✅

Python Study Week4 Solve

문제

❓ **후동이가 너무 심심한 나머지 친구와 Up-Down Game을 하려고 한다.**
하지만 1개로 하면 너무 심심하기 때문에 2개의 숫자로 동시에 진행을 하기로 했다🤔

진행 방식 및 조건

1. `while` **반복문을 사용합니다.**
2. `random` **으로 숫자 2개를 배정합니다.**
3. **각 숫자를 맞췄는지에 따라 경우가 4가지로 나뉩니다.**
   a. **둘 다 맞추지 못한 경우**
      - `"아직 하나의 숫자도 맞추지 못하셨습니다.."` **안내문 출력**
      - *** 첫번째 숫자, 두번째 숫자 둘 다 게임 진행**
   b. **첫번째 숫자만 맞춘 경우**
      - `"첫번째 숫자는 이미 맞추셨습니다! 두번째 숫자만 맞춰주시면 됩니다."` **안내문 출력**
      - *** 두번째 숫자만 게임 진행**
   c. **두번째 숫자만 맞춘 경우**
      - `"두번째 숫자는 이미 맞추셨습니다! 첫번째 숫자만 맞춰주시면 됩니다."` **안내문 출력**
      - *** 첫번째 숫자만 게임 진행**
   d. **모두 맞춘 경우**
      - `축하드립니다! N번 만에 정답을 맞추셨습니다!` **안내문 & 시도한 횟수 출력**
      - `게임을 종료합니다 :)` **안내문 출력 & 게임 종료**

> 👨‍💻 **이번에는 1번의 게임만 진행하는 걸 조건으로 하겠습니다!**
> **가능하다면 게임을 계속할지 중단할지 받는 것도 추가해보세요🙂**

4. **사용자로부터 답을 입력받습니다.**
   - `"첫번째 숫자를 맞춰보세요: "` or `"두번째 숫자를 맞춰보세요: "`
   - **정답보다 입력한 숫자가 크면** `Down!` **을, 작으면** `Up!` **을 출력합니다.**
   - **정답이면** `"첫번째 숫자를 맞추셨습니다!"` **or** `"두번째 숫자를 맞추셨습니다!"` **를 출력합니다.**

학습 목표

1. `random` **모듈**
2. `while` **반복문**
3. `input` **입출력문**
4. `if` **조건문**
5. `break, continue` **문**

👨‍💻 **이번 문제에서는 상태에 따라 경우가 나뉘니, if문의 조건을 어떻게 써줘야 좋을지 고민해보면 좋을 것 같아요 ❗**

출력 예시

```
아직 하나의 숫자도 맞추지 못하셨습니다..
첫번째 숫자를 맞춰보세요: 50
Down!
두번째 숫자를 맞춰보세요: 50
Down!
아직 하나의 숫자도 맞추지 못하셨습니다..
첫번째 숫자를 맞춰보세요: 25
Up!
두번째 숫자를 맞춰보세요: 25
Down!
아직 하나의 숫자도 맞추지 못하셨습니다..
첫번째 숫자를 맞춰보세요: 35
첫번째 숫자를 맞추셨습니다!
두번째 숫자를 맞춰보세요: 15
Down!
첫번째 숫자는 이미 맞추셨습니다! 두번째 숫자만 맞춰주시면 됩니다.
두번째 숫자를 맞춰보세요: 6
Up!
첫번째 숫자는 이미 맞추셨습니다! 두번째 숫자만 맞춰주시면 됩니다.
두번째 숫자를 맞춰보세요: 10
Up!
첫번째 숫자는 이미 맞추셨습니다! 두번째 숫자만 맞춰주시면 됩니다.
두번째 숫자를 맞춰보세요: 12
Down!
첫번째 숫자는 이미 맞추셨습니다! 두번째 숫자만 맞춰주시면 됩니다.
두번째 숫자를 맞춰보세요: 11
두번째 숫자를 맞추셨습니다!
축하드립니다! 7번 만에 정답을 맞추셨습니다!
게임을 종료합니다 :)
```

해설

```
import random                 # 랜덤 모듈 import

num1 = random.randint(1,100)  # Up-Down Game을 진행할 숫자1
num2 = random.randint(1,100)  # Up-Down Game을 진행할 숫자2
temp1 = False                 # 숫자1을 맞췄는지의 유무 확인
temp2 = False                 # 숫자2를 맞췄는지의 유무 확인
cnt = 0                       # 시도한 횟수

# <Up-Down Game 실행>
while True:
    # <1. 숫자를 둘 다 못 맞춘 상태>
    if not temp1 and not temp2:
        print('아직 하나의 숫자도 맞추지 못하셨습니다..')  # 안내문 출력
        cnt += 1                                          # 횟수 +1
        # <1-1. 첫번째 숫자 맞추기>
        answer1 = int(input('첫번째 숫자를 맞춰보세요: '))
        if num1 > answer1:
            print('Up!')
        elif num1 < answer1:
            print('Down!')
        else:                                             # 정답을 맞출 경우
            print('첫번째 숫자를 맞추셨습니다!')
            temp1 = True                                  # 첫번째 숫자를 맞춘 상태로 변경

        # <1-2. 두번째 숫자 맞추기>
        answer2 = int(input('두번째 숫자를 맞춰보세요: '))
        if num2 > answer2:
            print('Up!')
        elif num2 < answer2:
            print('Down!')
        else:                                             # 정답을 맞출 경우
            print('두번째 숫자를 맞추셨습니다!')
            temp2 = True                                  # 두번째 숫자를 맞춘 상태로 변경

    # <2. 첫번째 숫자만 맞춘 상태>
    elif temp1 and not temp2:
        print('첫번째 숫자는 이미 맞추셨습니다! 두번째 숫자만 맞춰주시면 됩니다.')  # 안내문 출력
        cnt += 1                                          # 횟수 +1
        # <2-1. 두번째 숫자 맞추기>
        answer2 = int(input('두번째 숫자를 맞춰보세요: '))
```

```
        if num2 > answer2:
            print('Up!')
        elif num2 < answer2:
            print('Down!')
        else:                                           # 정답을 맞출 경우
            print('두번째 숫자를 맞추셨습니다!')
            temp2 = True                                # 두번째 숫자를 맞춘 상태로 변경

    # <3. 두번째 숫자만 맞춘 상태>
    elif not temp1 and temp2:
        print('두번째 숫자는 이미 맞추셨습니다! 첫번째 숫자만 맞춰주시면 됩니다.')  # 안내문 출력
        cnt += 1                                        # 횟수 +1
        # <3-1. 첫번째 숫자 맞추기>
        answer1 = int(input('첫번째 숫자를 맞춰보세요: '))
        if num1 > answer1:
            print('Up!')
        elif num1 < answer1:
            print('Down!')
        else:                                           # 정답을 맞출 경우
            print('첫번째 숫자를 맞추셨습니다!')
            temp1 = True                                # 첫번째 숫자를 맞춘 상태로 변경

    # <4. 숫자를 모두 맞춘 상태>
    if temp1 and temp2:
        print(f'축하드립니다! {cnt}번 만에 정답을 맞추셨습니다!')
        print('게임을 종료합니다 :)')
        break
```

1. `temp1 = False` `temp2 = False` **를 설정해주어, 각 문제별 상태를 파악합니다.**
    - 둘 다 맞추지 못했을 경우에는, `not temp1 and not temp2` 이 `True` 이므로 첫번째 경우가 실행됩니다.
    - 첫번째 숫자만 맞춰 `temp1` 이 `True`가 된 상태에서는 `temp1 and not temp2` 가 `True` 이므로 두번째 경우가 실행됩니다.
    - 두번째 숫자만 맞춰 `temp2` 이 `True`가 된 상태에서는 `not temp1 and temp2` 가 `True` 이므로 세번째 경우가 실행됩니다.
    - 만약 두 숫자를 모두 맞춰 `temp1` 과 `temp2` 가 모두 `True` 가 되면 `temp1 and temp2` 가 `True` 이므로 횟수를 출력하고 게임이 종료됩니다.
2. **1,2,3번 경우가 시작될 때마다** `cnt += 1` **을 통해 횟수를 추가해줍니다.**

상태를 True or False로 나누어 조건문을 작성하는 방법은, 알아두시면 후에도 유용하게 쓰이니 이번에 잘 숙달하시기를 추천드립니다💡