

04-2 효율적인 데이터 관리 2: **딕셔너리**

월간 판매량 보고서 작성하기: **리스트 활용**

편의점 상품별 월간 판매량 보고서 작성하기

상품명	판매월	판매량			
		1주	2주	3주	4주
드링킹 요구르트	1	15	16	20	29
	2	8	11	25	21
	3	17	9	17	24

yo_qty_1_1 = 15

yo_qty_1_2 = 16

yo_qty_1_3 = 20

yo_qty_1_4 = 29

yo_qty_2_1 = 8

yo_qty_2_2 = 11

yo_qty_2_3 = 25

yo_qty_2_4 = 21

yo_qty_3_1 = 17

yo_qty_3_2 = 9

yo_qty_3_3 = 17

yo_qty_3_4 = 24

편의점 상품별 월간 판매량 보고서 작성하기

상품명	판매월	판매량			
		1주	2주	3주	4주
드링킹 요구르트	1	15	16	20	29
	2	8	11	25	21
	3	17	9	17	24

```
yo_qty_1 = [15, 16, 20, 29]
```

```
yo_qty_2 = [8, 11, 25, 21]
```

```
yo_qty_3 = [17, 9, 17, 24]
```


손코딩 실습

리스트를 사용해서 월간 판매량 보고서 작성

직접 해보는 손코딩



소스 코드 data03.py

```
01 yo_qty_1 = [15, 16, 20, 29]
```

```
02 yo_qty_2 = [8, 11, 25, 21]
```

```
03 yo_qty_3 = [17, 9, 17, 24]
```

리스트에 저장된 숫자 데이터의 합계를 계산

```
04
```

```
05 print("드링킹 요구르트 1월 판매량 합계: " + str(sum(yo_qty_1)))
```

```
06 print("드링킹 요구르트 2월 판매량 합계: " + str(sum(yo_qty_2)))
```

```
07 print("드링킹 요구르트 3월 판매량 합계: " + str(sum(yo_qty_3)))
```



```
yo_qty_1 = [15, 16, 20, 29]
```

```
yo_qty_2 = [8, 11, 25, 21]
```

```
yo_qty_3 = [17, 9, 17, 24]
```

```
print("드링킹 요구르트 1월 판매량 합계 : " + str(sum(yo_qty_1)))
```

```
print("드링킹 요구르트 2월 판매량 합계 : " + str(sum(yo_qty_2)))
```

```
print("드링킹 요구르트 3월 판매량 합계 : " + str(sum(yo_qty_3)))
```


4~6월 판매량 데이터 추가

상품명	판매월	판매량			
		1주	2주	3주	4주
드링킹 요구르트	4	19	29	29	4
	5	5	20	18	12
	6	21	7	27	14

```
yo_qty_4 = [19, 29, 29, 4]
```

```
yo_qty_5 = [5, 20, 18, 12]
```

```
yo_qty_6 = [21, 7, 27, 14]
```



```
yo_qty_1 = [15, 16, 20, 29]
```

```
yo_qty_2 = [8, 11, 25, 21]
```

```
yo_qty_3 = [17, 9, 17, 24]
```

```
yo_qty_4 = [19, 29, 29, 4]
```

```
yo_qty_5 = [5, 20, 18, 12]
```

```
yo_qty_6 = [21, 7, 27, 14]
```

```
print("드링킹 요구르트 1월 판매량 합계 : " + str(sum(yo_qty_1)))
```

```
print("드링킹 요구르트 2월 판매량 합계 : " + str(sum(yo_qty_2)))
```

```
print("드링킹 요구르트 3월 판매량 합계 : " + str(sum(yo_qty_3)))
```

```
print("드링킹 요구르트 4월 판매량 합계 : " + str(sum(yo_qty_4)))
```

```
print("드링킹 요구르트 5월 판매량 합계 : " + str(sum(yo_qty_5)))
```

```
print("드링킹 요구르트 6월 판매량 합계 : " + str(sum(yo_qty_6)))
```


10년치 데이터 추가

상품명	판매년도	판매월	판매량			
			1주	2주	3주	4주
드링킹 요구르트	2020	1	15	16	20	29
	2020	2	8	11	25	21
	2020	3	17	9	17	24
	(중략)					
	2029	10	2	22	14	22
	2029	11	16	23	24	2
	2029	12	4	3	8	16

10년치 데이터 출력을 위해 필요한 코드

- 월별로 1개의 리스트를 만들면, 10년이면 **리스트 120개**
- 리스트를 출력할 **print 명령어 120개**

연간 보고서 작성하기: 중첩된 리스트 활용

여러 개의 리스트를 모아서 하나의 새로운 리스트로 만들면?



리스트 120개를 변수 120개에 저장? 🤖

리스트 120개를 모아서 새로운 리스트로 만들고,

새로운 리스트를 변수에 저장하자!

중첩된 리스트(nested list)

연간 판매량을 중첩된 리스트로 표현하기

상품명	판매년도	판매월	판매량			
			1주	2주	3주	4주
드링킹 요구르트	2020	1	15	16	20	29
	2020	2	8	11	25	21
	2020	3	17	9	17	24
	(중략)					
	2029	10	2	22	14	22
	2029	11	16	23	24	2
	2029	12	4	3	8	16

1월 판매량

[15, 16, 20, 29]

2월 판매량

[8, 11, 25, 21]

3월 판매량

[17, 9, 17, 24] ... (중략)

연간 판매량

[[15, 16, 20, 29], [8, 11, 25, 21], [17, 9, 17, 24], ... (중략)]


```

yo_qty_2020 = [
    [15, 16, 20, 29],
    [8, 11, 25, 21],
    [17, 9, 17, 24],
    [19, 29, 29, 4],
    [5, 20, 18, 12],
    [21, 7, 27, 14],
    [19, 15, 4, 23],
    [3, 7, 15, 1],
    [14, 5, 25, 7],
    [17, 24, 2, 23],
    [0, 6, 2, 28],
    [20, 29, 22, 22],
]

```

상품명	판매년도	판매월	판매량			
			1주	2주	3주	4주
드링킹 요구르트	2020	1	15	16	20	29
		2	8	11	25	21
		3	17	9	17	24
		4	19	29	29	4
		5	5	20	18	12
		6	21	7	27	14
		7	19	15	4	23
		8	3	7	15	1
		9	14	5	25	7
		10	17	24	2	23
		11	0	6	2	28
		12	20	29	22	22

데이터에 이름을 붙여 관리하기: **딕셔너리**

리스트의 단점: 데이터에 대한 정보가 없다

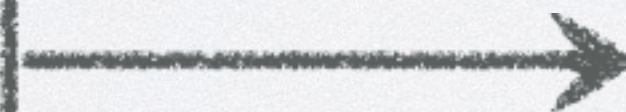
```
yo_qty = [  
    [15, 16, 20, 29],  
    [8, 11, 25, 21],  
    [17, 9, 17, 24],  
    [19, 29, 29, 4],  
    ...  
    [14, 5, 25, 7],  
    [17, 24, 2, 23],  
    [0, 6, 2, 28],  
    [20, 29, 22, 22],  
]
```

2029년 9월 판매량 데이터를
어떻게 쉽게 찾을 수 있을까?



데이터에 이름을 붙여 세트로 관리하기: **딕셔너리**

```
yo_qty = [  
    [15, 16, 20, 29],  
    [8, 11, 25, 21],  
    [17, 9, 17, 24],  
    [19, 29, 29, 4],  
    ...(중간생략)  
    [14, 5, 25, 7],  
    [17, 24, 2, 23],  
    [0, 6, 2, 28],  
    [20, 29, 22, 22],  
]
```



```
yo_qty = {  
    "2020년1월": [15, 16, 20, 29],  
    "2020년2월": [8, 11, 25, 21],  
    "2020년3월": [17, 9, 17, 24],  
    "2020년4월": [19, 29, 29, 4],  
    ...(중간생략)  
    "2029년9월": [14, 5, 25, 7],  
    "2029년10월": [17, 24, 2, 23],  
    "2029년11월": [0, 6, 2, 28],  
    "2029년12월": [20, 29, 22, 22],  
}
```


리스트와 비교하는 방법으로 **딕셔너리 만들기**

리스트를 만드는 방법

"python" "파이썬"

"python", "파이썬"

["python", "파이썬"]

데이터를 나열

컴마로 구분

괄호로 마무리

딕셔너리를 만드는 방법

"영문명": "python" "한글명": "파이썬"

"영문명": "python", "한글명": "파이썬"

{ "영문명": "python", "한글명": "파이썬" }

리스트와 딕셔너리의 상호 변환

변환할 리스트

```
[ "python", "파이썬" ]
```

대괄호를 중괄호로 변경

```
{ "python", "파이썬" }
```

데이터 이름 추가

```
{ "영문명" "python", "한글명" "파이썬" }
```

콜론(쌍점) 추가

```
{ "영문명": "python", "한글명": "파이썬" }
```


리스트와 딕셔너리 요약 정리 (공통점과 다른점)

공통점

- 여러 개의 데이터를 **하나의 데이터 세트로 관리**
- **요소** → 데이터 세트에 저장된 각각의 데이터, **여러 줄로 나눠 작성 가능**
- **마지막 요소** 오른쪽의 **컴마**는 **생략 가능**
- **모든 데이터 타입**을 요소로 저장 가능 → 예) 중첩된 리스트, 중첩된 딕셔너리

다른점

- **리스트** → 데이터 이름 **×** → 만들기 간편, 찾아 쓰기 불편
- **딕셔너리** → 데이터 이름 **○** → 만들기 불편, 찾아 쓰기 간편