

오호! 우리 똑똑이 탐험가들, 지난 시간에 배운 '상자 속의 상자' 같은 중첩 딕셔너리, 아주 잘 이해했더구나! 🍌
이번에는 그 복잡한 보물 상자에서 원하는 정보들만 쓱쓱 골라내거나, 새로운 형태로 착착 정리하는 마법 같은
기술들을 배워볼 거야! 바로 **컴프리헨션(Comprehension)** 과 멋진 **반복 필터링** 기술이지!

(선생님은 교실 칠판에 거대한 users 딕셔너리가 그려져 있다고 가정하고 이야기를 시작한다. 이 users
딕셔너리는 앞서 보여주신 많은 사용자 정보가 담긴 그 큰 딕셔너리란다!)

<제8장. 마법의 망원경과 정리함! 컴프리헨션!>

선생님: 애들아, 만약 우리가 가진 저 거대한 users 보물 상자에서 모든 사람의 이메일 주소만 쓱 뽑아서 새로운
목록을 만들고 싶다면 어떻게 할까? 지금까지 배운 방법으로는 for 반복문으로 하나씩 꺼내서 빈 리스트에
추가해야 했지?

(복습) 예전 방식대로 모든 사용자의 이메일 주소 모으기

```
emails = []
```

```
for user_info_dict in users.values(): # users 딕셔너리의 값들(각 사용자 정보 딕셔너리)을 하나씩 꺼내서
    emails.append(user_info_dict["email"]) # 그 딕셔너리에서 "email" 값을 찾아 emails 리스트에 추가!
# print(emails) # 이렇게 하면 모든 이메일이 담긴 리스트가 완성!
```

코딩새싹 🌱: 네! for문으로 빙글빙글 돌면서 append로 하나씩 담았었죠! 조금... 손이 많이 가는 느낌? 😊

선생님: 맞아! 그래서 파이썬에는 더 짧고 멋지게, 마치 마법 주문처럼 리스트나 딕셔너리를 만들 수 있는
컴프리헨션이라는 기술이 있단다! 먼저 **리스트 컴프리헨션**부터 만나볼까?

<1. 리스트 컴프리헨션: 원하는 것만 쓱쓱! 나만의 리스트 만들기!>

선생님: 리스트 컴프리헨션의 마법 주문은 이렇게 생겼어.

```
새로운_리스트 = [ (어떤 값을 넣을지) for (항목 이름) in (반복할 수 있는 것) if (특별한 조건) ]
```

여기서 if (특별한 조건) 부분은 필요할 때만 쓴단다.

자, 아까 그 이메일 주소 모으기를 리스트 컴프리헨션으로 바꿔보면?

리스트 컴프리헨션으로 모든 사용자의 이메일 주소 수집하기

```
emails = [user_info_dict["email"] for user_info_dict in users.values()]
print("모든 사용자 이메일 (컴프리헨션):", emails)
```

궁금이 💡: 와! 아까 여러 줄이었던 코드가 딱 한 줄로 줄었어요! user_info_dict["email"] 이게 '어떤 값을

넣을지' 부분이고, `for user_info_dict in users.values()` 이게 '항목 이름'과 '반복할 수 있는 것'이네요!

선생님: 정확해! `users.values()`는 우리 `users` 딕셔너리에서 각 사용자 정보가 담긴 딕셔너리들만 쓱쓱 뽑아주는 역할을 하지. 그걸 `user_info_dict`라는 이름으로 하나씩 받아서, 그 안의 "email" 값만 골라 새 리스트를 만드는 거야!

코딩새싹 🌱: 그럼 모든 사용자의 나이만 모아서 리스트를 만들 수도 있겠네요? `[user_info_dict["age"] for user_info_dict in users.values()]` 이렇게요?

선생님: (엄지 척!) 바로 그거야! 원리를 이해하니 바로 응용하는구나!

리스트 컴프리헨션으로 모든 사용자의 나이 수집하기

```
ages = [user_info_dict["age"] for user_info_dict in users.values()]
print("모든 사용자 나이:", ages)
```

궁금이 💡: 잠깐만요, 선생님! `users.values()` 이게 정확히 어떻게 생겼는지 한번 볼 수 있을까요?

선생님: 좋은 질문이야! `users.values()`는 딕셔너리의 값들만 모아서 보여주는데, 그 값 하나하나가 바로 각 사용자의 정보가 담긴 딕셔너리들이지.

`users.values()`가 어떤 모습인지 살짝 엿보기

```
value_example = users.values()
# print("users.values()의 일부 모습:", list(value_example)[0]) # 너무 기니까 첫 번째 사용자 정보만 살짝!
```

선생님: `list(value_example)[0]` 이렇게 하면 첫 번째 사용자의 정보 딕셔너리 { "name": "John Doe", "age": 30, ... } 이런 식으로 생긴 걸 볼 수 있단다.

<2. 딕셔너리 컴프리헨션: 키-값 짝꿍! 나만의 딕셔너리 만들기!>

선생님: 리스트 컴프리헨션처럼, 딕셔너리도 한 줄 마법으로 만들 수 있어! 이것 **딕셔너리 컴프리헨션**이라고 해. 새로운_딕셔너리 = { (키가 될 것) : (값이 될 것) for (항목 이름) in (반복할 수 있는 것) if (특별한 조건) }

예를 들어, 나이가 30세 이상인 사용자들의 이름과 그들의 관심사를 짝지어서 새로운 딕셔너리를 만들어볼까?

딕셔너리 컴프리헨션: 나이가 30세 이상인 사용자의 이름과 관심사

```
over_30_interests = {
    user_info_dict["name"]: user_info_dict["interests"]
```

```

    for user_info_dict in users.values()

    if user_info_dict["age"] >= 30
}

print("30세 이상 사용자들의 관심사:", over_30_interests)

```

코딩새싹 🌱: 와! 이번엔 중괄호 {}를 썼네요! 그리고 키랑 값이 콜론 :으로 연결되어 있고요! if user_info_dict["age"] >= 30 이렇게 조건도 넣을 수 있군요!

선생님: 맞아! '이름'을 키로, '관심사 리스트'를 값으로 하고, '나이가 30세 이상인' 사람들만 쏙 골라낸 거지!

궁금이 💡: 그럼 각 사용자의 이름과 그 사용자가 가진 친구의 수를 딕셔너리로 만들 수도 있겠어요! 친구 수는 len() 함수로 리스트의 길이를 재면 되니까요!

선생님: 오오, 궁금이! 아주 훌륭한 아이디어야! 어떻게 코드를 짜면 될까?

궁금이 💡: 음... { user_info_dict["name"] : len(user_info_dict["friends"]) for user_info_dict in users.values() } 이렇게 하면 될 것 같아요!

선생님: 정답! 정말 대단한걸!

딕셔너리 컴프리헨션: 각 사용자의 친구 수

```

friend_counts = {

    user_info_dict["name"]: len(user_info_dict["friends"])

    for user_info_dict in users.values()

}

print("사용자별 친구 수:", friend_counts)

```

<제9장. 특별한 손님 찾기! - 반복문과 조건문으로 필터링!>

선생님: 컴프리헨션은 새로운 리스트나 딕셔너리를 만들 때 아주 유용해. 그런데 때로는 그냥 특정 조건을 만족하는 정보를 찾아서 바로 화면에 보여주고 싶을 때도 있겠지? 예를 들어, '스포츠'를 관심사로 가진 사용자의 이름과 나이를 찾아 바로바로 알려주고 싶다면?

코딩새싹 🌱: 앓, 그럴 땐 컴프리헨션으로 새 리스트나 딕셔너리를 꼭 만들어야 하나요?

선생님: 꼭 그럴 필요는 없어! 우리가 잘 아는 for 반복문과 if 조건문을 사용해서 정보를 찾아 바로 출력할 수도 있단다.

```
# 'sports'를 관심사로 가지고 있는 사용자의 이름과 나이 출력하기

print("\n--- 'sports'를 좋아하는 사람들 ---")

for user_info in users.values(): # 모든 사용자 정보를 하나씩 살펴본다

    # user_info는 각 사용자의 정보가 담긴 딕셔너리

    # user_info['interests']는 그 사용자의 관심사 리스트

    if 'sports' in user_info['interests']: # 만약 'sports'가 관심사 리스트 안에 있다면

        print(f"이름: {user_info['name']}, 나이: {user_info['age']}")
```

궁금이 💡: 아하! users.values()로 각 사용자 정보 딕셔너리를 user_info로 받고, user_info['interests']가 관심사 리스트니까, 그 리스트 안에 'sports'가 있는지 in으로 확인하는 거네요!

선생님: 정확해! 이렇게 하면 'sports'를 좋아하는 사람들을 찾을 때마다 바로바로 화면에 보여줄 수 있지. 물론 이걸 컴프리헨션으로 만들어서 리스트에 담을 수도 있지만, 이렇게 바로 처리하는 것도 아주 유용한 방법이란다.

코딩새싹 🌱: 와, 선생님! 컴프리헨션이라는 마법 주문이랑, 원래 알던 for문이랑 if문을 같이 쓰니까 정말 못할 게 없을 것 같아요! 데이터가 아무리 많아도 다 찾아낼 수 있겠어요!

선생님: (호뭇하게 웃으며) 맞아! 오늘 배운 컴프리헨션과 필터링 기술은 여러분이 파이썬으로 데이터를 다룰 때 아주 강력한 무기가 될 거야. 복잡한 정보 속에서 원하는 것을 효율적으로 찾아내고 정리하는 능력, 정말 멋지지 않니? 오늘 탐험도 정말 훌륭하게 마쳤구나, 우리 용감한 데이터 탐험가들!