

1. 목표

프로그램을 개발할 때 오류가 없는지 확인하기 위해서 지금까지는 프로그램을 실행시켜서 확인해봤으나, 실행하기 전에 미리 오류가 없는지 확인하고 혹시나 실행이 됐다하더라도 오류가 있을 수 있기 때문에 프로그래밍 설계 과정에서 미리 정상 동작하는지 확인하기 위해 실시하는 unittest에 대해 학습한다.

2. 목표를 위해 알아야하는 것

unittest를 위해서는 프로그램 코드와 별개로 프로그램 테스트 코드가 있어야 한다. 테스트가 모두 통과해야만 에러 표시가 뜨지 않는다. 또한, 함수별로 각각의 테스트 코드가 있어야 한다.

3. 해결방법

(1) 테스트 통과

```
def hello_en():  
    return 'Hello, World!'  
  
def hello_kr():  
    return '안녕하세요!'
```

위와 같이 테스트 코드에 들어갔던 문자열을 그대로 사용했더니 테스트를 정상적으로 통과하여 결과가 알맞게 나왔다.

(2) 추가 언어

```
def test_greeting_jp(self):  
    self.assertEqual('Gonichiwa!',  
                      greeting.hello_jp())  
  
def hello_jp():  
    return 'Gonichiwa!'
```

위와 같이 테스트 코드에 새로운 언어인 일본어를 추가하고 “Gonichiwa!”를 테스트 문구로 선정하였다.

그 다음으로는 (1)과 같이 실행 코드에 “Gonichiwa!”를 추가하여 테스트 코드를 정상적으로 통과할 수 있게 하였다.

4. 결과

```
test_greeting_en (__main__.GreetingTest) ... ok  
test_greeting_jp (__main__.GreetingTest) ... ok  
test_greeting_kr (__main__.GreetingTest) ... ok  
  
-----  
Ran 3 tests in 0.001s
```

영어, 일본어, 한국어가 모두 테스트 코드를 통과하여 ok 판정을 받았다.

5. 느낀 점

자바에 익숙해져있다 파이썬을 사용해서 어색하긴 하지만 한 학기동안 열심히 노력해서 학기가 끝났을 때는 파이썬을 익숙하게 사용할 수 있으면 좋겠다.