Fastcampus

컴퓨터공학 입문 스쿨

Python Basic_Day3

2017.3.29

Tuple, Dictionary

Tuple

Tuple은 괄호를 이용해 선언할 수 있습니다.

```
tuple1 = (1, 2, 3, 4)
```

tuple은 삭제나 추가가 불가능합니다.

```
del tuple[1]
tuple1[1] = 'c'
```

tuple끼리 더하거나 반복하는 것은 가능합니다.

```
tuple2 = (5, 6)
print(tuple1 + tuple2)

print(tuple1 * 3)
```

tuple은 값을 편하게 바꿀 수 있습니다.

```
x = 1
y = 2

#이렇게 하면 안됩니다.
x = y
y = x

temp = x
x = y
y = temp

(x, y) = (y, x)
print(x,y)
```

혹은 함수에서 하나 이상의 값을 반환할 때 사용합니다.

```
def quot_and_rem(a,b):
    quot = a // b
    rem = a % b
    return (quot, rem)

(quot, rem) = quot_and_rem(3,10)
```

dictionary의 선언

```
dict1 = {}
print(dict1)
```

dictionary의 구조

```
some_dict = {
        'key':'value',
        'key':'value',
        'key':'value',
        'key':'value',
}
```

dictionary는 key와 value로 이루어져 있으며, 추가하는 법은 다음과 같습니다.

```
dict1 = {'name': 'foo bar'}
print(dict1)

score = {'korean': 95, 'math': 100, 'science': [80, 70, 90, 60]}
print(score)

dict1['english'] = "pass"
print(score)
```

요소 삭제는 del을 활용합니다.

```
del score['math']
print(score)
```

key를 활용해 value를 출력하는 법을 알아봅시다.

```
print(score['korean'])
```

key만 출력하는 법을 알아봅시다.

```
print(score.keys())
```

value만 출력할땐 이렇게 합니다.

```
print(score.values())
```

key와 value를 함께 출력합니다.

```
print(score.items())
```

조건문

If

```
if 조건:
        실행문
===================
if 조건1 and 조건2:
        실행문
===================
if 조건1 or 조건2:
        실행문
===================
if not 조건:
        실행문
```

Comparison Operators

```
x == n
x != n
x < n
x > n
x <= n
x >= n
```

else

```
if 조건:
        실행문1
else:
        실행문2
```

else if

```
if 조건1:
        실행문1
else:
        if 조건2:
                실행문2
        else:
                실행문3
```

elif

```
if 조건1:
        실행문1
elif 조건2:
        실행문2
elif 조건3:
        실행문3
...
else:
        실행문n
```

numguess

```python
import random


answer = random.randint(1,100)
print(answer)
```

numguess

```
username = input("Hi there, What's your name?? ")
guess = eval(input("Hi, "+ username + "guess the number: "))

if guess == answer:
        print("Correct! The answer was ", str(answer))
else:
        print("That's not what I wanted!! The answer was ", str(
```

numguess advanced!!

how to make it with more fun??

For, while

```
for 변수 in (리스트 or 문자열):
        실행문1
    ...
```

```
for i in ["python", "java", "golang"]:
        print(i)
```

For, while

```
sum = 0
for i in range(1,11):
    sum += i
    print(sum)
```

List Comprehension

```
result = [i for i in range(1,11)]
print(result)
```

For, while

```
while 조건:
        실행문1
        ...
```

```
while name != "foo bar":
        name = input("What's your name? ")
        print("Hi, " + name + "So, where is foo bar?")
```

```
while 1:
        print("Hello world!")
```

Fizzbuzz

```
num = eval(input("type the number: "))

for i in range(1, num + 1):
        if i % 15 == 0:
                print("fizzbuzz")
        elif i % 3 == 0:
                print("fizz")
        elif i % 5 == 0:
                print("buzz")
        else:
                print(i)
```

Refactoring numguess

```
import random


answer = random.randint(1,100)
username = input("Hi there, What's your name?? ")

while True:
        guess = eval(input("Hi "+ username + ", guess the number

        if guess == answer:
                print("Correct! The answer was ", str(answer))
                break
        else:
                print("That's not what I wanted!! Try again!!")
```

give a hint!!

```python
import random


answer = random.randint(1,100)
username = input("Hi there, What's your name?? ")

while True:
    guess = eval(input("Hi, "+ username + "guess the number: "))

    if guess == answer:
        print("Correct! The answer was ", str(answer))
        break
    elif guess > answer:
        print("Too high!! Try again!!")
    elif guess < answer:
        print("Too Low!! Try again!!")
```

limit trial

```python
import random


answer = random.randint(1,100)
username = input("Hi there, What's your name?? ")
trial = 5
while trial:
    guess = eval(input("Hi, "+ username + ". guess the number: "

    if guess == answer:
        print("Correct! The answer was ", str(answer))
        break
    elif guess > answer:
        trial -= 1
        print("Too high!! Try again!!(%d times left)" % (trial))
    elif guess < answer:
        trial -= 1
        print("Too Low!! Try again!!(%d times left)" % (trial))

if trial == 0:
    print("You are Wrong! The answer was ", str(answer))
```

Caesar Cipher

```python
import string
from string import ascii_uppercase as up_case
from string import ascii_lowercase as lo_case
```

encrypt code

```python
def caesar(s, k, decode = False):
    if decode: k = 26 - k
    return s.translate(
        str.maketrans(
            up_case + lo_case,
            up_case[k:] + up_case[:k] +
            lo_case[k:] + lo_case[:k]
            )
        )
```

get input and put output

```
while True:
    encrypt_key = int(input("Decide Secret number: "))
    msg = input("give me the some words to encrypt: ")
    print("encrypted message: ", caesar(msg, encrypt_key))
    print("decrypted message: ", caesar(caesar(msg, encrypt_key)
    exit = input("press any key to continue or 'q' to quit: ").l
    if 'q' in exit:
        break
```