

노트북 프로그램 만들기

- 노트북 프로그램
 - 페이지 단위로 노트를 삽입하는 노트북 객체를 이용하여 노트북 프로그램 구현

- 노트(note)를 정리하는 프로그램이다.
- 사용자는 노트에 콘텐츠를 적을 수 있다.
- 노트는 노트북(notebook)에 삽입된다.
- 노트북은 타이틀(title)이 있다.
- 노트북은 노트가 삽입될 때 페이지를 생성하며, 최대 300페이지까지 저장할 수 있다.
- 300페이지를 넘기면 노트를 더는 삽입하지 못한다.

노트북 프로그램 만들기

- Note class
 - 노트에 내용을 쓰고 삭제하는 기능을 포함하는 객체
 - 객체속성변수
 - contents #노트の内容
 - 객체함수 :
 - write_contents() #노트에 내용을 넣기
 - remove_all() #노트の内容 지우기

노트북 프로그램 만들기

- Note 클래스

```
class Note(object):  
    def __init__(self, contents):  
        self.contents = contents  
  
    def remove(self):  
        self.contents = "삭제된 노트입니다."  
  
    def __str__(self):  
        return self.contents
```

노트북 프로그램 만들기

- Notebook class

- 타이틀과 페이지에 따른 노트들을 저장할 속성변수
- 노트를 추가하고 삭제하는 기능
- 객체속성변수
 - title #노트북 제목
 - notes #노트 저장공간, dictionary {(페이지:노트), ..}
 - nextpage #추가할 다음 페이지 번호
- 객체함수 :
 - add_note() #노트추가
 - remove_note() #노트 삭제
 - print_all_pages() #모든 페이지 출력

노트북 프로그램 만들기

- Notebook 클래스

```
class Notebook(object):
    def __init__(self, title):
        self.title = title
        self.nextpage = 1
        self.notes = {}

    def add_note(self, note, page=0):
        if len(self.notes.keys()) < 300:
            if (page == 0):
                self.notes[self.nextpage] = note
                print(self.nextpage, "page 추가합니다.")
                self.nextpage += 1
            else:
                if (page not in self.notes.keys()):
                    self.notes[page] = note
                    print(page, "page 삽입합니다.")
                else:
                    print(page, "page에 노트가 존재합니다.")
        else:
            print("더 이상 노트를 추가하지 못합니다.")
```

노트북 프로그램 만들기

- Notebook 클래스

```
def remove_note(self, page):
    if (page in self.notes.keys()):
        del self.notes[page]
        print(page, "page 노트를 삭제합니다.")
    else:
        print(page, "page는 존재하지 않습니다.")

def print_all_pages(self):
    s = sorted(self.notes.items())
    for k,v in s:
        print(k,v)

def __str__(self):
    return "노트북 제목 : " + self.title + ", 총페이지수 : " + str(len(self.notes.keys()))
```

노트북 프로그램 만들기

- Note 사용

```
#Note 생성 및 삭제
good_sentence = ""진정한 벗은 제2의 자신이다. -아리스토텔레스""
note_1 = Note(good_sentence)

print(note_1)
note_1.remove()
print(note_1)
```

진정한 벗은 제2의 자신이다. -아리스토텔레스
삭제된 노트입니다.

노트북 프로그램 만들기

- Notebook 사용

```
#NoteBook 생성
wise_saying_notebook = Notebook("명언 노트")
print(wise_saying_notebook)
#Note 생성
good_sentence = ""오늘보다 더 가치있는 것은 없다. - 괴테 ""
note_2 = Note(good_sentence)
good_sentence = ""시간은 모든것을 밝혀낸다. - 에라스무스""
note_3 = Note(good_sentence)
#NoteBook에 Note 추가
wise_saying_notebook.add_note(note_2) #다음 페이지에 추가
wise_saying_notebook.add_note(note_3) #다음 페이지에 추가
print(wise_saying_notebook)
wise_saying_notebook.print_all_pages()
#NoteBook에서 Note 삭제
wise_saying_notebook.remove_note(1) #1 페이지 삭제
print(wise_saying_notebook)
wise_saying_notebook.print_all_pages()
#NoteBook에서 Note 추가
good_sentence = ""진정한 벗은 제2의 자신이다. -아리스토텔레스""
note_1 = Note(good_sentence)
wise_saying_notebook.add_note(note_1) #다음 페이지에 추가
print(wise_saying_notebook)
wise_saying_notebook.print_all_pages()
#NoteBook에서 Note 삽입
wise_saying_notebook.add_note(note_2,1) #1 page에 삽입
print(wise_saying_notebook)
wise_saying_notebook.print_all_pages()
```

노트북 제목 : 명언 노트, 총페이지수 : 0
1 page 추가합니다.
2 page 추가합니다.
노트북 제목 : 명언 노트, 총페이지수 : 2
1 오늘보다 더 가치있는 것은 없다. - 괴테
2 시간은 모든것을 밝혀낸다. - 에라스무스
1 page 노트를 삭제합니다.
노트북 제목 : 명언 노트, 총페이지수 : 1
2 시간은 모든것을 밝혀낸다. - 에라스무스
3 page 추가합니다.
노트북 제목 : 명언 노트, 총페이지수 : 2
2 시간은 모든것을 밝혀낸다. - 에라스무스
3 진정한 벗은 제2의 자신이다. -아리스토텔레스
1 page 삽입합니다.
노트북 제목 : 명언 노트, 총페이지수 : 3
1 오늘보다 더 가치있는 것은 없다. - 괴테
2 시간은 모든것을 밝혀낸다. - 에라스무스
3 진정한 벗은 제2의 자신이다. -아리스토텔레스

QUIZ!

Report 9. 객체지향 프로그래밍 연습

1. 노트북 프로그램에 기능 추가

- 1) NoteBook class에 해당 페이지를 얻어 내용을 확인하는 객체함수 (getPage()) 추가
- 2) NoteBook 객체에 본인이 좋아하는 5개의 명언을 추가
- 3) 4 페이지의 내용을 getPage()로 확인

2. 사각형(Rectangle) 클래스를 정의하여 사각형의 면적과 둘레를 계산하는 프로그램 작성

QUIZ!

- 학습활동 결과는 e-루리에 제출바랍니다.
- 제목 : Report9. 객체지향프로그램
- 제출내용 : Report9_OOP_성명. ipynb 파일을 제출
- 제출기한 : 2021년 5월 25일 오후 11:58