

컴퓨팅사고와 SW코딩

모듈 실습

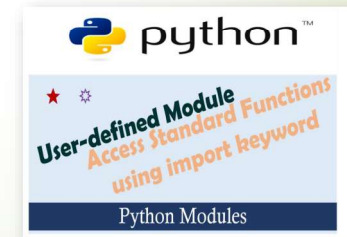
JEE, Jung Eun
rosaliejee@skku.edu

2

Contents

I. 모듈과 패키지

II. 모듈 실습



3

1. 모듈과 패키지 (1/5)

- ▶ 두 정수의 덧셈을 계산해주는 함수와, 곱셈을 계산해주는 함수를 정의하고 있는 사용자 정의 모듈을 만드시오.
 - ▶ 사용자 정의 모듈명 : `addmul.py`
 - ▶ `add()` 함수 : 두 정수의 덧셈을 구하여 반환 (매개변수 있고, 반환값 있음)
 - ▶ `multiply()` 함수 : 두 정수의 곱셈을 구하여 반환 (매개변수 있고, 반환값 있음)
 - ▶ `a.py`
 - ▶ 사용자 정의 모듈(`addmul.py`) 불러오기
 - ▶ 덧셈과 곱셈을 구하는 함수를 사용하여 결과 확인하기
 - ▶ `b.py`
 - ▶ 사용자 정의 모듈(`addmul.py`) 불러오기
 - ▶ 덧셈과 곱셈을 구하는 함수를 사용하여 결과 확인하기
 - ▶ shell 창
 - ▶ 모듈(`addmul.py`)을 다양하게 불러와서 실행하기

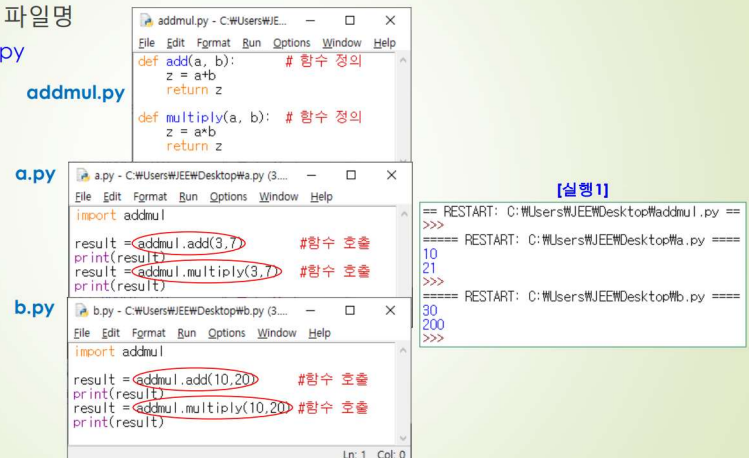
모듈

4

1. 모듈과 패키지 (2/5)

▶ Ex1. 사용할 파일명

- ▶ `addmul.py`
- ▶ `a.py`
- ▶ `b.py`



모듈

1. 모듈과 패키지 (3/5)

Ex2. 사용할 파일명 : addmul.py

[실행2] == RESTART: C:\Users\WJEE\Desktop\addmul.py ==
 >>> import addmul import 모듈명
 >>> addmul.add(3,8) ① 모듈명.함수명(인수1,인수2)
 11
 >>> total = addmul.add ② 변수 = 모듈명.함수명
 total(3,8) 변수(인수1,인수2)
 11
 >>> addmul.multiply(3,8) ③ 모듈명.함수명(인수1,인수2)
 24
 >>> mul = addmul.multiply ④ 변수 = 모듈명.함수명
 mul(3,8) 변수(인수1,인수2)
 24
 >>>

[실행3] == RESTART: C:\Users\WJEE\Desktop\addmul.py ==
 >>> import addmul as ad import 모듈명 as alias
 >>> ad.add(5,82)
 87
 >>> ad.multiply(4,7)
 28
 >>> ad.add(5,9)
 14
 >>> addmul.add(5,9)
 Traceback (most recent call last):
 File "<ipython1#21>", line 1, in <module>
 addmul.add(5,9)
 NameError: name 'addmul' is not defined
 >>>

[실행4] == RESTART: C:\Users\WJEE\Desktop\addmul.py ==
 >>> from addmul import * from 모듈명 import *
 >>> add(55,9)
 64
 >>> multiply(30,8)
 240
 >>>

[실행5] == RESTART: C:\Users\WJEE\Desktop\addmul.py ==
 >>> from addmul import add
 >>> add(11,8) from 모듈명 import 함수명
 19
 >>> from addmul import add, multiply
 >>> add(4,5)
 9
 >>> multiply(4,5)
 20
 >>> from addmul import multiply as mul
 >>> mul(5,90) from 모듈명 import 함수명 as alias
 450
 >>> multiply(5,90)
 450
 >>>

1. 모듈과 패키지 (4/5)

3의 배수(1~50)를 출력하고, 프로그램 실행시간을 구하시오.

time 모듈 불러오기 (파일명 : Ex12_time.py)

```
Ex12_time.py - C:\Users\WJEE...
File Edit Format Run Options Window Help
import time
start = time.time()
print('i =>', end=' ')
while True:
    ...
end = time.time()
print('\n\nstart time :',start)
print('end time :',end)
```

[실행 결과 예시]

```
== RESTART: C:\Users\WJEE\Desktop\time_module.py ==
i => 3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48
start time : 1589345689.7794864
end time : 1589345678.0227587
실행시간 : 8.243273258209229
>>>
== RESTART: C:\Users\WJEE\Desktop\time_module.py ==
i => 3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48
start time : 1589345689.9283273
end time : 1589345694.2375813
실행시간 : 8.309253931045532
>>>
== RESTART: C:\Users\WJEE\Desktop\time_module.py ==
i => 3 6 9 12 15 18 21 24 27 30 33 36 39 42 45 48
start time : 1589345712.4491384
end time : 1589345720.8744724
실행시간 : 8.425333976745605
```

1. 모듈과 패키지 (5/5)

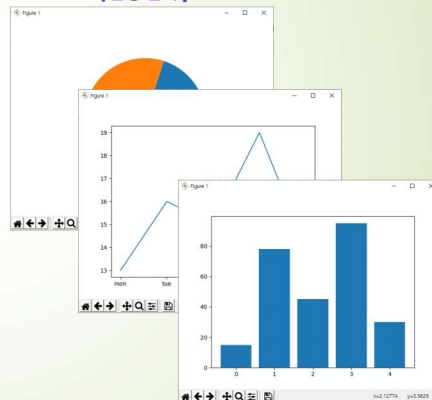
다양한 차트나 플롯(plot)을 그리시오. (파일명 : Ex12_matplotlib.py)

matplotlib 패키지 설치

pyplot 모듈 불러오기

```
Ex12_matplotlib.py - C:\Users\WJEE\Desktop\Ex12_matplotlib.py ...
File Edit Format Run Options Window Help
#matplotlib 패키지 사용
import matplotlib.pyplot as plt
#from matplotlib import pyplot as plt
#pie 차트 그리기
plt.show() #그래프 보이기
#라인 플롯(plot) 그리기
plt.show() #그래프 보이기
#막대 차트 그리기
plt.show() #수직막대차트
plt.show() #그래프 보이기
```

[실행 결과]



2. 모듈 실습 (1/5)

[실습12-1]

사칙연산 프로그램(모듈 사용) (파일명 : op_module.py, op_main.py)

요구사항

1. 두 정수에 대한 사칙연산 결과를 반환하는 함수(매개변수 있고, 반환값 있음)를 정의하고 있는 사용자 정의 모듈을 작성하시오.
 - 사용자 정의 모듈명 : op_module.py
 - 덧셈 함수명 : add() / 뺄셈 함수명 : sub() / 곱셈 함수명 : mul() / 나눗셈 함수명 : div()
2. 두 정수를 입력 받아 사칙연산 함수를 호출한 후 반환값을 받아 출력한다.
 - 파일명 : op_main.py

문제해결 및 알고리즘

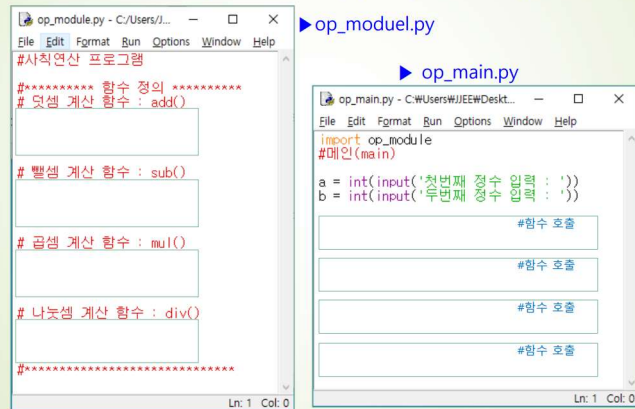
- > [op_module.py] add(), sub(), mul(), div() 함수를 정의한다.

[op_main.py]

- op_module을 import한다.
- 사용자로부터 두 정수를 입력받고 각 함수를 호출하여 결과를 출력한다.

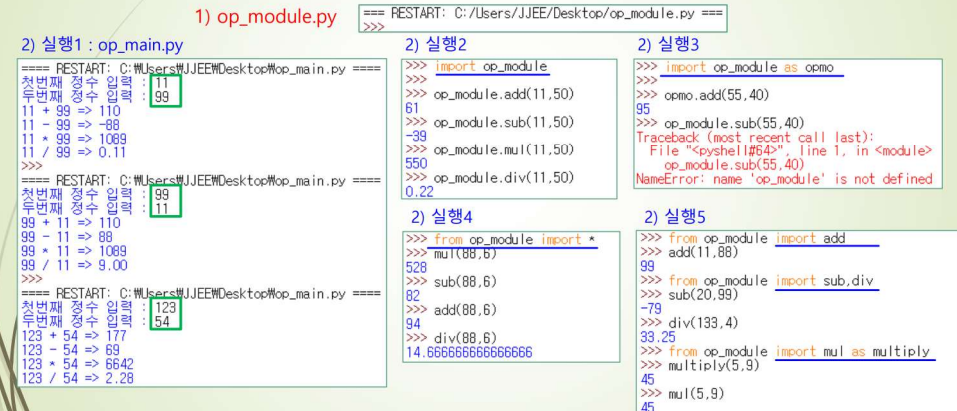
2. 모듈 실습 (2/5)

- ▶ [실습12-1] 파일명 : op_module.py, op_main.py



2. 모듈 실습 (3/5)

- ▶ [실습12-1] 실행 방법



2. 모듈 실습 (4/5)

- ▶ [실습12-2]

- ▶ 점심 메뉴 추천 프로그램 (editor 창 사용 → 파일명 : Ex12-2.py)

▶ 요구사항

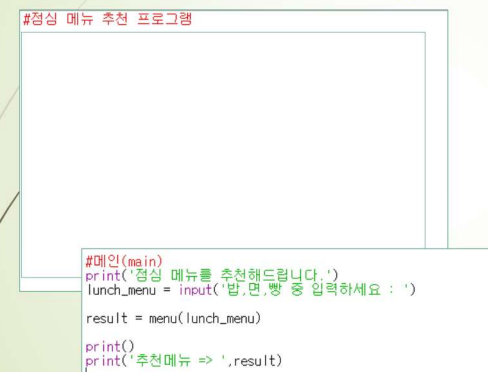
1. 사용자 정의 함수로 메뉴 함수(`menu()` : 매개변수 있고 반환값 있음)를 정의한다.
2. 메뉴 함수에서 밥, 면, 빵, 그 외 메뉴를 각각 리스트 변수로 초기화한다.
 - ▶ 밥(rice): 제육덮밥, 김치볶음밥, 된장찌개 / 면(noodle): 짜장면, 쌀국수, 칼국수
 - ▶ 빵(bread): 와퍼세트, 햄치즈샌드위치, 페퍼로니피자 / 그 외(etc): 떡볶이, 스파게티, 마라샹귀
3. 사용자의 입력을 받아 점심 메뉴를 랜덤하게 추천해준다.

▶ 문제해결 및 알고리즘

- ▶ `random` 모듈을 `import`한다.
- ▶ `menu()` 함수를 정의한다.
- ▶ 사용자의 입력을 받아 점심 메뉴를 추천하여 출력한다.

2. 모듈 실습 (5/5)

- ▶ [실습12-2] 파일명 : Ex12-2.py



[실행 결과 예시]



[w12] 과제 (1/2)

❖ 학번과 이름은 본인 정보를 사용하세요.

➡ [hw1] 홀수/짝수 게임 프로그램을 작성하시오. ➡ 파일명 : w12_hw1.py (editor 창 사용)

■ 요구사항

- ❖ 필요한 변수와 함수명은 **임의로** 작성하고, 프로그램 과정은 **출력 예시**(메시지 자율 작성)를 참고하시오.
- 1. 컴퓨터가 1~100 중 하나의 정수를 **랜덤하게 생성**하여 **반환**하는 **사용자 정의 함수**를 작성하시오.
 - 함수 유형 : 매개변수가 없고, 반환값이 있는 함수, 컴퓨터 생성 숫자 : **random** 모듈 사용
- 2. 컴퓨터가 생성한 정수의 홀수/짝수와 사용자가 입력한 홀수/짝수가 **같은지**를 **비교**하여 **메시지**를 **반환**하는 **사용자 정의 함수**를 작성하시오.
 - 함수 유형 : 매개변수가 있고, 반환값이 있는 함수
- 3. 전체 횟수와 승리한 횟수로 **승률을 계산**하여 **반환**하는 **사용자 정의 함수**를 작성하시오.
 - 함수 유형 : 매개변수가 있고, 반환값이 있는 함수
- 4. **무한 반복문**을 사용하고, 사용자에게 **정수형**으로 숫자를 입력 받으시오.
- 5. 사용자는 **홀수[1]**, **짝수[2]**를 입력하여 게임을 하거나, **0**을 입력하여 프로그램을 종료 하시오.
- 6. **게임 진행 횟수**와 **승리 횟수**, **승률** (소수점 둘째자리까지 표현)을 출력 후, 프로그램을 종료 하시오.
- 7. 여러 횟수(5~15회 정도, 출력예시 참고)로 게임을 진행한 **실행 화면 2가**지를 캡처 하시오.

[w12] 과제 (2/2)

[w12_hw1.py 출력 예시]

```

=== RESTART: C:\Users\HJEE\Desktop\w12_hw1.py ===
학번 : 20201111, 이름 : 지성온

< 홀수/짝수 게임 >

컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<3회> 정답~축복작 (컴퓨터 생성 숫자:21)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<2회> 정답~축복작 (컴퓨터 생성 숫자:33)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<3회> 틀렸습니다. (컴퓨터 생성 숫자:5)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<4회> 정답~축복작 (컴퓨터 생성 숫자:94)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<5회> 틀렸습니다. (컴퓨터 생성 숫자:35)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<6회> 정답~축복작 (컴퓨터 생성 숫자:17)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<7회> 정답~축복작 (컴퓨터 생성 숫자:78)
>>>6회 중 4번 승리(승률: 66.67%)<<<
게임 종료...다음에 또 만나요!
  
```

```

=== RESTART: C:\Users\HJEE\Desktop\w12_hw1.py ===
학번 : 20201111, 이름 : 지성온

< 홀수/짝수 게임 >

컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<3회> 틀렸습니다. (컴퓨터 생성 숫자:3)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<2회> 정답~축복작 (컴퓨터 생성 숫자:42)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<3회> 정답~축복작 (컴퓨터 생성 숫자:77)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<3회> 정답~축복작 (컴퓨터 생성 숫자:17)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<4회> 틀렸습니다. (컴퓨터 생성 숫자:2)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<5회> 틀렸습니다. (컴퓨터 생성 숫자:42)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<6회> 틀렸습니다. (컴퓨터 생성 숫자:3)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<12회> 틀렸습니다. (컴퓨터 생성 숫자:47)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<14회> 틀렸습니다. (컴퓨터 생성 숫자:3)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<15회> 정답~축복작 (컴퓨터 생성 숫자:86)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 0
>>>15회 중 7번 승리(승률: 46.67%)<<<
게임 종료...다음에 또 만나요!
  
```

```

컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<3회> 틀렸습니다. (컴퓨터 생성 숫자:94)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<2회> 틀렸습니다. (컴퓨터 생성 숫자:35)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<3회> 정답~축복작 (컴퓨터 생성 숫자:44)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<4회> 정답~축복작 (컴퓨터 생성 숫자:8)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<5회> 정답~축복작 (컴퓨터 생성 숫자:4)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<6회> 틀렸습니다. (컴퓨터 생성 숫자:17)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<7회> 틀렸습니다. (컴퓨터 생성 숫자:62)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 2
<8회> 정답~축복작 (컴퓨터 생성 숫자:94)
컴퓨터가 1~100 중 하나의 정수를 생성하였습니다.
홀수[1]/짝수[2] 선택(종료:0)> 1
<9회> 틀렸습니다. (컴퓨터 생성 숫자:96)
>>>9회 중 4번 승리(승률: 33.33%)<<<
게임 종료...다음에 또 만나요!
  
```

❖ 12주차 과제 제출

➡ [과제]

1) [hw1]

- editor 창에서 코드 작성(파일명 : w12_hw1.py) : 화면 캡처하기
- 작성한 코드 실행하여 결과 확인 : 화면 캡처하기

■ 캡처한 화면을 한글(or 워드)파일에 붙여넣기

- 한글(or 워드)파일명 : 이름_학번_w12.hwp(or docx)

■ 최종 제출

- 압축파일명 : 이름_학번_w12.zip
 - w12_hw1.py, 이름_학번_w12.hwp(or docx) ← 2개 파일 압축
- ❖ 한 개의 압축 파일(이름_학번_w12.zip) 제출하기!!!

