

6) 중첩 루프

이전 실에서는 중첩 루프(nested loop)에 대해 알아보았습니다. 여기서 루프라는 용어는 반복을 의미하고 중첩이라는 것은 여러 개가 겹치는 것을 의미합니다. 즉, 여러번 어떤 것이 반복되는 구조를 중첩 루프라고 합니다. 보통 두 개의 반복문이 겹쳐 있는 이중 루프와 세 개의 반복문이 겹쳐 있는 삼중 루프를 가장 많이 사용합니다.

다음은 반복문 두 개가 겹쳐 있는 이중 루프의 예입니다. 반복문은 for 키워드를 사용하고 for 문 내에서 조건을 만족할 때 수행되는 문장에는 pass 키워드를 사용합니다. 참고로 파이썬의 pass 키워드는 아무것도 수행하지 않음을 의미합니다.

```
>>> for i in [1, 2, 3, 4]:
>>>     for j in [5, 6, 7, 8]:
>>>         pass
>>>
```

파이썬 반복문에는 한행일 문장이 최소한 하나라도 있어야 문법 오류가 나지 않으므로 pass를 사용해 문법 오류가 발생하는 것을 방지 할 것입니다. 물론 pass 대신 print()와 같은 구문을 넣어도 프로그램의 동작을 간단히 테스트해볼 수도 있습니다.

그렇다면 어떤 경우에 두 개의 반복문을 겹쳐서 사용하는 것일까요? 그림 4.3은 제곱 알고 알고리즘의 각 세대를 간단히 표시해 본 것입니다. 1세대는 101로, 102로, 103로, 104로가 있고 2세대는 201로, 202로, 203로, 204로가 있습니다. 지금까지 배운 반복문으로는 101로, 102로, 103로, 104로에 대해 반복적인 일을 수행할 수 없습니다. 그런데 그림 4.3은 1세대만 세대가 있는 것이 아니라 3중, 4중, 5중, 6중 세대가 있습니다. 우리가 자주 사용하는 역삼차 데이터가 그림 4.3과 같은 형태로 저장되는 데 이러한 구조를 2차원 데이터라고 표현합니다. 이러한 2차원 데이터를 다룰 때 필요한 것이 바로 이중 루프입니다.

	401	402	403	404
	301	302	303	304
	201	202	203	204
	101	102	103	104
행				
				열

그림 4.3 2차원 데이터의 표현

그림 4.3의 각 세대에 대해 순환을 자동으로 배열하는 코드를 만든다고 가정해 봅시다. 지금까지 배운 반복문을 사용하면 다음과 같이 코드를 프로그래밍해야 할 것입니다.

```
1세대 가서 3중의 각 세대의 순환 배열
2세대 가서 2중의 각 세대의 순환 배열
3세대 가서 1중의 각 세대의 순환 배열
4세대 가서 4중의 각 세대의 순환 배열
```

- 2차원 구조 = 축(변수)가 두개 존재하는 공간
- 3차원 구조 = 축(변수)가 세개 존재하는 공간