



6. 모듈화 프로그래밍: 일 처리는 명확하게

순천향대학교 컴퓨터공학과
이 상 정

프로그래밍 기초

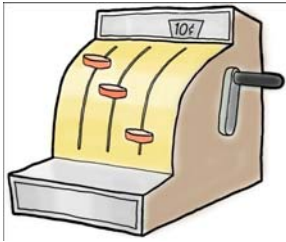
학습 내용

- 한 번 작성된 코드는 재사용되고 공유
- 작성된 함수가 재사용을 위해선 요구사항에 맞게 수정
- 프로그램의 재사용 및 공유를 쉽게 하기 위해 모듈화 프로그래밍 기법을 사용



헬스클럽 예

- 헤드 퍼스트 헬스클럽 새 CEO는 새로운 POS 시스템을 구축하고자 함
 - POS(Point-Of-Sale) 시스템은 현금 출납기처럼 작동하는 컴퓨터
 - 먼저 커피바 부터 시도



트랜잭션(transaction) 파일

- POS 시스템은 transactions.txt 파일에 신용카드 거래를 기록
 - 하루 영업이 끝나면 이 파일을 은행에 전송
 - 은행은 신용카드 계좌에서 헬스클럽의 은행 계좌로 돈을 이체
- 거래은행의 데이터 파일 형식
 - 포맷된 데이터(formatted data)
 - 처음 16글자 신용카드 번호
 - 다음 7글자 가격 (\$1.75 -> 0000175)
 - 다음 16글자는 판매한 물건 설명

62189672574056180000220	LATTE
75362847324839070000120	TEA
93343374684577850000150	DONUT
42657423454758780000095	GRAIN BAR



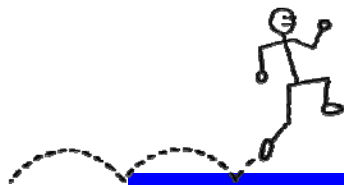
스트링을 포맷하기

□ 스트링 포맷 실행 예

```
>>> print("There are %5d %s available" % (17, "donuts"))
There are 17 donuts available
```

• 설명

- 스트링(" ... ") 뒤에 %가 오면 스트링 포맷을 처리
- % 다음에 나오는 값들(17, "donuts")은 순서대로 포맷된 스트링에 삽입
 - 스트링 안의 %에 대응하여 삽입
- %5d
 - 숫자 포맷 지정
 - 17을 5자리 10진수로 삽입, 빈자리는 공백
- %s
 - 스트링 포맷 지정
 - donuts을 스트링으로 삽입, 길이 지정 생략 시 스트링 길이만큼만 삽입



스트링 포맷 예

```
>>> print("%7d" % (11232/3))
3744          공백을 빈칸으로 채우고 계산 결과(정수) 출력

>>> print("%07d" % (11232/3))
0003744      공백을 0으로 채우고 계산 결과(정수) 출력

>>> print("%x" % 127)
7f           16진수로 출력

>>> print("%s\n" % "Banna swirl")
Banna swirl

>>> print("%s %f" % ("Value is", 16.0 ** 0.5))
Value is 4.000000   스트링 다음에 계산결과(부동소수점수, 실수) 출력

>>> print("%s %e" % ("Value is", 16.0 ** 0.5))
Value is 4.000000e+00   스트링 다음에 계산결과(실수) 지수형으로 출력

>>> print("%s is $%4.2f" % ("Popsicle", 1.754))
Popsicle is $1.75   부동소수점 수를 전체 4자리 소수점 이하 2자리로 출력
```

헬스클럽 코드: 커피바 POS [실습 1]

```
def save_transaction(price, credit_card, description):
    file = open("transactions.txt", "a")
    file.write("%16s%07d%16s\n" % (credit_card, price * 100, description))
    file.close()

items = ["DONUT", "LATTE", "FILTER", "MUFFIN"]
prices = [1.50, 2.20, 1.80, 1.20]
running = True

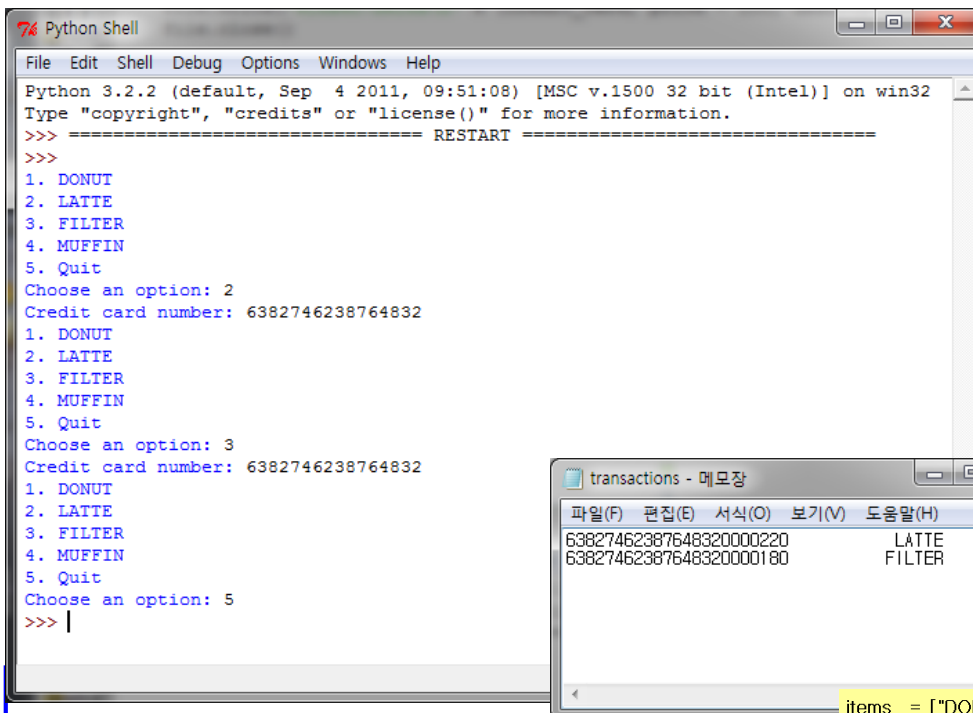
while running:
    option = 1
    for choice in items:
        print(str(option) + ". " + choice)
        option = option + 1
    print(str(option) + ". Quit")
    choice = int(input("Choose an option: "))
    if choice == option:
        running = False
    else:
        credit_card = input("Credit card number: ")
        save_transaction(prices[choice-1], credit_card, items[choice - 1])
```

순차

프로그래밍



시험주행

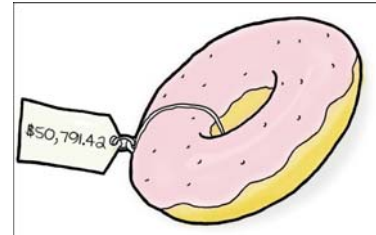


```
items = ["DONUT", "LATTE", "FILTER", "MUFFIN"]
prices = [1.50, 2.20, 1.80, 1.20]
```

새로운 레코드 포맷과 새 코드

☐ 헬스클럽이 거래 은행을 변경

- 새 거래 은행의 새 레코드 포맷
 - 처음 7글자 가격
 - 다음 16글자 신용카드 번호
 - 다음 16글자는 판매한 물건 설명



50791428746281510000150	DONUT
00035005002454912715921	WORKOUT

☐ 업데이트되지 않은 기존 커피바 프로그램 적용

- 도넛 한 개 5만불 요청하여 은행으로부터 거래 중지

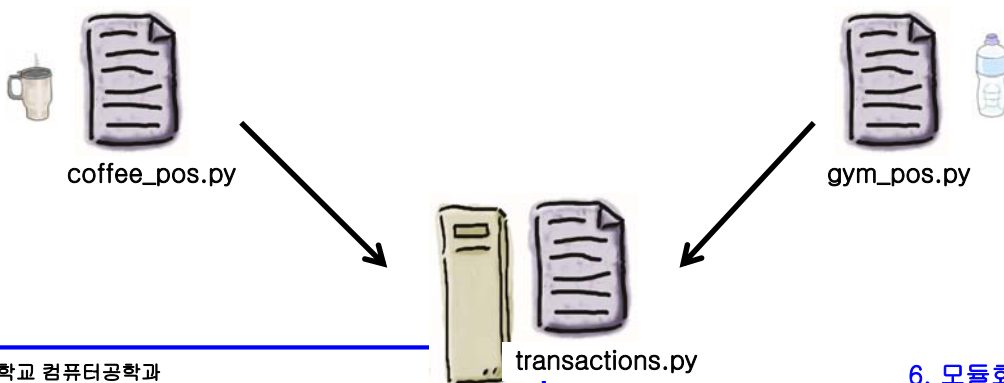
☐ 수정된 커피바 POS와 새로운 헬스클럽 POS가 필요!!



모듈화된 코드

☐ 프로그램을 여러 모듈로 분할

- 모듈이 프로그램 간에 공유
- 커피바 POS (`coffee_pos.py`), 헬스클럽 POS (`gym_pos.py`), 트랜잭션 코드 (`transaction.py`)로 분할
- `transactions.py`가 공유
 - 다른 프로그램에서는 아래 명령으로 참조
`from transactions import *`



커피바 코드: coffee_pos.py [실습 2]

```

from transactions import *

items = ["DONUT", "LATTE", "FILTER", "MUFFIN"]
prices = [1.50, 2.20, 1.80, 1.20]
running = True

while running:
    option = 1
    for choice in items:
        print(str(option) + ". " + choice)
        option = option + 1
    print(str(option) + ". Quit")
    choice = int(input("Choose an option: "))
    if choice == option:
        running = False
    else:
        credit_card = input("Credit card number: ")
        save_transaction(prices[choice-1], credit_card, items[choice - 1])

```

순천향대학

그레밍

헬스클럽 코드: gym_pos.py [실습 2]

```

from transactions import *

items = ["WORKOUT", "WEIGHTS", "BIKES"]
prices = [35.0, 10.0, 8.0]
running = True

while running:
    option = 1
    for choice in items:
        print(str(option) + ". " + choice)
        option = option + 1
    print(str(option) + ". Quit")
    choice = int(input("Choose an option: "))
    if choice == option:
        running = False
    else:
        credit_card = input("Credit card number: ")
        save_transaction(prices[choice-1], credit_card, items[choice - 1])

```

순천향대학

그레밍

트랜잭션 코드: transactions.py [실습 2]

```
def save_transaction(price, credit_card, description):
    file = open("transactions.txt", "a")
    file.write("%07d%16s%16s\n" % (price * 100, credit_card, description))
    file.close()
```



시험주행

```
Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep 4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
1. WORKOUT
2. WEIGHTS
3. BIKES
4. Quit
Choose an option: 1
Credit card number: 6432425412474321
1. WORKOUT
2. WEIGHTS
3. BIKES
4. Quit
Choose an option: 4
>>> |
```



```
Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep 4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
1. DONUT
2. LATTE
3. FILTER
4. MUFFIN
5. Quit
Choose an option: 1
Credit card number: 7649463856424326
1. DONUT
2. LATTE
3. FILTER
4. MUFFIN
5. Quit
Choose an option: 5
>>> |
```

파일(F)	편집(E)	서식(O)	보기(V)	도움말(H)
00035006432425412474321				WORKOUT
00001507649463856424326				DONUT

헬스클럽의 새로운 요구사항

- 더 많은 고객 유치를 위해 한 달 동안 커피바의 음료의 10% 할인 행사
- coffee_pos.py 코드 수정 대신 할인 가격을 계산하는 promotion.py 라는 새 모듈 작성

```
def discount(price):
    return 0.9*price
```



커피바 코드: coffee_pos.py [실습 3]

```
from transactions import *
from promotion import *

items = ["DONUT", "LATTE", "FILTER", "MUFFIN"]
prices = [1.50, 2.20, 1.80, 1.20]
running = True

while running:
    option = 1
    for choice in items:
        print(str(option) + ". " + choice)
        option = option + 1
    print(str(option) + ". Quit")
    choice = int(input("Choose an option: "))
    if choice == option:
        running = False
    else:
        credit_card = input("Credit card number: ")
        new_price = discount(prices[choice-1])
        save_transaction(new_price, credit_card, items[choice - 1])
```




```

Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep 4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
1. DONUT
2. LATTE
3. FILTER
4. MUFFIN
5. Quit
Choose an option: 2
Credit card number: 3489203918924782
1. DONUT
2. LATTE
3. FILTER
4. MUFFIN
5. Quit
Choose an option: 5
>>> |

```

파일(F)	편집(E)	서식(O)	보기(V)	도움말(H)
0003500643242541	2474321			WORKOUT
000015076494638564	24326			DONUT
0000198348920391	8924782			LATTE

라떼 \$2.2 이
10% 할인되어 \$1.98

추가 요구사항: 스타버즈 할인 추가

- ❑ 스타버즈 할인 카드 제시 고객에 추가로 5% 할인
- ❑ 스타버즈 제공 코드 활용
 - starbuzz.py

```

# Official Starbuzz Discount Module
# Copyright(c) Starbuzz Corporation
# All Rights Reserved.
# This function calculates a 5% discount on a price
def discount(price):
    return 0.95 * price

```



함수 이름 완전 수식

- 할인 함수의 이름이 같아서 충돌
 - `promotion.py`, `discount` 함수
 - `starbuzz.py`, `discount` 함수

- 완전수식(Fully Qualified Name, FQN) 적용
 - 함수이름 앞에 모듈이름을 붙임
 - импорт 방식 변경
 - 예


```
import promotion
promotion.discount(1.75)
```



```
from transactions import *
import promotion
import starbuzz
```

```
items = ["DONUT", "LATTE", "FILTER", "MUFFIN"]
prices = [1.50, 2.20, 1.80, 1.20]
running = True
```

```
while running:
    option = 1
    for choice in items:
        print(str(option) + ". " + choice)
        option = option + 1
    print(str(option) + ". Quit")
    choice = int(input("Choose an option: "))
    if choice == option:
        running = False
    else:
        credit_card = input("Credit card number: ")
        price = promotion.discount(prices[choice - 1])
        if input("Starbuzz card? ") == "Y":
            price = starbuzz.discount(price)
        save_transaction(price, credit_card, items[choice - 1])
```

수정된 커피바 코드:
`coffee_pos.py` [실습 4]



```

Python Shell
File Edit Shell Debug Options Windows Help
Python 3.2.2 (default, Sep 4 2011, 09:51:08) [MSC v.1500 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> ===== RESTART =====
>>>
1. DONUT
2. LATTE
3. FILTER
4. MUFFIN
5. Quit
Choose an option: 4
Credit card number: 5413765853576543
Starbuzz card? N
1. DONUT
2. LATTE
3. FILTER
4. MUFFIN
5. Quit
Choose an option: 2
Credit card number: 5413765853576543
Starbuzz card? Y
1. DONUT
2. LATTE
3. FILTER
4. MUFFIN
5. Quit
Choose an option: 5
>>> |

```



머핀 \$1.2 이
10% 할인되어 \$1.08

파일(F)	편집(E)	서식(O)	보기(V)	도움말(H)
00035006432425412474321				WORKOUT
00001507649463856424326				DONUT
00001983489203918924782				LATTE
00001085413765853576543				MUFFIN
00001885413765853576543				LATTE

라떼 \$2.2 이
10% 할인된 \$1.98에서
5% 추가되어 \$1.88



프로그래밍 도구

- **스트링 포맷**을 이용하면 스트링을 포맷하는 포맷 지시자를 사용
- **포맷 지시자**는 형, 길이, 패딩(공간을 채우는 문자)을 지정
- 코드를 다른 파일로 분할하는 것을 **모듈화 프로그래밍**
- 모듈화된 코드는 **프로그램들 간의 공유**가 수월
- 모듈화된 코드를 작성하는 것은 **자신의 라이브러리**를 만드는 것
- **완전 수식 명칭(Fully Qualified Name, FQN)**은 모듈 이름을 포함한 함수 이름



프로그래밍 도구상자 (2)

□ 파이썬 도구

- 모듈 이름 없이 함수를 호출하기 위해서는 ‘`from …… import *`’ 형태를 사용
- FQN을 사용하여 모듈 이름과 함께 함수 이름을 지정하기 위해서는 ‘`import ……`’ 형태를 사용

6장 실습

[실습 1] 헬스클럽 코드: 커피바 POS

[실습 2] 모듈화된 코드

[실습 3] 할인이 추가된 커피바 코드

[실습 4] 수정된 커피바 코드

1. [실습 1] ~ [실습 4]의 프로그램 작성 및 실행 과정
2. 앞의 “[실습 4] 수정된 커피바 코드”를 아래와 같이 수정하여라.
 - 모듈 menu에 메뉴를 화면에 출력하는 코드를 함수 menu_list()로 작성하고, 이를 호출하여 실행하라.
 - Menu_list() 함수를 모듈화하여, 헬스클럽 코드에서도 호출`
3. 앞에서 배운 내용을 사용한 임의의 프로그램 작성
 - 프로그램 설명
 - 프로그램 소스
 - 실행 결과