

6장 딕셔너리, 튜플, 집합 연습문제

6.1 다음과 같은 파이썬 딕셔너리 구조의 프로그램을 실행할 적에, 다음 밑줄 친 곳에 들어갈 알맞은 내용은 무엇인가? (5)번 문제의 출력을 위하여 len() 함수를 사용하시오.

<pre>>>> price = {'김밥': 5000, '어묵': 3000, '떡볶이': 2000} >>> price['김밥'] (1) _____ >>> price['김밥'] = 6000 >>> price (2) _____</pre>	<pre>>>> price.values() (3) _____ >>> price.keys() (4) _____ >>> (5) _____ 이 식당의 메뉴 개수는 3개 입니다.</pre>
---	--

6.2 문제 6.1의 price 딕셔너리에 ('순대' : 4500) 메뉴를 추가하는 코드를 (1)에 넣으시오. (2)에는 그 후의 출력 결과를 적으시오. (3)은 price['순대']를 사용하여 출력하시오. (4)는 반드시 len() 함수를 이용하시오.

<pre>>>> (1) _____ >>> price (2) _____</pre>	<pre>>>> (3) _____ 순대의 가격은 4500원 입니다. >>> (4) _____ 이 식당의 메뉴 개수는 4개 입니다.</pre>
--	--

6.3 동윤이가 새로 문을 열고 운영하는 커피 가게에는 Americano, Ice Americano, Cappuccino, Caffe Latte, Espresso의 5가지 메뉴가 있으며, 각 메뉴의 가격은 각각 3,000원, 3,500원, 4,000원, 4,500원, 3,600원이다. 이 목록을 menu라는 딕셔너리로 작성하라.

- 1) 이 menu의 내용을 for key in menu: 구문을 사용하여 다음과 같이 출력하여라.
- 2) 사용자로부터 다음과 같은 주문을 받은 후 주문한 내용이 메뉴에 있는지 구하는 프로그램을 작성하라(힌트: in 연산자를 사용할 것).

<p>1) 실행 결과]</p> <pre>Americano 가격 : 3,000원 Ice Americano 가격 : 3,500원 Cappuccino 가격 : 4,000원 Cafe Latte 가격 : 4,500원 Espresso 가격 : 3,600원</pre>	<p>2) 실행 결과]</p> <pre>Americano 가격 : 3,000원 Ice Americano 가격 : 3,500원 Cappuccino 가격 : 4,000원 Cafe Latte 가격 : 4,500원 Espresso 가격 : 3,600원 위의 메뉴중 하나를 선택하세요: Espresso Espresso는 3,600원 입니다. 결제를 부탁드립니다. [메뉴에 없을 경우 메시지] 위의 메뉴중 하나를 선택하세요: Fan Cake 미안합니다. Fan Cake는 메뉴에 없습니다.</pre>
--	---

6.4 다음의 코드에서 잘못된 부분은 무엇인가?

<pre>>>> t = (10, 20, 30, 40) >>> t.append(50) >>> t = (10, 20, 30, 40) >>> t.remove(50)</pre>	<pre>>>> t = (10, 20, 30, 40) >>> t[0] = 0</pre>
--	--

6.5 다음의 실행 결과를 적으시오.

```
>>> t = (10, 20, 30, 40, 50, 60)
>>> t[0]
(1) _____
>>> t[0:2]
(2) _____
>>> t[1:]
(3) _____
```

```
>>> t[3]
(4) _____
>>> t[1:2]
(5) _____
>>> t[::-1]
```

6.6 다음 코드의 실행 결과는 무엇인가?

```
t1 = 'a', 'b', 'c'
t2 = ('a', 'b', 'c')
t3 = ('d', 'e')
t1 == t2
1) _____
t2 + t3
2) _____
```

```
my_list = []
for e1 in t1:
    for e2 in t3:
        my_list.append(e1+e2)
my_list
3) _____
```

6.7 주어진 튜플은 10일 동안 일일 매장 매출을 기록한 것이다. 매출이 전날보다 감소한 날이 며칠인지 출력하는 코드를 작성하라.

실행 결과]

일일 매출 기록: (100, 121, 120, 130, 140, 120, 122, 123, 190, 125)
지난 10일 동안 전일대비 매출이 감소한 날은 3일입니다.

실행 결과

6.8 주어진 튜플 tup에서 한 번 이상 나타나는 원소를 찾아 집합 dup에 넣어서 이를 출력하라. 다음의 튜플에서 1, 2, 4는 2번, 9는 3번 나타난다. 따라서 결과는 {1, 2, 4, 9}가 된다(힌트 : 비어있는 집합을 만드는 것은 dup = set()과 같이 할 수 있다. 이 dup 집합에 tup의 원소를 하나하나 추출한 다음 tup.count(item) > 1과 같이 item이 2개 이상 원소를 찾아서 dup에 추가하여 구할 수 있다).

실행 결과]

주어진 튜플은: (1, 2, 5, 4, 3, 2, 9, 1, 4, 7, 8, 9, 9)
중복 원소는: {1, 2, 4, 9}

6.9 중복 요소를 가진 튜플 tup에서 중복 요소가 하나만 나타나도록 수정한 튜플을 생성하라. 예들 들어 주어진 튜플에는 2가 두 번 나타나지만 중복을 제거하면 아래와 같이 한 번 나타난다(힌트 : 원래의 튜플 tup를 집합 함수 set()에 넣어 집합 객체로 만든 후 다시 튜플 함수 tuple()에 넣도록 한다).

실행 결과]

주어진 튜플: (1, 2, 5, 4, 3, 2, 1, 4, 7, 8, 9, 9, 3, 7, 3)
중복 제거 튜플: (1, 2, 3, 4, 5, 7, 8, 9)

6.10 주어진 튜플에서 가장 많이 나타나는 요소를 출력하라. 이때 가장 많이 나타나는 요소가 두 개 이상일 경우 이들 중에서 가장 큰 값을 출력하여라.

실행 결과]

주어진 튜플은: (1, 2, 5, 4, 3, 2, 1, 4, 7, 8, 9, 9, 3, 7, 3)

가장 많이 나타나는 원소는: 3

6.11 다음과 같은 리스트에는 원소를 가지는 항목과 원소를 가지지 않는 빈 튜플, 빈 문자열, 빈 리스트가 존재한다. 이 리스트에서 빈 튜플과 빈 문자열, 빈 리스트 만을 제거한 결과를 출력하는 코드를 작성하라. 단 (0,) 튜플은 빈 튜플이 하나 있는 튜플로 간주하여 제거하지 않는다.

실행 결과]

주어진 리스트 : [0, (1,), [], 'abc', 0, 0, (1,), ('a'), ('a', 'b'), (0,), '']

빈 원소를 제거한 결과 : [(1,), 'abc', (1,), ('a'), ('a', 'b'), (0,)]

6.12 튜플 데이터를 이용하면 임시변수를 사용하지 않고 두 변수의 값을 서로 교환(swap)할 수 있다. 이 교환 방법을 이용하여 다음과 같이 주어진 리스트에서 가장 큰 값 12를 제일 마지막으로 옮겨 놓는 프로그램을 작성해 보라.(제약 조건 : 리스트의 원소를 살펴보기 위해 사용하는 for 문의 인덱스 변수 i를 제외하고는, 주어진 리스트 이외에 다른 어떤 변수도 추가로 사용해서는 안 된다)

실행 결과]

주어진 리스트는 = [5, 6, 3, 9, 2, 12, 3, 8, 7]

가장 큰 수를 마지막으로 옮긴 결과: [5, 3, 6, 2, 9, 3, 8, 7, 12]

6.13 6.12 문제를 개선하여 처음에는 가장 큰 수를 제일 마지막으로 옮기고, 두 번째 단계는 끝으로 옮겨진 원소를 제외한 리스트에서 가장 큰 수를 뒤에서 두 번째 위치로 옮기며, 세 번째 단계에서는 뒤의 두 원소를 제외한 나머지 리스트에서 가장 큰 수를 뒤에서 세 번째로 옮긴다. 이를 일반화 하면 k 번째 단계에서는 뒤에서 k-1 개 원소를 제외하고, 남은 리스트에서 가장 큰 원소를 뒤에서 부터 k 번째 위치에 옮겨 놓는 일을 처리할 리스트가 남지 않을 때까지 반복하면 큰 수부터 차례로 뒤에 쌓이게 된다. 이를 거품(bubble) 정렬이라고 한다. 이를 구현해 보라(힌트 : sorted() 함수나. 리스트의 sort() 메소드를 사용하지 말고 구현하시오).

실행 결과]

정렬전 리스트 : [5, 6, 3, 9, 2, 12, 3, 8, 7]

정렬된 리스트 : [2, 3, 3, 5, 6, 7, 8, 9, 12]

6.14 앞서 작성한 버블 정렬 문제에서 정렬되기 전의 원소를 왼쪽에 두고 정렬된 원소를 오른쪽 리스트에 두어 각 단계별로 정렬이 진행되는 과정을 다음과 같이 시각화하시오.

실행 결과]

정렬전 리스트 : [5, 6, 31, 9, 2, 12, 13, 8, 7]

1 단계 : [5, 6, 9, 2, 12, 13, 8, 7], [31]

2 단계 : [5, 6, 2, 9, 12, 8, 7], [13, 31]

3 단계 : [5, 2, 6, 9, 8, 7], [12, 13, 31]

4 단계 : [2, 5, 6, 8, 7], [9, 12, 13, 31]

5 단계 : [2, 5, 6, 7], [8, 9, 12, 13, 31]

6 단계 : [2, 5, 6], [7, 8, 9, 12, 13, 31]

7 단계 : [2, 5], [6, 7, 8, 9, 12, 13, 31]

8 단계 : [2], [5, 6, 7, 8, 9, 12, 13, 31]

정렬된 리스트 : [2, 5, 6, 7, 8, 9, 12, 13, 31]

6.15 어떤 튜플이 주어졌을 때, 처음에는 튜플 전체를, 그 다음에는 마지막 한 원소를 제외하고, 그리고 그 다음 단계에는 마지막 두 원소를 제외하고 출력하는 코드를 슬라이싱을 이용하여 작성해 보라.

실행 결과]

```
(4, 5, 2, 3, 8, 1, 9, 0)
(4, 5, 2, 3, 8, 1, 9)
(4, 5, 2, 3, 8, 1)
(4, 5, 2, 3, 8)
(4, 5, 2, 3)
(4, 5, 2)
(4, 5)
(4,)
```

6.16 튜플을 원소로 가지는 student_tuple 튜플이 있다. 이 튜플의 원소 튜플은 (학번, 이름, 전화번호)로 이루어져 있다. 데이터의 예는 아래와 같다. 이를 이용하여 (학번:이름)의 딕셔너리를 만들고 학번을 입력하면 학생의 이름이 나오도록 하라.

주어진 데이터

```
student_tuple = (('191101', '홍길동', '010-123-45xx'), ('191102', '임꺽정', '010-223-45xx'),
                  ('191103', '장길산', '010-323-45xx'))
```

(힌트 : 튜플을 이용하여 딕셔너리를 만드는 방법은 다음과 같다.)

```
a = ('key1', 'value1')
b = ('key2', 'value2')
dictionary = dict((a, b))
print(dictionary)
```

이 프로그램은 다음과 같이 학번을 입력하면 이름을 출력하는 기능을 가지며, 학번 값으로 음수가 입력되면 프로그램이 종료된다.

실행 결과]

```
학생 정보: {'191101': '홍길동', '191102': '임꺽정', '191103': '장길산'}
학번을 입력하세요 : 191102
191102번 학생은 임꺽정입니다.
학번을 입력하세요 : 191106
해당 학번의 학생이 없습니다.
학번을 입력하세요 : -1
프로그램을 종료합니다.
```

6.17 다음 코드의 수행 결과를 적으시오. 이 중에서 에러가 발생하는 부분은 어느 부분인가?

```
>>> s1 = set('abcd')
```

```
>>> s1
```

```
(1) _____
```

```
>>> s2 = set('defg')
```

```
>>> s2
```

```
(2) _____
```

```
>>> s1 == s2
```

```
(3) _____
```

```
>>> s1 + s2
```

```
(4) _____
```

```
>>> s1 & s2
```

```
(5) _____
```

6.18 다음과 같은 집합에 대한 연산을 적용할 적에, 다음 밑줄 친 부분에 들어갈 알맞은 결과는 무엇인가?

>>> s1 = {0, 1, 2, 3, 4, 5}	
>>> s2 = {3, 4, 5, 6, 7}	
>>> s1 & s2	>>> s1 - s2
(1) _____	(4) _____
>>> s1 s2	>>> s1 ^ s2
(2) _____	(5) _____
>>> s2 - s1	>>> 2 in s1
(3) _____	(6) _____

6.19 어떤 문자열을 뒤집었을 때에 원래의 문자열과 같은 것을 회문(palindrome)이라고 한다. 예를 들어 'mom'은 문자열을 뒤집어도 'mom'이 되므로 회문이다. 어떤 문자열을 입력하면 이 문자열이 회문인지 아닌지 검사하는 코드를 작성하라.

실행 결과] 문자열을 입력하시오 : racecar 회문입니다.	문자열을 입력하시오 : racing 회문이 아닙니다.
--	----------------------------------

6.20 학생의 이름과 국어, 수학, 과학 성적이 하나의 튜플을 이룬다. 이렇게 튜플로 표시된 학생의 성적을 원소로 하는 튜플 scores가 아래와 같이 있다.

scores = (('박동규', 88, 95, 90), ('강영민', 85, 90, 95), ('박동민', 70, 90, 80), ('홍승주', 90, 90, 95))

1) 이러한 튜플을 zip() 함수를 사용하여 언패킹하여 수학 성적만 추출한 뒤에 수학 성적의 평균을 구하는 코드를 작성하라.

실행 결과] 학생들의 수학 성적의 평균은 91.25입니다.

2) 이러한 튜플을 언패킹하여 수학, 과학 성적만 추출한 뒤에 수학과 과학 성적의 평균을 구하는 코드를 작성하라.

실행 결과

실행 결과] 학생들의 수학과 과학 성적의 평균은 90.62입니다.

3) 이러한 튜플에 대하여 학생의 이름과 국어, 수학, 과학 성적의 평균 점수를 student_dic이라는 딕셔너리 구조에 넣은 후 다음과 같이 이름과 평균성적을 출력하여라.

실행 결과]	
이름 평균성적	

박동규 91.00	
강영민 90.00	
박동민 80.00	
홍승주 91.67	

6.21 string 모듈에 정의된 ascii_uppercase는 다음과 같이 알파벳 대문자의 목록이 들어있으며 string.ascii_uppercase로 그 목록을 다음과 같이 문자열 형태로 볼 수 있다.

실행 결과]

ABCDEFGHIJKLMNOPQRSTUVWXYZ
BCDEFGHIJKLMNOPQRSTUVWXYZA
CDEFGHIJKLMNOPQRSTUVWXYZAB
DEFGHIJKLMNOPQRSTUVWXYZABC
EFGHIJKLMNOPQRSTUVWXYZABCD
FGHIJKLMNOPQRSTUVWXYZABCDE
GHIJKLMNOPQRSTUVWXYZABCDEF
HIJKLMNOPQRSTUVWXYZABCDEFG
JKLMNOPQRSTUVWXYZABCDEFGH
KLMNOPQRSTUVWXYZABCDEFGHI
LMNOPQRSTUVWXYZABCDEFGHIJ
MNOPQRSTUVWXYZABCDEFGHIJK
NOPQRSTUVWXYZABCDEFGHIJKL
OPQRSTUVWXYZABCDEFGHIJKLMN
PQRSTUVWXYZABCDEFGHIJKLMNO
QRSTUVWXYZABCDEFGHIJKLMNOP
RSTUVWXYZABCDEFGHIJKLMNOPQ
STUVWXYZABCDEFGHIJKLMNOPQR
TUVWXYZABCDEFGHIJKLMNOPQRS
UVWXYZABCDEFGHIJKLMNOPQRST
VWXYZABCDEFGHIJKLMNOPQRSTU
WXYZABCDEFGHIJKLMNOPQRSTUV
XYZABCDEFGHIJKLMNOPQRSTUVW
YZABCDEFGHIJKLMNOPQRSTUVWX
ZABCDEFGHIJKLMNOPQRSTUVWXY