



# PushPull



언어/Python3

## Chapter 03. 프로그램 사용자로부터의 입력 그리고 코드의 반복

mineeeee.\_. | 2021. 1. 5. 16:58 | 수정 | 삭제

앞에선 인자에 직접 숫자나 문자를 넣어서 결과값을 얻었는데요!

이번엔 입력 값을 받아보는 법을 배워보도록 하겠습니다!

전에 글에서 main을 배웠으니 이제부터의 코드작성은 다 main함수를 써서 할게요!

<https://mineeeee.tistory.com/50?category=905765>



### Chapter 02. 간단한 함수 만들기

mineeeee.tistory.com

```
def Q1():  
    a=input("how old are you? ")  
    print("age:",a)  
  
def main():  
    Q1()  
main()
```

이렇게 input을 이용해주면 입력값을 저기 a에 저장해 줄 수 있습니다!

실행을 시켰을 때 input받기전에 뜨는 문구가 뜨기 싫다! 나는 변수에 저장만 할 거다! 라고 하신다면 그냥

**a = input()**

이렇게 하시면 됩니다~

여기서 저는 제 나이가 23이라 23을 입력했는데요~



input은 받은 값을 모두 문자열의 형태로 처리합니다!

그래서 a=23이 아닌 a="23"으로 저장이 됩니다!

그래서 만약에 input을 통해 12와 23을 입력받고 그 둘을 더한다 하면

12+23=35가 아닌 "12"+"34"="1234"가 나오게 됩니다!

그렇다면 수로 입력 받고 싶을땐 어떻게 해야될 지 알아보겠습니다!

바로 **eval**을 쓰는건데요!

eval함수를 쓰면

```
def Q1():
    a=eval(input("how old are you? "))
    a=a+1
    print("age:",a)

def main():
    Q1()
main()
```

이렇게 하면 23을 입력하면 24가 출력됩니다!

하지만 eval함수는 보안쪽이 조금 취약하여,,,

예를 들어보겠습니다!

```
def ret():
    return 12

def Q2():
    a=eval(input("입력해주세요! "))
    print(a)

def main():
    Q2()
main()
```

이렇게 코드를 짜봤는데요!

만약에 저기 input에 ret()을 넣어주면

```
PS C:\Users\mineee\Documents> python3 03_01.py
입력해주세요! ret()
12
PS C:\Users\mineee\Documents>
```

이런 모습을 볼 수 있습니다!



이 이유는 eval가 문자에 저장된 내용을 산술 연산이 가능한 '수'로 바꿔주는 역할을 했는데요!

eval이 evaluate의 앞 부분을 딴 것인데, 그 이름처럼 전달된 문자열이 내용을 평가 및 해석해서 무엇을 할지 결정해줍니다!

그래서 ret()을 입력했을 때 eval함수에는 문자열 "ret()"이 전달되었고, eval 함수는 이 내용을 분석해서 프로그래머가 ret 함수 호출을 명령하지 않았음에도 불구하고 ret함수를 호출한 것입니다!

그래서 이는 보안의 취약점으로 여겨지는 부분 중 하나입니다!

그래서 eval 함수의 호출은 가급적 제한해야 됩니다!

이번엔 반복문을 구현해 볼 차례인데요!

먼저 **for 문**에 대해 공부해봅시다!

```
for i in [0,1,2]:  
    print(i)
```

간단하죠??

위에서 i는 변수이고, [0,1,2]는 반복 횟수를 결정짓는 정보입니다!

즉,

**for <변수> in <범위>:**

**<for에 속하는 문장1>**

**<for에 속하는 문장2>**

이런식으로 쓰면 됩니다!

하지만 범위가 막 1부터 100까지! 이러면 1,2,3,4,... 이렇게 하나하나 다 못쓰니까 다른 방법을 알아봐야겠죠??

그럴 때 쓰는 것이 바로 **range**라는 것입니다!

[1,2,3,4,5,6,7,8,9,10]은 range(1,11)로 바꿔서 쓸 수 있는데요!

그럼 for문을 조금 고쳐보면



```
for i in range(1,11):  
    print(i)
```

이렇게 간단하게 고쳐서 쓸 수 있습니다!

위에서 추측할 수 있듯이 range(1,11)은 for의 변수 i에 1부터 11 이전까지의 값을 넣어서 반복을 진행하라는 의미를 갖고있습니다!

즉, range(n,m)은 변수 i에 n부터 m이전, 즉 m-1까지의 값을 넣어서 반복을 진행하라는 의미입니다!

```
for i in range(0,3):  
    print("Happy")
```

이 경우는 range(0,3)인데요! 이건 0,1,2 , 즉 3번 반복한다는 뜻입니다!

이는

```
for i in range(3):  
    print("Happy")
```

이렇게 바꿔서 써줘도 됩니다~

만약에 반복횟수를 입력받고싶다! 그러면 앞에서 배운 것을 활용해서 코드를 작성해보면

```
def happy(cnt):  
    for i in range(cnt):  
        print("Happy")  
def main():  
    happy(int(input("반복 횟수: ")))  
main()
```

이렇게 작성해줄 수 있습니다!

아까 eval을 쓰는게 취약하다고 하기도하고,,, 그래서 저는 input을 int로 감싸줘서 입력값을 int로 바꿔줬습니다!

그럼 이번 포스팅은 여기서 마치도록 하겠습니다 :D



♡ 1 📄 📄 📄



'언어 > Python3' 카테고리의 다른 글

**Chapter 03. 프로그램 사용자로부터의 입력 그리고 코드의 반복** (0)

2021.01.05

Chapter 02. 간단한 함수 만들기 (0)

2021.01.05

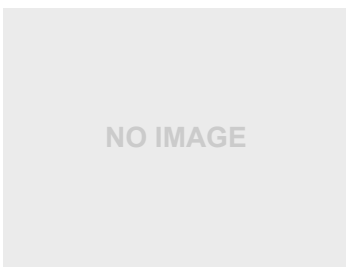
Chapter 01. 파이썬에게 질문하기 (0)

2021.01.05

파이썬 공부 시작! (0)

2021.01.05

## '언어/Python3' Related Articles



**Chapter 02. 간단한 함수 만들기**

**Chapter 01. 파이썬에게 질문하기**

**파이썬 공부 시작!**

여러분의 소중한 댓글을 입력해주세요.

댓글달기

