트 프로그램을 작성하시오. 첫 번째 객체에 대해 최대 속도, 크기 10, 색상 노란색 그리고 전원 켜짐을 할당하시오. 두 번째 객체에 대해 중간 속도, 크기 5, 색상 파란색 그리고 전원은 꺼짐을 할당하시오. 각 객체의 속도, 반지름, 색상 및 전원 속성을 출력하시오.

***7.8** (스톱워치) `StopWatch` 클래스를 설계하시오. 클래스는 다음을 포함한다.

- private 데이터 필드 `startTime`과 `endTime` 및 `get` 메소드
- 현재시간으로 `startTime`을 초기화하는 생성자
- 현재시간으로 `startTime`을 재설정하는 `start()` 메소드
- 현재시간을 `endTime`으로 설정하는 `stop()` 메소드
- 스톱워치의 경과시간을 밀리초 단위로 반환하는 `getElapsedTime()` 메소드

`StopWatch` 클래스의 UML 다이어그램을 작성한 후 클래스를 구현하시오. 1부터 1,000,000까지 숫자를 더하는 데 소요되는 실행 시간을 측정하는 테스트 프로그램을 작성하시오.

12.1 (Triangle 클래스) GeometircObject 클래스를 확장한 Triangle 클래스를 설계 하시오. Triangle 클래스는 다음의 항목들을 포함해야 한다.

- 삼각형의 세 변을 나타내는 실수타입 데이터 필드인 `side1`, `side2`, `side3`
- 지정된 `side1`, `side2`, `side3`(기본값은 1.0)를 이용하여 삼각형을 만들어 주는 생성자
- `side1`, `side2`, `side3` 데이터 필드 모두에 대한 접근자 메소드
- 삼각형의 넓이를 반환하는 `getArea()` 메소드
- 삼각형의 둘레를 반환하는 `getPerimeter()` 메소드
- 삼각형의 설명을 문자열로 반환하는 `__str__()` 메소드

삼각형의 넓이 계산을 위한 수식은 프로그래밍 연습문제 2.14를 참조하시오. `__str__()` 메소드는 다음과 같이 구현된다.

```
return "Triangle: side1 = " + str(side1) + "side2 = " +  
    str(side2) + "side3 = " + str(side3)
```

`Triangle`과 `GeometricObject` 클래스에 관한 UML 다이어그램을 그리시오. 아울러, `Triangle` 클래스를 구현하시오. 사용자로부터 삼각형의 세 변의 값, 색상 값, 삼각형 내부의 채움 여부를 나타내는 1 또는 0의 값을 입력받는 테스트 프로그램을 작성하시오. 이들 입력값을 이용하여 `Triangle` 객체를 생성하고 삼각형의 넓이, 둘레, 색상을 출력해야 한다. 또한 삼각형 내부가 색상으로 채워져 있는지의 여부를 `True` 혹은 `False`로 출력해야 한다.


****13.1** (텍스트 제거) 텍스트 파일로부터 특정 문자열이 나타난 모든 부분을 제거하는 프로그램을 작성하시오. 프로그램은 사용자로부터 파일명과 제거할 문자열을 입력받는다. 실행 예는 다음과 같다.

파일이름을 입력하세요: test.txt

제거할 문자열을 입력하세요: morning

완료

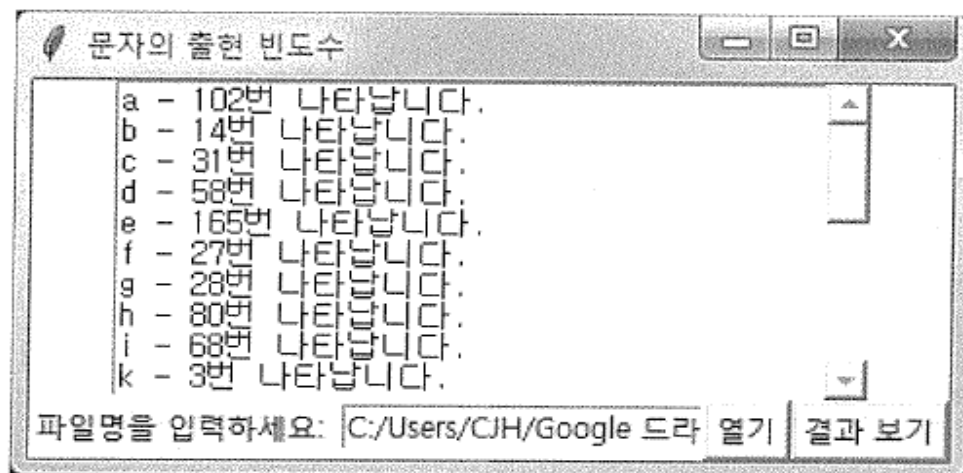
***13.2** (파일에 문자, 단어 및 행의 개수 세기) 파일에 포함된 문자, 단어 및 행의 개수를 세는 프로그램을 작성하시오. 단어는 공백으로 분리된다. 프로그램은 사용자로 부터 파일이름을 입력받는다. 실행 예는 다음과 같다.

파일이름을 입력하세요: test.txt 

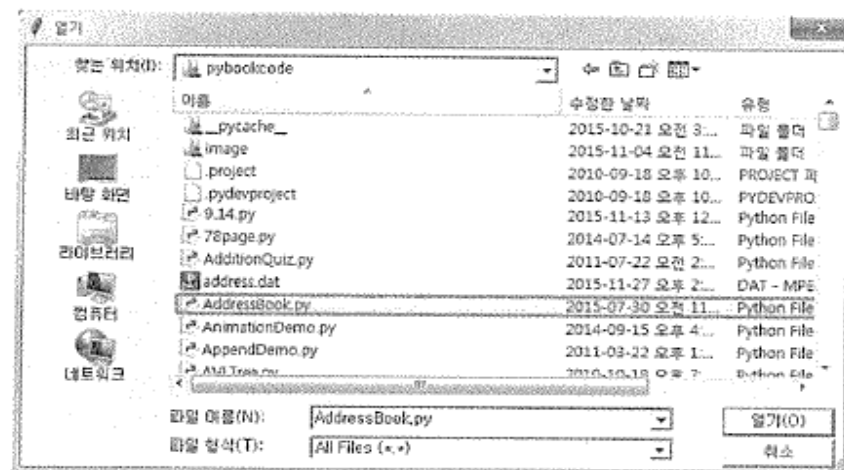
문자 1777 개

단어 210 개

행 71 개



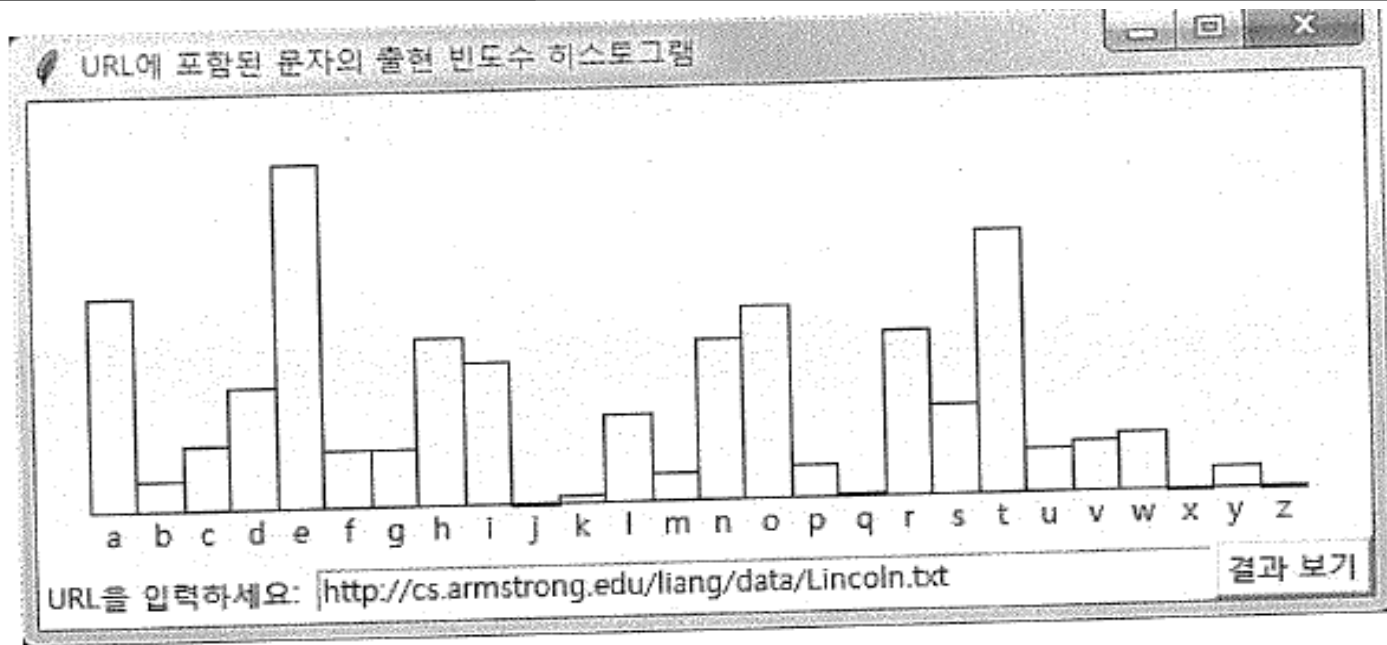
(a)



(b)

[그림 14.3] 이 프로그램은 파일을 선택하고 파일 내 각 문자의 빈도수를 센다.

*14.4 (Tkinter: 각 문자의 빈도수 세기) 그림 14.3(a)와 같이 입력 필드로부터 파일 이름을 입력받는 GUI를 사용할 수 있도록 코드 14.5를 재작성하시오. 또한 그림 14.3(b)와 같이 파일오픈 대화상자를 출력하기 위해 열기 버튼을 클릭하여 파일을 선택할 수 있도록 하라. 결과 보기 버튼을 클릭하면 텍스트 위젯에 그 결과를 출력한다. 입력받은 파일이 존재하지 않으면, 메시지 박스를 통해 메시지를 출력해야 한다.



[그림 14.6] 이 프로그램은 파일을 URL 형태로 입력받고, 각 문자의 빈도수를 막대그래프로 출력한다.

*14.7 (Tkinter: 각 문자의 빈도수 세기) 그림 14.6와 같이 결과가 막대그래프로 출력되도록 이전 프로그래밍 연습문제를 수정하시오. URL이 존재하지 않으면, 메시지 박스를 통해 메시지를 출력해야 한다.