



자료구조 및 알고리즘

보고서 #01

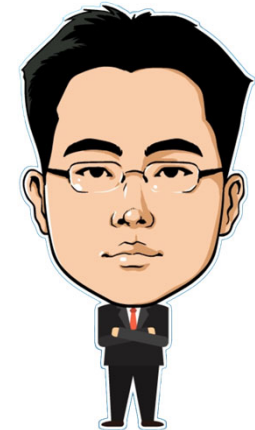
Python 프로그래밍



Seo, Doo-Ok

Clickseo.com

clickseo@gmail.com



보고서 #01

- **보고서 세부 내용**

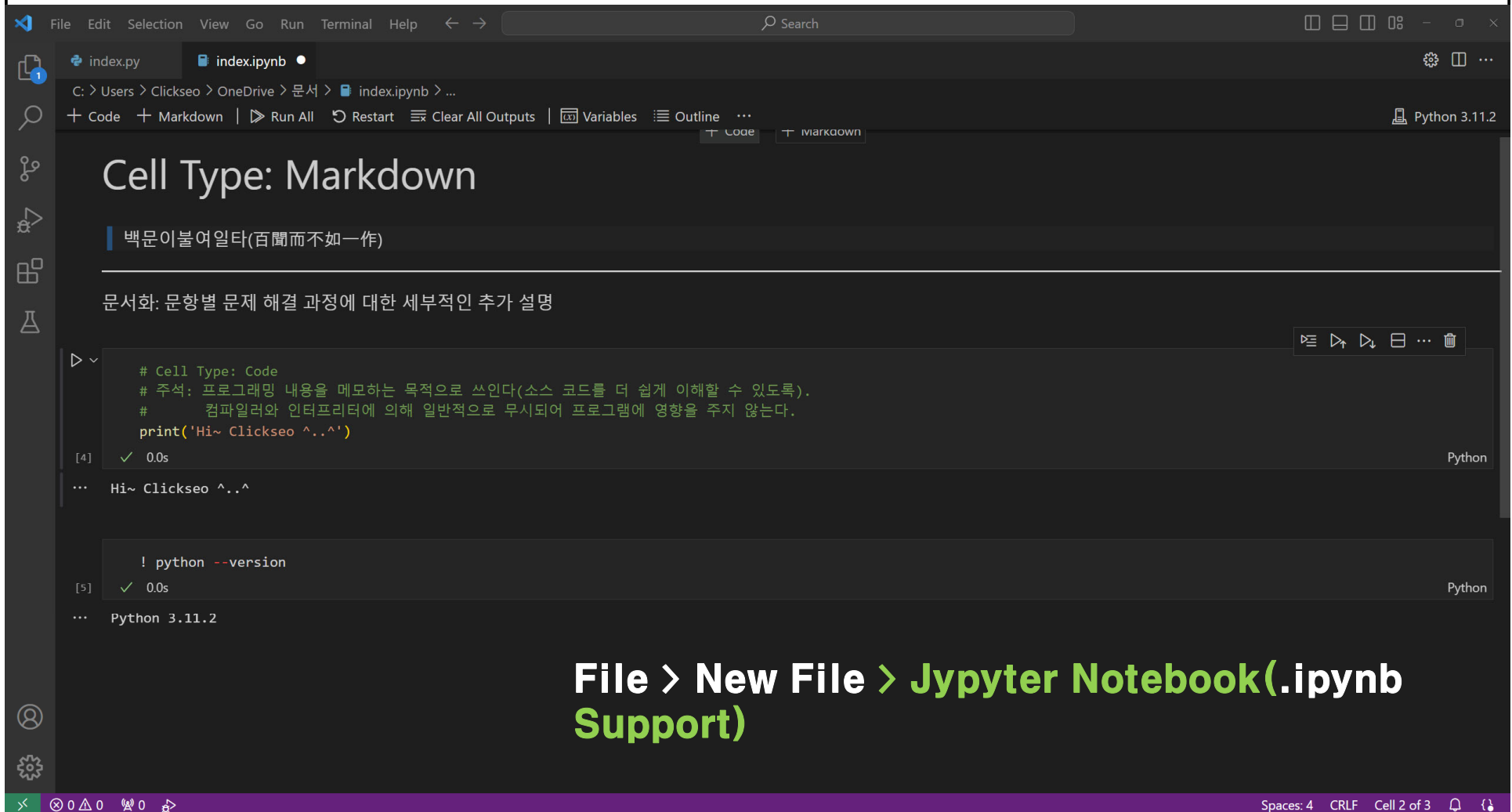
1. 사칙연산 계산: 예외 처리
2. 최댓값과 최솟값 반환 함수: 리스트
3. 전화번호부: 딕셔너리
4. 로또 복권: 번호 자동 생성
5. 계산기 클래스: 상속과 메소드 재정의
6. Albamon 클래스: 다중 상속

- **제출방법: (eCampus) 과제 제출에 파일 업로드**

- JupyterLab(Notebook) 또는 Google Colab: .ipynb 파일 제출
 - ✓ (Code Cell) 프로그램 소스 코드 및 주석
 - ✓ (Markdown 또는 Text Cell) 각 문항별 문제 해결 과정에 대한 세부적인 추가 설명(문서화)
- 보고서 분량은 별도 제한 없음.

- **제출 마감: (4주차) 2025년 09월 23일(화) 23:59**

Visual Studio Code

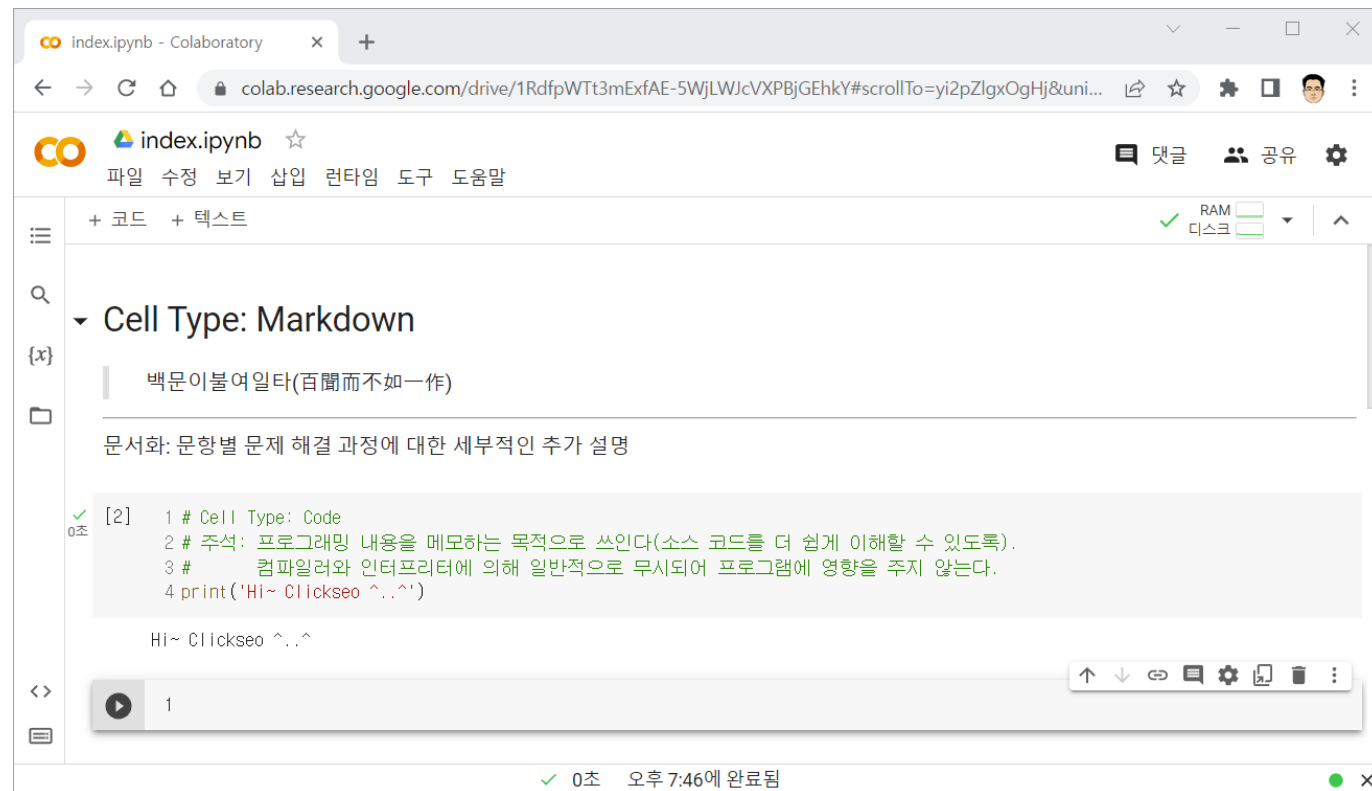


Google Colab

- Google Colab

- Cell Type: **Code**(소스 코드 및 주석 / 실행 결과), **Text**(추가 설명)

- 파일 > 다운로드 > **.ipynb** 다운로드

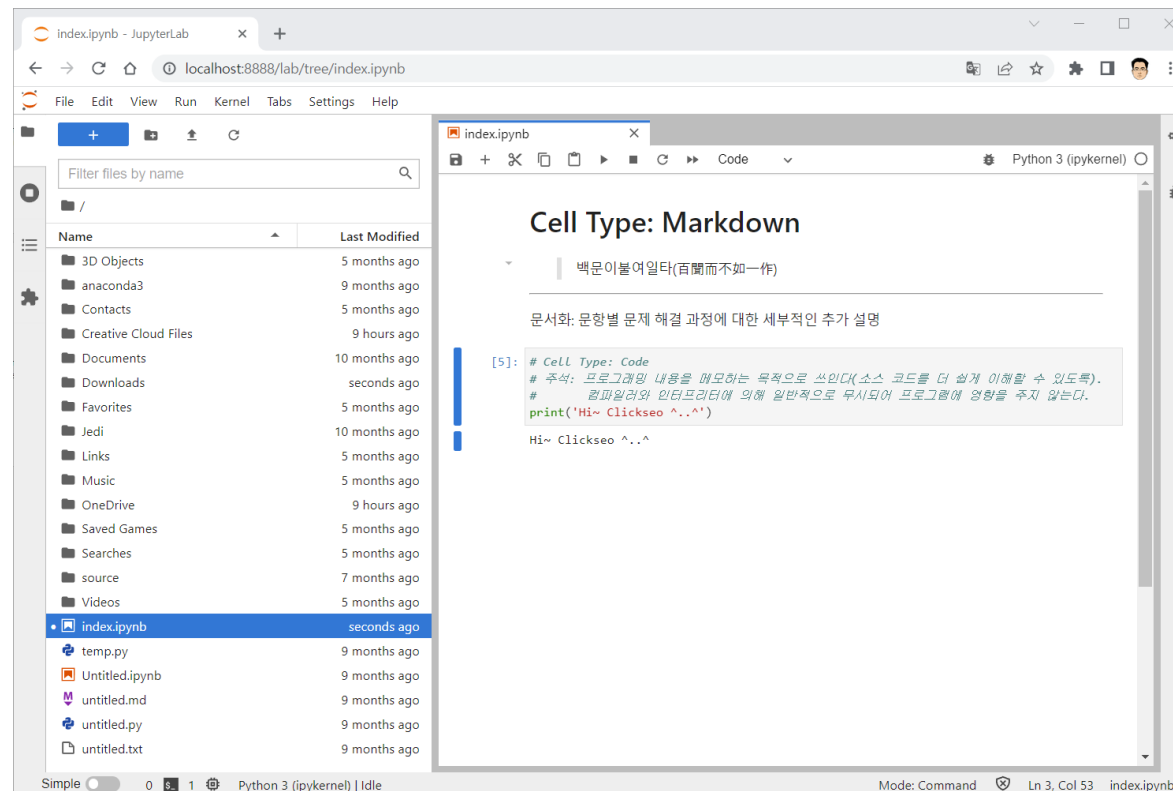


Project Jupyter (1/2)

● JupyterLab

○ Cell Type: **Code**(소스 코드 및 주석), **Markdown**(추가 설명)

- File > Download: Notebook(.ipynb) 파일

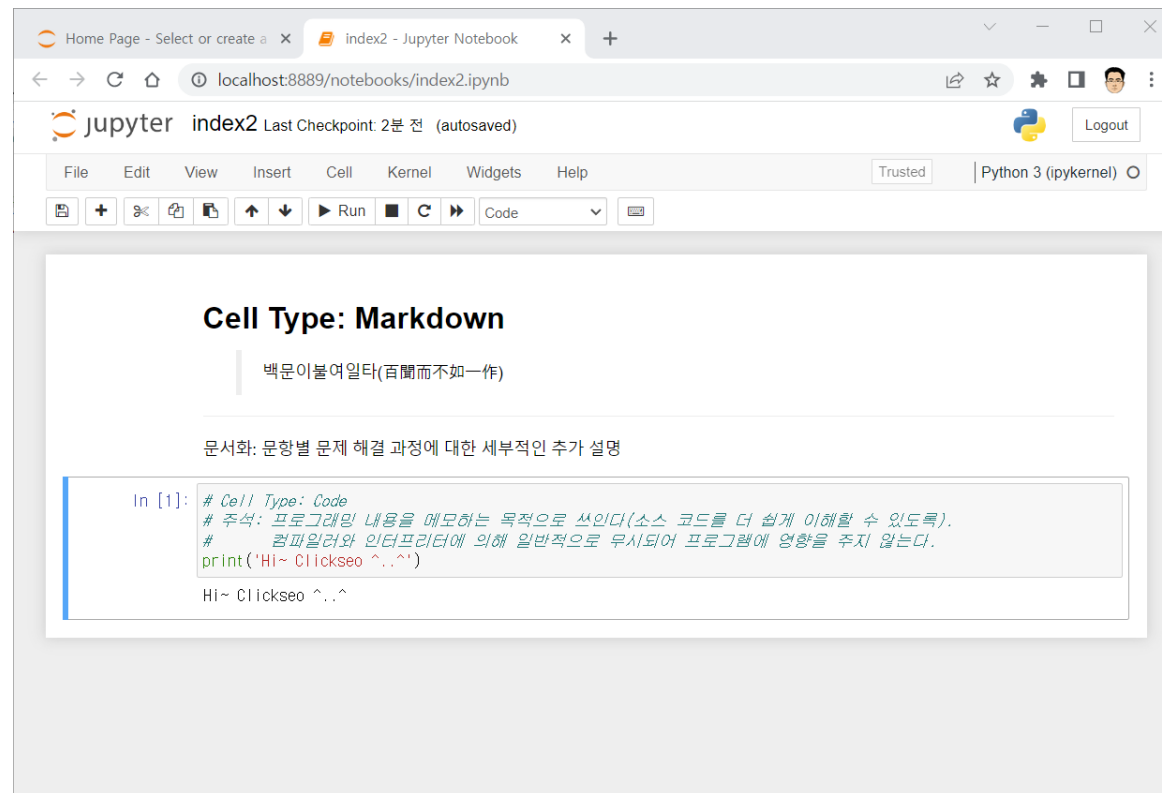


Project Jupyter (2/2)

● Jupyter Notebook

○ Cell Type: **Code**(소스 코드 및 주석), **Markdown**(추가 설명)

- File > Download as > Notebook(.ipynb)



자료구조 및 알고리즘

연습문제: Python 프로그래밍



연습문제 #01

실습 진행 문제

● 사칙연산 계산: 예외 처리

○ 사용자로부터 **수식(연산자, 피연산자)**을 입력 받아서, 주어진 연산자로 사칙 연산 계산 결과를 계산하는 프로그램을 작성하세요.

- 입력된 수식 계산 시 사칙연산(+, -, *, /)만을 지원한다.
- 단, 다중 선택(match-case) 구조를 이용하여 작성하세요.
- 또한 지원하지 않는 연산자일 경우와 0으로 나누려고 할 경우 **예외 처리** 기법을 활용하여 작성하세요.

○ 프로그램 실행 결과는 다음과 같다.

```
IDLE Shell 3.11.2
File Edit Shell Debug Options Window Help
Python 3.11.2 (tags/v3.11.2:878ead1, Feb
Type "help", "copyright", "credits" or "li
>>>
===== RESTART: C:/Users/click/
수식을 입력하세요(예: 10 + 20): 10 + 3
10 + 3 = 13
>>>
===== RESTART: C:/Users/click/
수식을 입력하세요(예: 10 + 20): 10 / 3
10 / 3 = 3.3333333333333335
>>>
===== RESTART: C:/Users/click/
수식을 입력하세요(예: 10 + 20): 10 / 0
오류: division by zero
>>>
===== RESTART: C:/Users/click/
수식을 입력하세요(예: 10 + 20): 10 ** 3
오류: 지원하지 않는 연산자입니다.
>>>
```


연습문제 #02

실습 진행 문제

- **최댓값과 최솟값 반환 함수: 리스트**

- 사용자로부터 임의의 정수를 입력 받아서, 그 중 가장 큰 수와 가장 작은 수를 모두 반환하는 함수를 작성하세요.

- 단, 파이썬 내장 함수인 **max**, **min** 함수를 사용하지 마세요. 제발!!!

- 프로그램 작성 시 함수 원형은 다음과 같다.

```
def MAXMIN(sList) :
```

- 프로그램 실행결과는 다음과 같다.

리스트

```
IDLE Shell 3.11.2
File Edit Shell Debug Options Window
Python 3.11.2 (tags/v3.11.2:87f35e6, Dec 4 2023)
Type "help", "copyright", "credits() or "license()" for more
>>>
===== RESTART: C:\Windows\system32\cmd.exe
임의의 정수를 입력하세요...
sList[0]: 5
sList[1]: 1
sList[2]: 9
sList[3]: 3
sList[4]: end
최대값: 9, 최소값: 1
>>>
```

```
IDLE Shell 3.11.2
File Edit Shell Debug Options Window
Python 3.11.2 (tags/v3.11.2:87f35e6, Dec 4 2023)
Type "help", "copyright", "credits() or "license()" for more
>>>
===== RESTART: C:\Windows\system32\cmd.exe
임의의 정수를 입력하세요...
sList[0]: 8
sList[1]: 3
sList[2]: 6
sList[3]: 2
sList[4]: end
최대값: 8, 최소값: 2
>>>
```

연습문제 #03

● 전화번호부: 딕셔너리

○ 딕셔너리를 사용하여 이름과 연락처를 저장하는 전화번호부 프로그램을 작성하세요.

- 입력 모드와 검색 모드로 동작하도록 구현하세요.
 - [입력 모드] : 이름과 전화번호를 입력 받고 딕셔너리에 저장한다.
 - [검색모드] : 친구들의 이름으로 전화번호를 검색할 수 있다.
 - 각각 이름을 입력하지 않고 엔터키를 치면 입력과 검색 모드를 종료한다.
- 각각의 기능을 함수로 구현하고, 메뉴 기능을 추가하세요.

○ 프로그램의 실행 결과는 다음과 같다.

```
*IDLE Shell 3.13.2*
File Edit Shell Debug Options Window Help
Python 3.13.2 (tags/v3.13.2:6e35360, Dec 6 2024)
Type "help", "copyright", "credits() or "license()" for more
>>>
##### 전화번호부 #####
1: 입력
2: 검색
3: 전체 출력
4: 프로그램 종료
메뉴 선택:
```

연습문제 #03: 실행 결과

● 전화번호부: 디렉터리

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

##### 전화번호부 #####
1: 입력
2: 검색
3: 전체 출력
4: 프로그램 종료

메뉴 선택: 1
[ 입력 모드 ]
이름(입력 종료: end): 홍길동
연락처: 010-1111-1111
이름(입력 종료: end): 이순신
연락처: 010-2222-2222
이름(입력 종료: end): 강감찬
연락처: 010-3333-3333
이름(입력 종료: end):
계속하려면 아무 키나 누르십시오 . . .
```

출력

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

##### 전화번호부 #####
1: 입력
2: 검색
3: 전체 출력
4: 프로그램 종료

메뉴 선택: 2
[ 검색 모드 ]
이름(검색 종료: end): 홍길동
홍길동의 전화번호는 010-1111-1111입니다.
이름(검색 종료: end): 이순신
이순신의 전화번호는 010-2222-2222입니다.
이름(검색 종료: end): 강감찬
강감찬의 전화번호는 010-3333-3333입니다.
이름(검색 종료: end):
계속하려면 아무 키나 누르십시오 . . .
```

검색

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

##### 전화번호부 #####
1: 입력
2: 검색
3: 전체 출력
4: 프로그램 종료

메뉴 선택: 3
##### 전화번호부 #####
('홍길동', '010-1111-1111')
('이순신', '010-2222-2222')
('강감찬', '010-3333-3333')
계속하려면 아무 키나 누르십시오 . . .

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

##### 전화번호부 #####
1: 입력
2: 검색
3: 전체 출력
4: 프로그램 종료

메뉴 선택: 4
프로그램 종료!!!
PS C:\Users\click\OneDrive\문서>
```

종료

연습문제 #04

실습 진행 문제

● 로또 복권: 번호 자동 생성

- 사용자가 원하는 매수 만큼 로또 복권의 번호를 자동 생성하여 출력하는 프로그램을 작성하세요.
 - 임의의 정수 데이터는 리스트를 활용하여 처리하세요.
 - 로또 복권은 1 ~ 45 사이의 임의의 난수 값으로 발생 시킨다.
 - 또한, 같은 줄에는 중복된 값이 절대로 존재하면 안된다.
 - 임의의 난수는 `random` 모듈의 `randrange` 또는 `randint` 조작 함수를 활용하세요.

```
import random

random.randrange(start=0, stop, step=1)

random.randint(start, stop) # randrange(start, stop+1)
```

- 프로그램 실행 결과는 다음과 같다.

```
IDLE Shell 3.10.4
File Edit Shell Debug Options Window Help
Python 3.10.4 (tags/v3.10.4:9d38120, Mar 23
Type "help", "copyright", "credits" or "lic
>>>
===== RESTART: C:\Users\Wclick
로또 복권을 몇 장 구입 하실래요? 5
[20, 29, 36, 39, 42, 45]
[2, 7, 9, 11, 12, 30]
[10, 12, 14, 25, 28, 42]
[7, 24, 30, 32, 37, 42]
[2, 5, 11, 12, 27, 45]
>>>
```

연습문제 #05

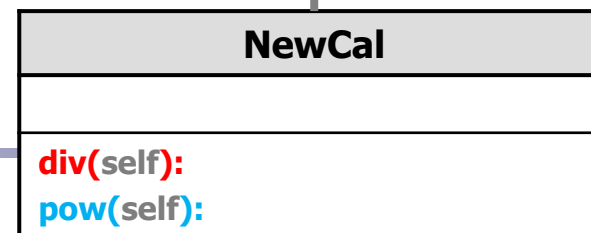
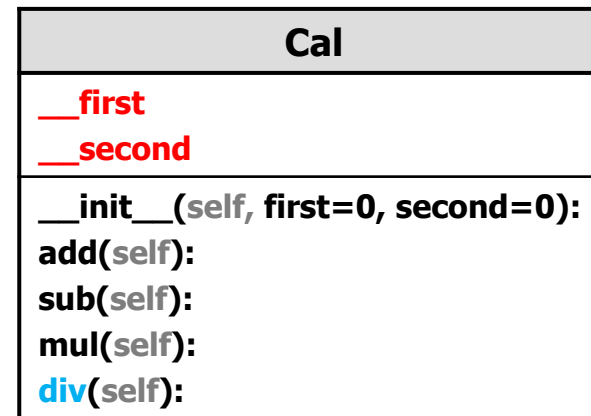
● 계산기 클래스: 상속과 메소드 재정의

- 사용자로부터 임의의 정수를 입력 받은 후에, 사칙 연산 결과 값을 출력하는 계산기 프로그램을 작성하세요.
 - 단, 데이터 접근은 해당 클래스의 메소드를 통해서만 접근이 가능하다(정보 은닉).
 - 즉, 클래스의 데이터 속성은 해당 클래스에서만 접근을 허용한다.
 - 단, 계산기 클래스를 상속받아 지수 계산(**) 기능을 추가하고, 또한 0으로 나눌 경우 반환 값이 0으로 처리되도록 나눗셈 연산을 재정의 하세요.

○ 프로그램 실행결과는 다음과 같다.

```
Python 3.11.2 (tags/v3.11.2:878ead1, Feb 7 2023, 16
Type "help", "copyright", "credits" or "license()" f
>>>
===== RESTART: C:\Users\Clickseo\OneDrive\
두 개의 정수를 입력하세요...
first: 10
second: 0
Traceback (most recent call last):
  File "C:\Users\Clickseo\OneDrive\문서\index.py", line 37, in <module>
    print(f'{xCal.getFirst()} / {xCal.getSecond()} = {xCal.div()}')
  File "C:\Users\Clickseo\OneDrive\문서\index.py", line 17, in div
    def div(self): return self.__first / self.__second
ZeroDivisionError: division by zero
>>>
```

```
Python 3.11.2 (tags/v3.11.2:878ead1, Feb 7 2023, 16
Type "help", "copyright", "credits" or "license()" f
>>>
===== RESTART: C:\Users\Clickseo\OneDrive\
두 개의 정수를 입력하세요...
first: 10
second: 3
10 + 3 = 13
10 - 3 = 7
10 * 3 = 30
10 / 3 = 3.3333333333333335
10 ** 3 = 1000
>>>
```



상속과 다형성

Albamon 클래스: 다중 상속

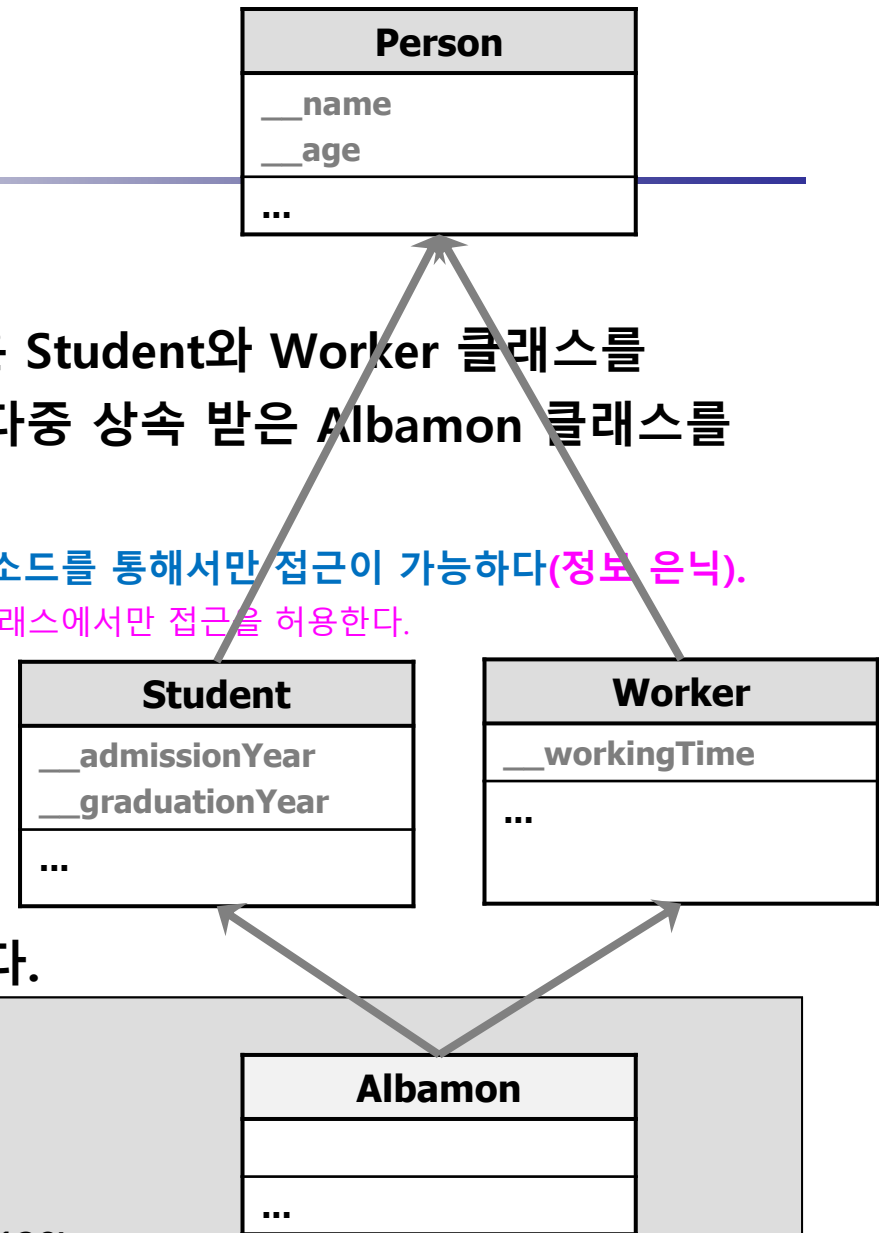


연습문제 #06

● Albamon 클래스

- Person 클래스와 이를 상속 받은 Student와 Worker 클래스를 설계하고, 이 두 가지 클래스를 다중 상속 받은 Albamon 클래스를 설계하세요.

- 단, 데이터 접근은 해당 클래스의 메소드를 통해서만 접근이 가능하다(정보 은닉).
 - 즉, 클래스의 데이터 속성은 해당 클래스에서만 접근을 허용한다.



- 프로그램 실행결과는 다음과 같다.

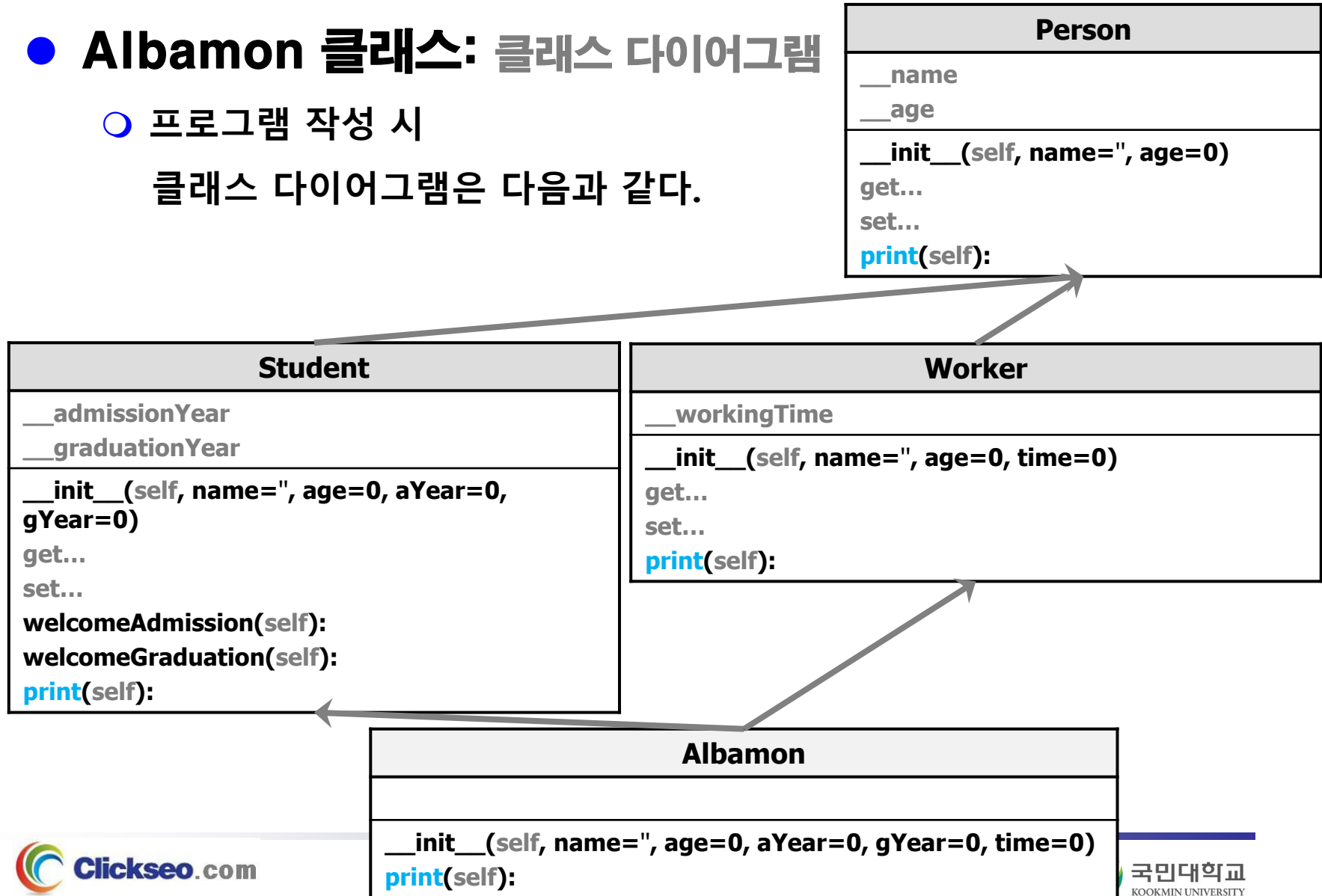
```
aPerson = Person('단군', 100)
bStudent = Student('이도', 25, 2022, 2026)
cWorker = Worker('이순신', 50, 100)
dAlbamon = Albamon('홍길동', 20, 2024, 2028, 120)
```

연습문제 #06: 클래스 다이어그램

- Albamon 클래스: 클래스 다이어그램

- 프로그램 작성 시

클래스 다이어그램은 다음과 같다.



연습문제 #06: 실행 결과

● Albamon 클래스: 실행 결과

```
IDLE Shell 3.11.2
File Edit Shell Debug Options Window Help
Python 3.11.2 (tags/v3.11.2:878ead1, Feb 7 2023, 16:38:35) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\Clickseo\OneDrive\문서\index.py =====
이름: 단군, 나이: 100
{'_Person__name': '단군', '_Person__age': 100}

이름: 이도, 나이: 20
입학 년도: 2023
졸업 년도: 2027
이도님 2023년도 입학 축하 드립니다.
이도님 2027년도 졸업을 축하 드립니다.
{'_Person__name': '이도', '_Person__age': 20, '_Student__admissionYear': 2023, '_Student__graduationYear': 2027}

이름: 이순신, 나이: 50
작업 시간: 100
{'_Person__name': '이순신', '_Person__age': 50, '_Worker__workingTime': 100}

이름: 철수, 나이: 20
입학 년도: 2023
졸업 년도: 2027
작업 시간: 120
{'_Person__name': '철수', '_Person__age': 20, '_Worker__workingTime': 120, '_Student__admissionYear': 2023, '_Student__graduationYear': 2027}
>>>
```

```
aPerson = Person('단군', 100)
bStudent = Student('이도', 25, 2022, 2026)
cWorker = Worker('이순신', 50, 100)
dAlbamon = Albamon('홍길동', 20, 2024, 2028, 120)
```