6주차(1)_VS Code Debugging

Debugging

Debug는 "벌레를 잡다"라는 뜻으로, 프로그램에서 발생하는 오류를 찾아내어 수정하는 작업을 말합니다.

코드 중 **정확히 어디에서 문제가 발생**하였는지 확인하기 위해 범위를 좁혀나가면서 버그를 찾고, 이를 고쳐야 합니다.

> 💡 **Bug** = 프로그램이 오작동하는 것
> **Debugging** = 오작되는 현상들을 해결하는 것 → 오류들을 찾아내기 위한 테스트 과정

생각한 그대로를 완벽하게 코딩하는 것은 어렵습니다. 더욱이 프로젝트의 난이도가 높아질 수록 어렵고 복잡한 코드를 작성하기 때문에, 디버깅은 프로젝트의 난이도가 높아질 수록 필수적입니다.

디버깅에서 가장 중요한 것은 어떤 문제가 발생한 것이고, 어떤 버그인지, 어떤 에러가 발생한 것인지 등 **문제 상황을 정확하게 정의**하는 것입니다. 문제를 정의해야 적절한 해결방법을 찾을 수 있기 때문입니다.

디버깅은 오류나 버그를 해결할 때에만 사용하지는 않습니다. 원하는 Logit, Flow를 구현할 때, 심지어는 UX/UI를 구현할 때에도 사용하기도 합니다. 디버깅 이외에도 log 파일을 분석하는 등 다양한 방식으로 프로파일링이 가능합니다.

Debugger 사용법

많은 tools이 있지만, 현재 가장 많이 사용되는 코드 편집기인 **Visual Studio**를 사용하여 이번 실습을 진행할 예정입니다. 편의상 python으로 실습을 진행하겠습니다.

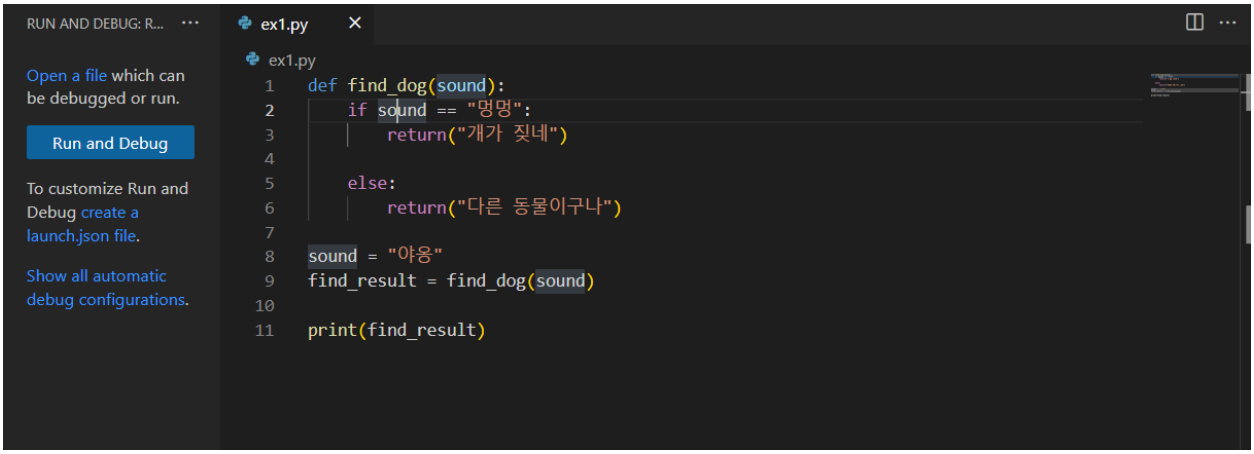
"멍멍"을 입력받으면 "개가 짖네"라는 문장을 출력하고, 그 외의 경우는 "다른 동물이구나"라는 문장을 출력하는 코드입니다.

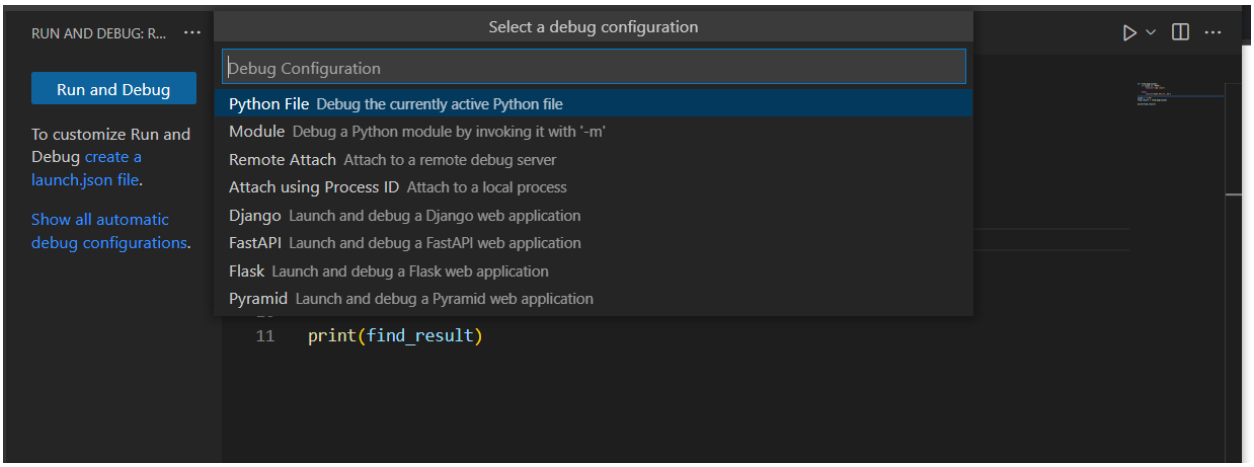
ex0.py

```
def find_dog(sound):
    if sound == "멍멍":
        return("개가 짖네")

    else:
        return("다른 동물이구나")

sound = "야옹"
find_result = find_dog(sound)

print(find_result)
```

VSCode를 실행하고, 아래처럼 ex1.py에 상단의 코드를 입력해주세요.

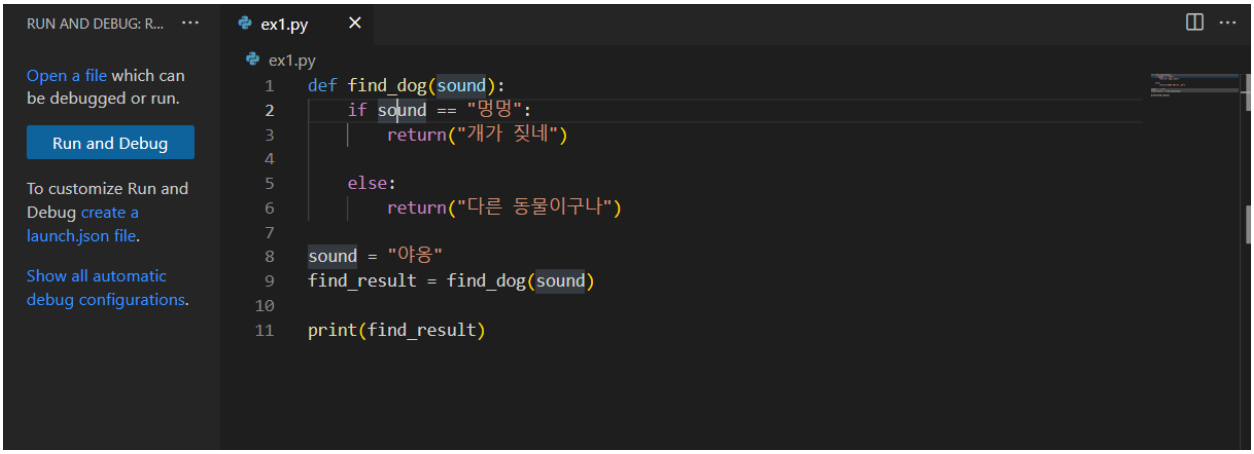
이후, Run Debug를 클릭하여 Python File을 선택하여 실행해주세요.



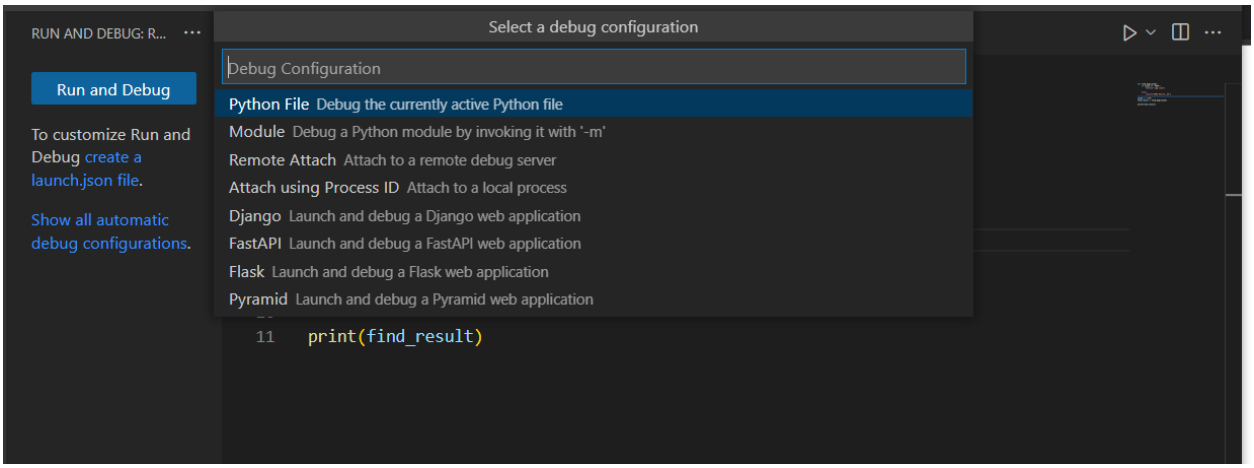


만약 위 과정처럼 Python File을 일일히 선택하고 싶지 않다면?

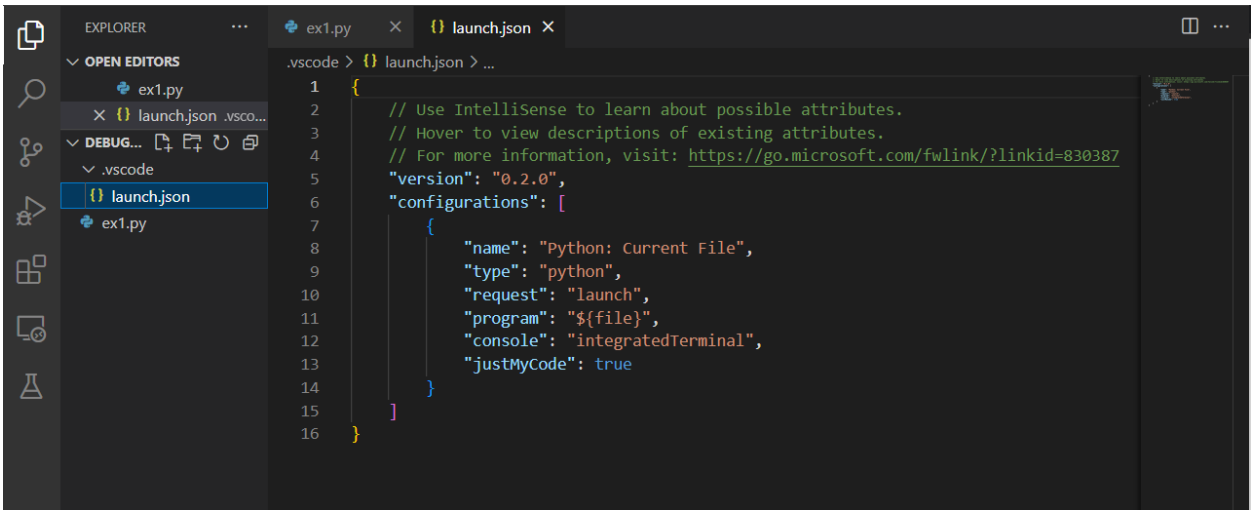
왼쪽의 `Run and Debug(실행 및 디버그)` 아래에 `create a launch.json(lanch.json 파일 만들기)` 을 클릭해주세요.



앞선 과정과 동일하게 Python File을 선택해주세요.



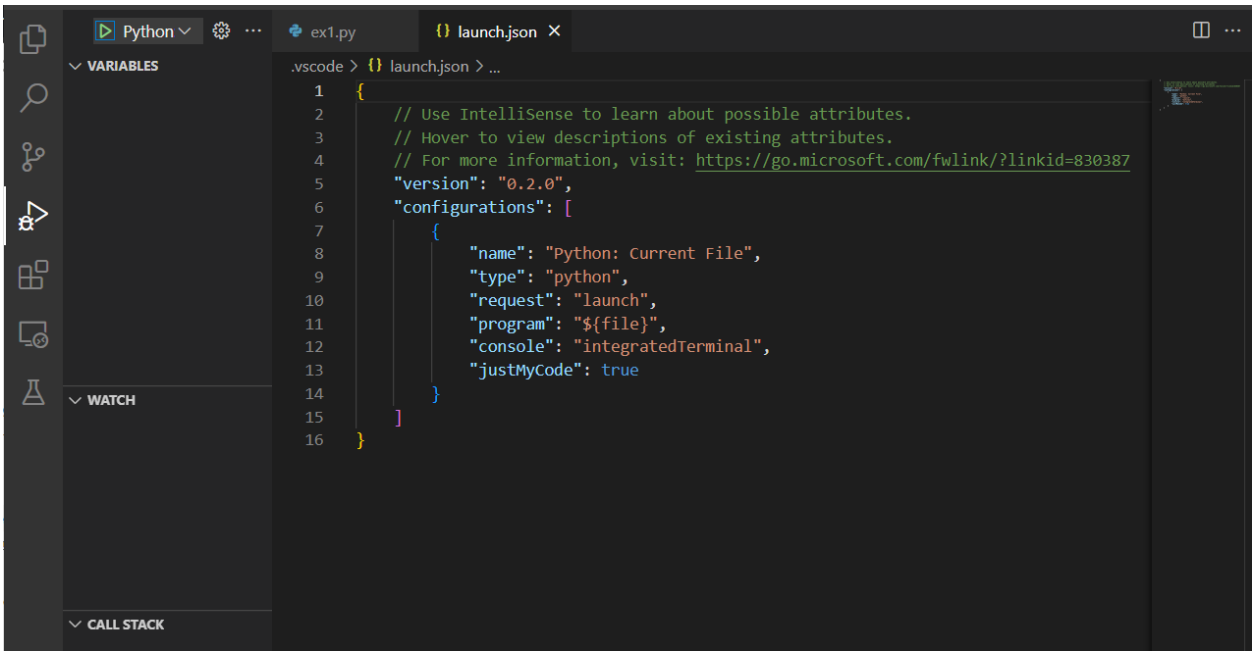
`lanch.json`파일이 생성되면 세팅이 완료된 것입니다. 해당 json파일은 닫아주셔도 됩니다.



단, 해당 설정은 각 프로젝트 폴더에서만 유효합니다. 다른 프로젝트 폴더를 생성한다면 새로 설정하셔야 합니다.

설정이 완료되면, 아까의 `run and Debug(실행 및 디버그)` 버튼이 사라지고, 왼쪽 상단에 `RUN Python:Current File` 버튼이 생성된 것을 확인할 수 있습니다.

또한 왼쪽에 `variables(변수), watch(조사식), call stack(호출 스택)` 창 3개가 추가되었습니다.

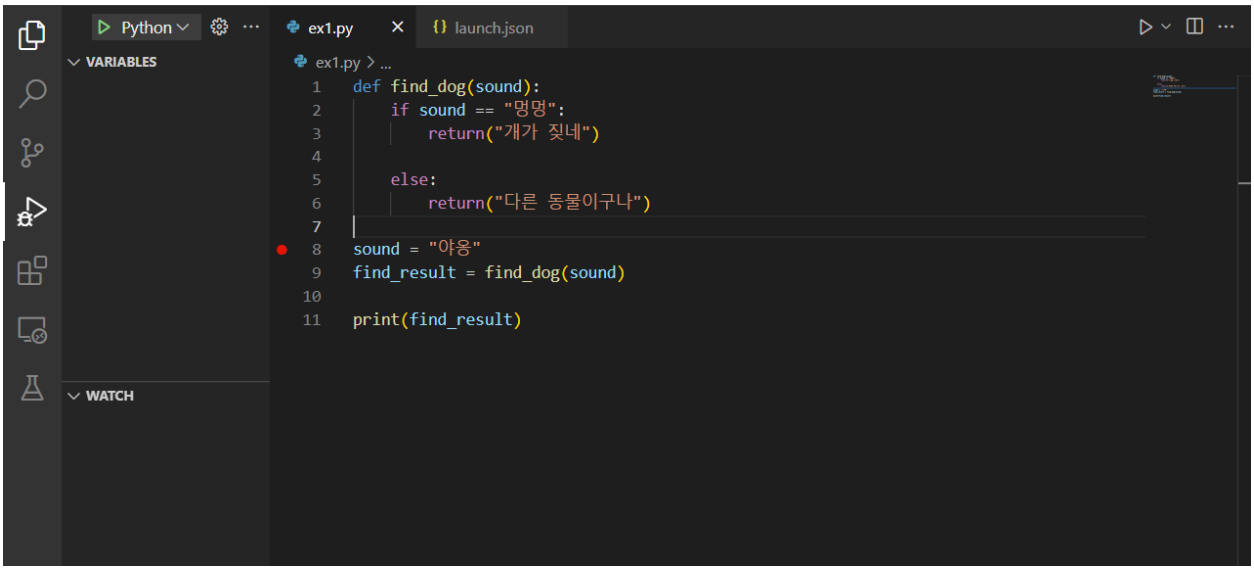


VS Code 디버깅

1. Break Point(중단점) 지정

코드에 왼쪽의 라인 넘버 쪽에 커서를 대면 빨간색 원형의 Break Point가 나타납니다. 클릭하면 Point가 찍히게 됩니다.

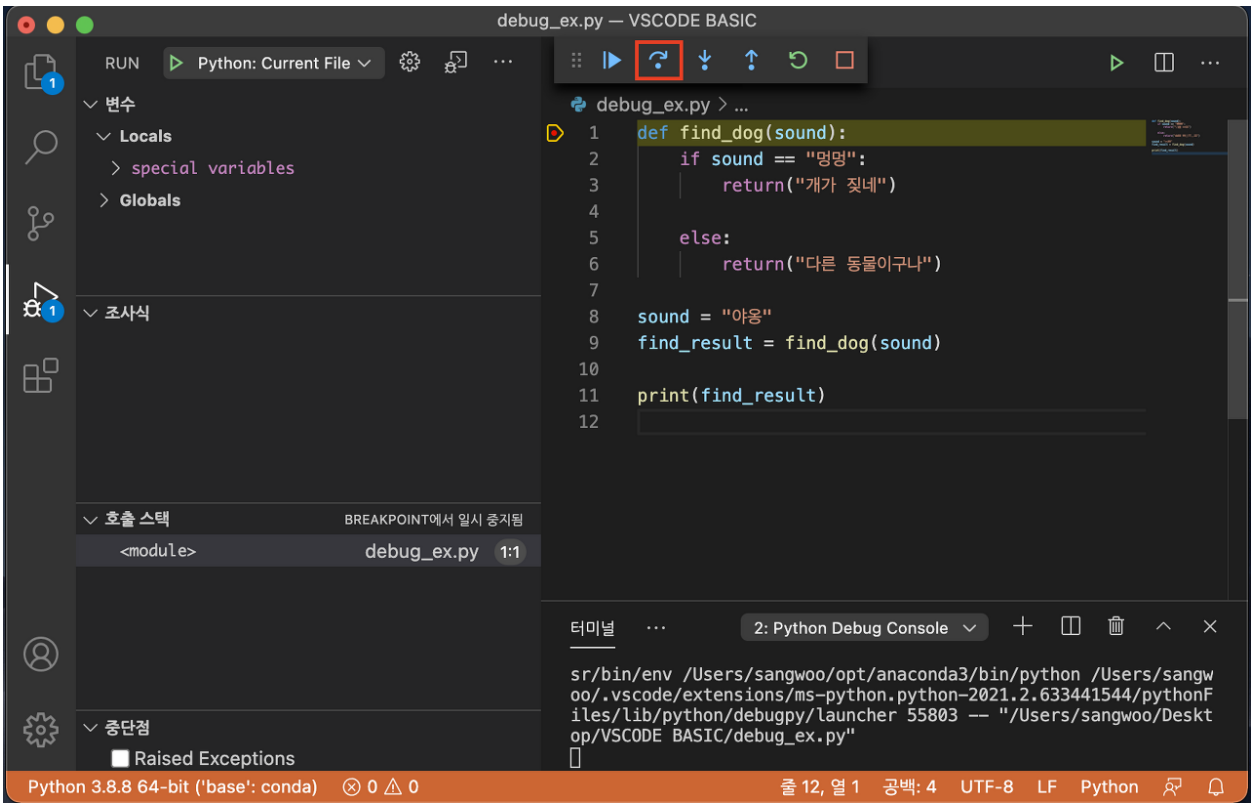
디버깅을 할 때, Break Point를 찍게 되면 프로그램이 실행되다가 해당 Point의 라인 전까지만 실행되고 일시정지됩니다.



2. 디버깅 실행(Run 버튼 클릭)

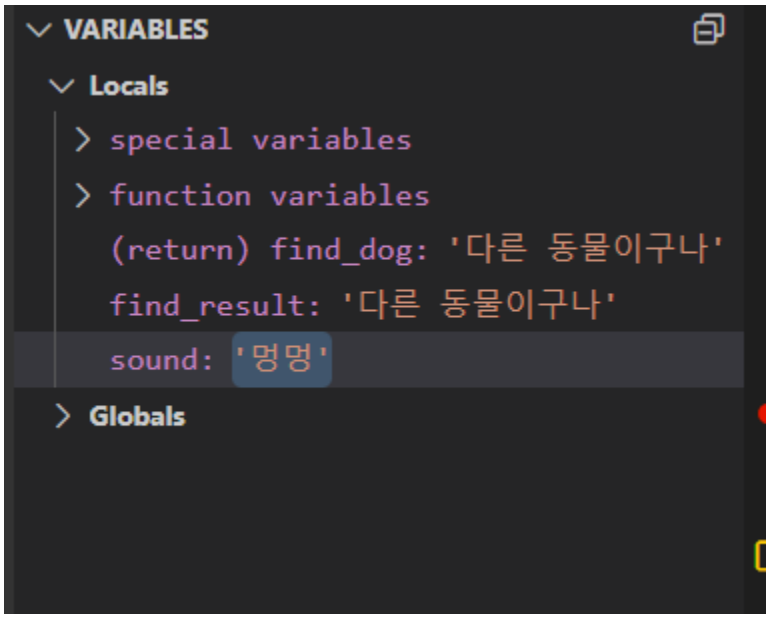
왼쪽 상단의 `RUN Python:Current File`를 클릭하면, 아직 한줄도 실행되지 않은 상태가 됩니다.

상단의 Step Over 버튼을 눌러가며 코드를 한 줄씩 실행시키며 왼쪽창의 변화를 봐주세요.

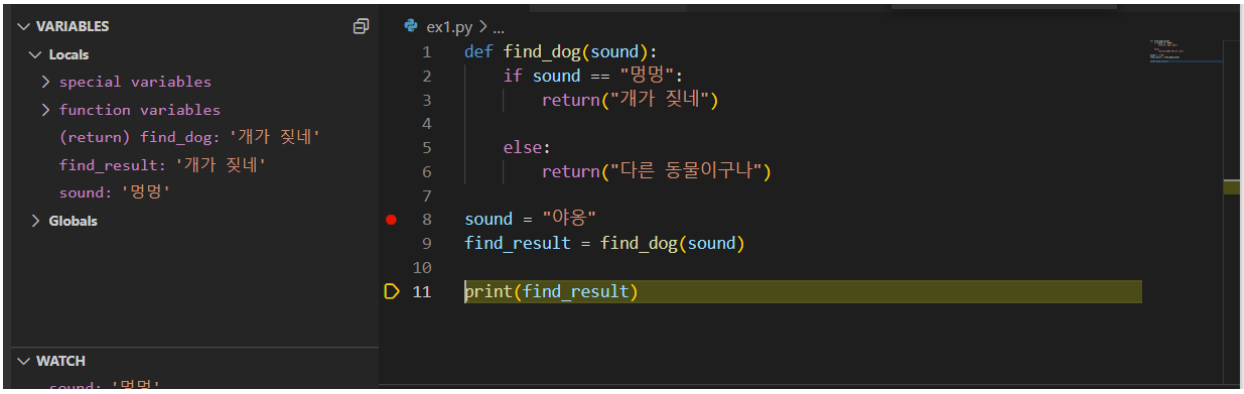


2. 변수(Variables) 사용법

이때, Variable을 더블클릭하면 변수명이나 함수명을 바꿀 수 있습니다.

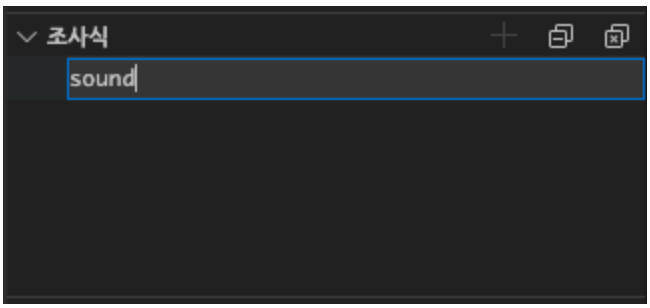


8번째 줄을 실행시킨 다음 variable창에서 해당 값을 '멍멍'으로 바꾸고, step over를 눌러서 9번째 줄을 실행시키면 이번에는 아래와 같은 결과가 나오는 것을 볼 수 있습니다.

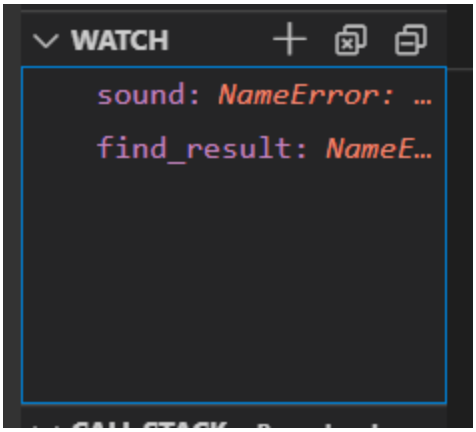


3. 조사식(Watch) 사용법

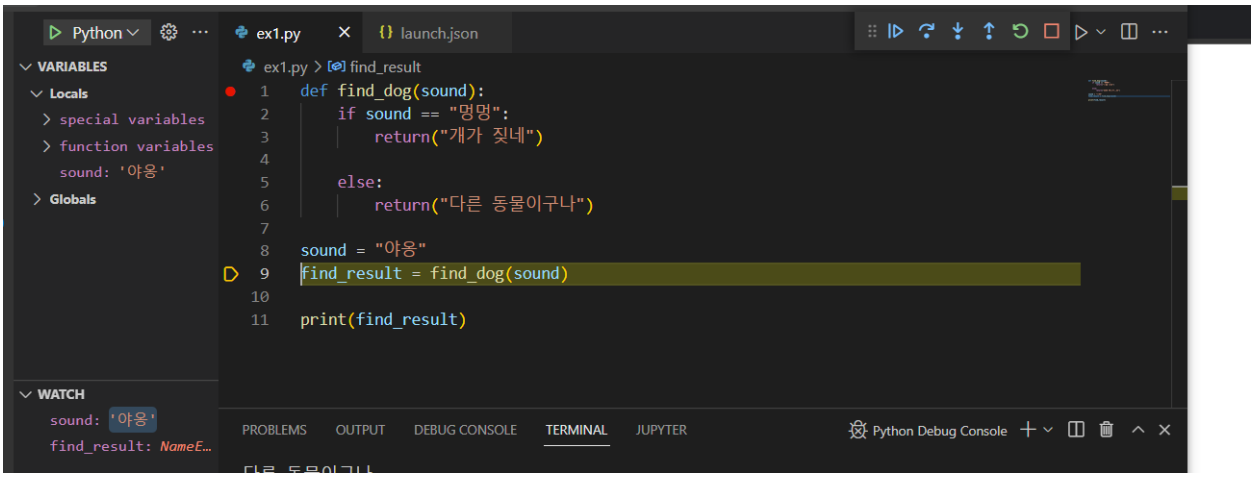
Watch(조사식) 창에 커서를 올리면 + 아이콘이 나타납니다. 클릭하여 확인하고 싶은 변수명을 입력합니다.



sound, find_result 값을 입력하면, 현재는 변수들이 정의되지 않은 상태이기 때문에 NameError가 발생합니다.



첫줄에 break point를 달고 `step over`를 클릭하면 아래와 같이 값을 받는 시점에 sound부분이 “야옹”으로 변하는 것을 확인할 수 있습니다.



한번 더 step over를 눌렀더니 find_result 값에 `다른 동물이구나`가 추가되는 것을 확인할 수 있습니다.