

PYTHON TUTORING #3

School of Computing, KAIST & 대덕고등학교 빛나리

INTRO

- ① 저번 시간 REVIEW
- ② `string` 자료형
- ③ `list` 자료형
- ④ 프로그래밍 예제

Conditional Statement REVIEW

현재 조건을 판단하여

특정 조건을 만족할 때만 실행되는 코드

if 조건문:

elif 조건문:

else:

예시

```
if grade >= 90:  
    print("A")  
elif grade >= 80:  
    print("B")  
elif grade >= 70:  
    print("C")  
else:  
    print("F")
```

Conditional Statement REVIEW

비교연산자	설명
<code>x < y</code>	x가 y보다 작다
<code>x > y</code>	x가 y보다 크다
<code>x == y</code>	x와 y가 같다
<code>x != y</code>	x와 y가 같지 않다
<code>x >= y</code>	x가 y보다 크거나 같다
<code>x <= y</code>	x가 y보다 작거나 같다

연산자	설명
<code>x or y</code>	x와 y 둘중에 하나만 참이면 참이다
<code>x and y</code>	x와 y 모두 참이어야 참이다
<code>not x</code>	x가 거짓이면 참이다

예시

```
if x < z and y < z:
    print("Biggest z")
```

```
if not y == 0:
    print("Can divide")
    z = x / y
```

Loop Control REVIEW

같은 코드를 여러 번 반복시키고 싶을 때,
for, while을 이용한다.

n번 반복하고 싶다면

for 변수이름 **in range(n):**

예시

```
for i in range(10):  
    print("hello")  
  
for i in range(10):  
    line = "9 * "  
    line += str(i)  
    line += " = "  
    line += str(9 * i)  
    print(line)
```

Loop Control REVIEW

`while`은 조건문과 함께 사용

`while` 조건문:
반복할 문장

예시

```
i = 0
while i < 10:
    print(i)
    i = i + 1

count = 0
while True:
    count += 1
    print(count)
```

예제

구구단 전체를 출력하는 프로그램을 작성하시오.

Hint:

```
line = str(i) + " * " + str(j) + " = " + str(i * j)
print(line)
```

string 자료형

string이란 문자들로 구성된 문자열

따옴표로 둘러싸여 있으면 string

string들의 연산은 숫자와 다르게 정의

len(string)은 주어진 string의 길이를 반환하는 기본 기능

예시

```
str1 = "ABC"
str2 = "123"
str3 = "I love Python"
str4 = "내 이름은?"

print(str1 + str2)
print(str3 * 2)
print(str1 == str2)
print(len(str3))
```


string indexing

`string[n]`은 $n+1$ 번째 문자를 의미한다.

`str3[0]: "I"`

`str3[1]: " "`

`str3[3]: "o"`

...

`str3[12]: "\n"`

예시

```
str3 = "I love Python"
```

```
print(str3[0])
```

```
print(str3[1])
```

```
print(str3[3])
```

```
print(str3[12])
```

string slicing

`string[a:b]`는 `a+1`번째 문자부터 `b`번째 문자까지를 의미한다.

```
str3[2:10] = "ove Pyt"
```

```
str3[:9] = "I love Py"
```

```
str3[4:] = "ve Python"
```

```
str3[:] = "I love Python"
```

예시

```
str3 = "I love Python"
print(str3[2:10])

print(str3[:9])
print(str3[4:])

print(str3[:])
```

Loop Control with string

for 변수이름 **in** stringname:

혹은

```
for i in range(len(stringname)):
    stringname[i]
```

예시

```
str3 = "I love Python"
```

```
for s in str3:
    print(s)
```

```
for i in range(len(str3)):
    print(i)
    print(str3[i])
```

string functions

string 뒤에 **.함수이름(값)**을 붙여서 원하는 기능을 호출할 수 있다.

count는 string에 괄호 안의 요소가 등장한 횟수를 반환한다.

find는 괄호 안의 요소가 string의 몇 번째에 처음으로 등장하는지 반환한다.

예시

```
str3 = "I love Python"

c = str3.count("o")
print(c)

i = str3.find("P")
print(i)
print(str3[i])
```

예제

`input` 함수를 이용해서 문자열을 받은 뒤,

문자열에 "A"가 얼마나 많이 등장하는지 세는 프로그램을 작성하여라.

list 자료형

list는 하나 이상의 값을 가지는 자료형

대괄호로 감싸져 있으면 list

list 안의 요소는 숫자, string, list
관계 없이 전부 가능

list 사이의 연산도 따로 적용

예시

```
odd = [1, 3, 5, 7, 9]
```

```
empty = []
```

```
word = ["even", "odd"]
```

```
arr = [1, 2, "number"]
```

```
print(odd + word)
```

```
print(word * 2)
```

list indexing

`list[n]`은 $n+1$ 번째 요소를 의미한다.

`arr[0]`: 1

`arr[1]`: 2

`arr[2]`: "number"

예시

```
arr = [1, 2, "number"]
```

```
print(arr[0])
```

```
print(arr[1])
```

```
print(arr[2])
```

list slicing

`list[a:b]`은 $a+1$ 번째 요소부터 b 번째 요소까지를 의미한다.

`arr[:2]:` [1, 2]

`arr[1:]`: [2, "number"]

`arr[:]`: [1, 2, "number"]

예시

```
arr = [1, 2, "number"]
```

```
print(arr[:2])
```

```
print(arr[1:])
```

```
print(arr[:])
```


Loop Control with list

for 변수이름 **in** listname:

혹은

```
for i in range(len(listname)):
    listname[i]
```

예시

```
odd = [1, 3, 5, 7, 9]
```

```
for o in odd:
    print(o)
```

```
for i in range(len(odd)):
    print(i)
    print(odd[i])
```

list functions

string과 마찬가지로 `.` 을 이용하여 특정 기능을 호출할 수 있다.

`append`는 `list`의 맨 끝에 괄호 안의 요소를 추가한다.

`remove`는 괄호 안의 요소가 `list` 안에 있으면 첫 번째 요소를 지운다.

예시

```
arr = [1, 2, "number"]

arr.append(3)
print(arr)

arr.append(2)
print(arr)
arr.remove(2)
print(arr)
```

예제

- 1) `input` 함수를 이용하여 횃수에 해당하는 숫자를 입력 받아,
주어진 횃수만큼 입력 받은 모든 문자열을 `list`에 넣어라.
- 2) 해당 `list` 안에 같은 문자열이 있는지 검사하여라.