

파이썬의 결정 마법사: 불리언, 비교 연산자, 그리고 if 조건문! 🚦✅?🐢

등장인물:

- **슬기 선생님:** 친절하고 유머러스한 코딩 선생님
- **지혜:** 똑똑하고 질문이 많은 학생
- **코딩이:** 코딩을 처음 배우지만 호기심 많은 학생

슬기 선생님: 여러분, 안녕하세요! 우리가 지금까지 파이썬에게 여러 가지 명령을 내리는 법을 배웠죠? 오늘은 파이썬이 스스로 **상황을 판단하고 결정을** 내릴 수 있게 하는 아주 중요한 마법 도구들을 배워볼 거예요! 바로 “예” 또는 “아니오”로 답하는 **불리언(Boolean)**, 두 값을 비교하는 **비교 연산자**, 그리고 조건에 따라 다른 행동을 하게 하는 **if 조건문**입니다!

코딩이: 스스로 결정을 내려요? 파이썬이 생각도 할 수 있는 거예요? “이 사탕 먹을까, 말까?” 이런 고민도 하나요?

슬기 선생님: (웃음) 하하, 파이썬이 사탕 고민을 하진 않지만, 우리가 정해진 조건에 따라서 “이게 맞나, 틀리나?”를 판단하고, 그 판단 결과에 따라 다른 명령을 수행할 수 있어요. 마치 우리가 “신호등이 초록불이면 길을 건너고, 빨간불이면 멈춘다”처럼 행동하는 것과 비슷하죠!

1. 불리언 (Boolean): 딱 두 가지만 아는 똑똑이!

슬기 선생님: 가장 먼저 만나볼 친구는 ****불리언(Boolean)****이라는 아주 특별한 데이터 형태예요. 이 친구는 딱 두 가지 값만 가지고 있습니다. 바로 **True (참, 맞다!)** 와 **False (거짓, 틀리다!)** 예요. 대문자로 시작하는 것 꼭 기억해주세요!

지혜: True랑 False요? 네, 아니오 같은 건가요? 왜 이름이 불리언이예요?

슬기 선생님: 맞아요! “예” 또는 “아니오”, “켜짐” 또는 “꺼짐”처럼 세상의 많은 상황을 딱 두 가지로 표현할 수 있을 때 사용해요. ‘불리언’이라는 이름은 이 개념을 처음 생각한 영국의 수학자 겸 논리학자 조지 불(George Boole) 아저씨의 이름에서 따왔답니다. `type(True)` 라고 쳐보면 `<class 'bool'>` 이라고 나올 거예요.

코딩이: 아하! 조지 불 아저씨가 만든 True, False 똑똑이!

2. 비교 연산자: 누가누가 더 큰가, 같은가 비교하기!

슬기 선생님: 그럼 파이썬은 언제 True나 False라는 값을 사용할까요? 바로 **비교 연산자**를 사용할 때랍니다! 비교 연산자는 두 개의 값을 비교해서 그 결과가 맞으면 True, 틀리면 False를 우리에게 알려줘요.

- `a == b`: a랑 b랑 똑같으면 True, 다르면 False (등호가 두 개!! = 하나는 값을 넣을 때 썼죠?)
- `a != b`: a랑 b랑 다르면 True, 똑같으면 False (“!”는 “아니다”라는 느낌!)
- `a > b`: a가 b보다 크면 True
- `a < b`: a가 b보다 작으면 True
- `a >= b`: a가 b보다 크거나 같으면 True
- `a <= b`: a가 b보다 작거나 같으면 True

자, 예를 들어볼까요? `a = 10` 이고 `b = 3` 이라고 해봅시다.

```
a = 10
b = 3
```

```
print("a == b:", a == b) # a랑 b랑 같나요?
print("a != b:", a != b) # a랑 b랑 다른가요?
print("a > b:", a > b)   # a가 b보다 큰가요?
print("a < b:", a < b)   # a가 b보다 작은가요?
print("a >= 10:", a >= 10) # a가 10보다 크거나 같나요?
print("b <= 3:", b <= 3)  # b가 3보다 작거나 같나요?
print("a <= b:", a <= b)  # a가 b보다 작거나 같나요? (사용자 요청 예시)
```

지혜: 와! `a == b`는 False가 나오고, `a != b`는 True가 나왔어요! `a > b`도 True고요! 정말 비교한 결과가 맞는지 틀린지 알려주세요!

3. if 조건문: “만약 ~하면, ~해라!”

슬리 선생님: 이제 이 True, False를 이용해서 파이썬에게 조건을 걸어 명령을 내릴 수 있게 해주는 마법 주문, if를 배워볼 거예요! if는 “만약 ~이라면”이라는 뜻이죠?

```
# if 조건식:
#   조건식이 True일 때만 실행할 명령들 (들여쓰기 필수!)
```

```
a = 2
if a > 5: # 만약 a가 5보다 크다면... (a는 2니까 이 조건은 False죠?)
    print("a는 5보다 크다.") # 그래서 이 줄은 실행되지 않아요!
```

```
print("여기는 if 조건문 바깥입니다.") # 이 줄은 if랑 상관없이 항상 실행돼요!
```

코딩이: 어? a가 2니까 `a > 5`는 틀린 말(False)이라서 `print("a는 5보다 크다.")`는 화면에 안 나왔어요! 대신 “여기는 if 조건문 바깥입니다.”만 나왔네요! if 다음에 오는 조건이 True일 때만 그 아래 들여쓰기 된 명령을 실행하는 거군요!

슬기 선생님: 정확해요! if문은 콜론(:)으로 끝나고, 그 조건이 True일 때 실행될 명령들은 반드시 들여쓰기를 해서 “이만큼이 if 조건이 맞을 때 할 일이야!”라고 묶어줘야 해요. 만약 조건이 False면 들여쓰기 된 부분은 그냥 통째로 건너뛰답니다.

4. if문 활용: 짝수 홀수 판별기!

```
number = int(input("원하는 수를 입력하세요: ")) # 숫자를 입력받아 정수로 변환!
```

```
if number % 2 == 0: # 만약 숫자를 2로 나눈 나머지가 0이면 (짝수 조건!)
    print("이 수는 짝수입니다.")
```

```
if number % 2 != 0: # 만약 숫자를 2로 나눈 나머지가 0이 아니면 (홀수 조건!)
```

```
print("이 수는 홀수입니다.")
```

지혜: number % 2는 숫자를 2로 나눈 나머지를 알려주는 거죠? 그래서 나머지가 0이면 짝수, 0이 아니면 홀수라고 판단해서 맞는 메시지를 보여주네요! 제가 10을 넣으니까 “이 수는 짝수입니다.”라고 나왔고, 7을 넣으니까 “이 수는 홀수입니다.”라고 나왔어요!

슬기 선생님: 맞아요! 이 예제에서는 if문 두 개를 사용해서 각각 짝수일 때와 홀수일 때를 검사했죠? 그런데 사실 “짝수가 아니면 당연히 홀수”잖아요? 이럴 때 더 편리하게 쓸 수 있는 마법이 있습니다!

5. else 마법: “그렇지 않으면, 이것 해!”

슬리 선생님: else는 if문의 조건이 False일 때, 즉 “그렇지 않으면” 실행할 명령을 지정해 줄 수 있어요.

```
age = int(input("나이를 입력하세요.: "))
```

```
if age >= 18: # 만약 나이가 18세 이상이라면...
```

```
    print('운전면허증 시험 등록 가능')
```

```
else: # 그렇지 않다면 (즉, 18세 미만이라면...)
```

```
    print('운전면허증 등록 불가')
```

코딩이: 와! else를 쓰니까 훨씬 간단해졌어요! age >= 18 조건이 True면 “등록 가능”이 나오고, 그 조건이 False면 else 아래 있는 “등록 불가”가 나오는 거네요! 아까 짝수 홀수 판별기도 else로 바꿀 수 있겠어요!

```
# 짝수 홀수 판별기 (if-else 버전)
```

```
# number = int(input("원하는 수를 입력하세요.: "))
```

```
# if number % 2 == 0:
```

```
#     print("이 수는 짝수입니다.")
```

```
# else: # 짝수가 아니라면 (즉, 홀수라면)
```

```
#     print("이 수는 홀수입니다.")
```

슬기 선생님: 바로 그거예요! if와 else를 함께 사용하면, 조건에 따라 둘 중 하나의 명령을 선택해서 실행하도록 만들 수 있습니다. 마치 갈림길에서 왼쪽으로 갈지, 오른쪽으로 갈지 정하는 것과 같죠.

오늘 우리는 파이썬이 “맞다/틀리다”를 판단하는 불리언, 값을 비교하는 비교 연산자, 그리고 그 판단에 따라 다른 행동을 하게 하는 if와 else 조건문까지 배웠어요. 이 도구들을 잘 활용하면 훨씬 더 똑똑하고 유연한 프로그램을 만들 수 있을 거예요!

지혜 & 코딩이: 네! 선생님! 이제 파이썬이 상황에 맞게 다르게 행동하도록 만들 수 있겠어요! 정말 신기하고 재미있어요!