

Python 4주차 실습과제

과목명 : 파이썬프로그래밍

학과 : 정보보안공학과

학번 : 202121556

성명 : 곽지현

In [50]: # 1. 다음 코드의 변수 a, b, c, d의 값과 자료형의 종류를 코드를 실행해 확인해보세요.

```
a = str(0) # a에 문자열 0을 대입
b = int(a) # b에 a(0)를 정수형으로 대입
c = str(b) + str("3") # c에 b(0)의 문자열 플러스 문자열 "3"을 대입
d = int(c+a) # d에 정수형 c(3) 플러스 정수형 a(0)를 대입
e = 2 / 1 # e에 2 나누기 1을 대입

print(a, type(a)) # a의 값과 a의 자료형인 str을 출력
print(b, type(b)) # b의 값과 b의 자료형인 int를 출력
print(c, type(c)) # c의 값과 c의 자료형인 str을 출력
print(d, type(d)) # d의 값과 d의 자료형인 int를 출력
print(e, type(e)) # e의 값과 e의 자료형인 float를 출력
```

```
0 <class 'str'>
0 <class 'int'>
03 <class 'str'>
30 <class 'int'>
2.0 <class 'float'>
```

In [5]: # 2. 문자열 내장함수 split() 함수의 이해를 위해 아래 코드를 실행해 결과를 확인해보세요.

```
str = "Hello Python Programming" # 문자열변수에 "Hello Python Programming"를 대입
print(str.split()) # 공백을 기준으로 문자열을 나눠서 출력
print(str.split(" ")) # 공백을 기준으로 문자열을 나눠서 출력
print(str.split("o")) # "o"를 기준으로 문자열을 나눠서 출력
```

```
['Hello', 'Python', 'Programming']
['Hello', 'Python', 'Programming']
['Hell', ' Pyth', 'n Pr', 'gramming']
```

In [51]: # 3. 문자열의 길이를 알아낼 수 있는 len() 함수의 이해를 위해 아래 코드를 실행해 결과를 .

```
user = input() # 입력받기
string_len = len(user) # 변수에 입력받은 문자의 길이를 대입
print("입력된 문자열 ", user + "의 길이는", string_len, "입니다.") # 입력받은 데이터와 더
print("문자열 연산으로 추가된 문자열의 길이는", len(user+"!!!"), "입니다")
# 입력받은 데이터에 "!!!"를 더하여 길이를 구해 출력
print("하지만, !!!가 추가된 문자열은 변수에 저장한 것이 아니라서 user값은", user, "입니다")
```

23

입력된 문자열 23의 길이는 2 입니다.

문자열 연산으로 추가된 문자열의 길이는 5 입니다

하지만, !!!가 추가된 문자열은 변수에 저장한 것이 아니라서 user값은 23 입니다

In [7]: # 4. 이스케이프 문자를 활용한 예제를 따라 해 결과를 확인해보세요.

```
print("이게 원본\\r 이게 수정") # "이게 원본\\r 이게 수정"을 출력
print("c:\\Program Files (x86)\\") # "c:\\Program Files (x86)\\"을 출력
```

이게 수정

c:\\Program Files (x86)\\

In [26]: # 5. 복합 대입 연산자가 어떤 역할을 하는지 아래 코드를 실행해 결과를 확인해보세요.

```
a = 10 # a에 10을 대입
a = a + 10; print(a) # a에 10을 더한 a의 값을 출력
a = a - 10; print(a) # a(20)에 10을 뺀 a의 값을 출력
a = a * 10; print(a) # a(10)에 10을 곱한 a의 값을 출력
```

```
a = a / 10; print(a) # a(100)를 10으로 나눈 a의 값을 출력

a = 10 # a에 10을 대입
a += 10; print(a) # a에 10을 더한 후 대입, a의 값을 출력
a -= 10; print(a) # a(20)에 10을 뺀 후 대입, a의 값을 출력
a *= 10; print(a) # a(10)에 10을 곱한 후 대입, a의 값을 출력
a /= 10; print(a) # a(100)에 10d를 나눈 후 대입, a의 값을 출력
```

```
20
10
100
10.0
20
10
100
10.0
```

In [10]: # 6. 자료형 캐스팅을 이해하기 위해 아래 코드를 실행해 결과를 확인해보세요.

```
a = 100 # a에 100을 대입
print(a, type(a)) # a의 값과 a의 자료형인 int를 출력

a = str(a) # a에 문자열 a(100)을 대입
print(a, type(a)) # a의 값과 a의 자료형인 str을 출력

a = int(a) / 10 # a(100) 나누기 10을 대입
print(a, type(a)) # a의 값과 a의 자료형인 float를 출력

a = "1", str(a) # "1", a(10.0)의 문자열을 대입
print(a, type(a)) # a의 값과 a의 자료형인 tuple을 출력

a = float(a) / 14 # a(10.0) 나누기 14를 대입
print(a, type(a)) # a의 값과 a의 자료형인 float를 출력

a = int(a) # 정수 a를 대입
print(a, type(a)) # a의 값과 a의 자료형인 int를 출력
```

```
100 <class 'int'>
100 <class 'str'>
10.0 <class 'float'>
('1', '10.0') <class 'tuple'>
0.7142857142857143 <class 'float'>
0 <class 'int'>
```

In [62]: # 7. 문자열 인덱싱, 슬라이싱을 이해하기 위해 아래 코드를 실행해 결과를 확인해보세요.

```
str = "영일이삼사오육칠팔구십" # 문자열 "영일이삼사오육칠팔구십" 대입
print(str[0]) # 첫번째 문자열 출력
print(str[:1]) # 0번째 부터 1번째 앞까지 문자열 출력
print(str[:2]) # 0번째 부터 2번째 앞까지 문자열 출력
print(str[0:10]) # 0번째 부터 10번째 앞까지 문자열 출력
print(str[0:-1]) # 0번째 부터 마지막 -1 까지 문자열 출력
print(str[5:]) # 5번째 부터 마지막까지 문자열 출력
print(str[-5:]) # 마지막에서 -5 부터 마지막까지 문자열 출력
```

영
영
영일
영일이상사오육칠팔구
영일이상사오육칠팔구
오육칠팔구십
육칠팔구십

```
In [69]: # 8. 포맷스트링을 이해하기 위해 다음 코드를 실행해 결과를 확인해보세요.
# 포맷스트링 : {~~~}.format()의 ~~ 부분
# 숫자를 문자열로 변환

print("{: d}".format(33)) # 양수 출력 (기호 부분 공백)
print("{: d}".format(-33)) # 음수 출력 (기호 부분 공백)

print('Wn') # 줄 바꿈

print("{:7d}".format(35)) # 7칸 출력
print("{:+7d}".format(35)) # 기호를 붙이고 7칸 출력
print("{:=-7d}".format(-35)) # 기호를 앞으로 밀고 7칸 출력

print('Wn') # 줄 바꿈

print("{:07d}".format(35)) # 0을 채우고 7칸 출력
print("{:07d}".format(-35)) # 0을 채우고 7칸 출력
print("{:+07d}".format(35)) # 기호를 붙이고 0을 채우고 7칸 출력

print('Wn') # 줄 바꿈

print("|{:<25s}|{:^25s}|{:>25s}|".format("왼쪽정렬", "가운데정렬", "오른쪽정렬"))
# 첫번째는 25칸에 제일 왼쪽에서 출력, 두번째는 25칸에 가운데에서 출력, 세번째는 25칸에
```

33
-33

.... 35
.... +35
-.... 35

0000035
-000035
+000035

|왼쪽정렬|.....|..... 가운데정렬|.....|..... 오른
쪽정렬|

```
In [35]: #9. 단일연산자의 이해를 위해 다음 코드를 실행하여 결과를 확인해보세요.
# 산술 연산자
print("= " * 50) # "="을 50개 출력
print(10 % 4) # 10을 4로 나누었을 때 나머지를 출력
print(2 ** 10) # 2의 10승을 출력

# 비교 연산자
print("a" in "AAA") # "AAA"안에 "a"가 없어서 False를 출력
print(0 == "0") # 0은 문자열 "0"이 아니어서 False를 출력
print(0 < 5) # 0은 5보다 작으므로 True를 출력
```

```
=====
2
1024
False
False
True
```

```
In [37]: # 10. 사용자 입력의 이해를 위해 다음 코드를 실행하여 결과를 확인해보세요.
```

```
user = int(input()) # 숫자 입력받기
print(user, type(user)) # 입력받은 숫자와 입력받은 자료형 출력

user, user2 = input().split() # 공백 기준 두 단어 입력받기
print("user 변수값:", user, "user2 변수값:", user2) # 입력받은 숫자 출력
```

```
23
23 <class 'int'>
45 56
user 변수값: 45 user2 변수값: 56
```