Fastcampus

컴퓨터공학 입문 스쿨

Python Basic_Day3

2017.3.29

어제의 숙제!

Monty Hall Simulation

Monty Hall Simulation

```
import random


stay = 0
switch = 0
```

Monty Hall Simulation

```
for i in range(1000):
    doors = [1,0,0]
    random.shuffle(doors)

    choice = random.randrange(3)

    user = doors[choice]

    if user == 1:
        stay = stay + 1
    else:
        switch = switch + 1
```

Monty Hall Simulation

```
print("stay: %4d" % stay)
print("switch: %4d" % switch)
```

Ethiopian Multiplication

2로 나누고 곱하는 과정으로 두 수의 곱을 구현하는 방법

https://en.wikipedia.org/wiki/Ancient_Egyptian_multiplication

```
12 *     7       struck  ---
 6      14       struck  ---
 3      28       keep     28
 1      56       keep     56
--> 28 + 56 = 84
```

Ethiopian Multiplication

```
numbers = str(input("two nums with space: ")).split()

result = 0
num1 = int(numbers[0])
num2 = int(numbers[1])
```

Ethiopian Multiplication

```
while num1 >= 1:
    if num1 % 2 == 0:
        print("%4d %7d struck" % (num1, num2))
    else:
        print("%4d %7d keep" % (num1, num2))
        result += num2

    num1 = num1 // 2
    num2 = num2 * 2
```

Ethiopian Multiplication

```
print("The result is ", result)
```

Toggl

https://blog.toggl.com/wp-content/uploads/2016/12/toggl-it-jobs-explained-with-changing-lightbulb.jpg

https://assets.toggl.com/images/toggl-how-to-save-the-princess-in-8-programming-languages.jpg

Dictionary, Set

dictionary의 선언

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

dictionary는 key와 value로 이루어져 있으며, 추가하는 법은 다음과 같습니다.

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

dict1 = {'korean': 95, 'math': 100, 'science': [80, 70, 90, 60]}
print(dict1)

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

요소 삭제는 del을 활용합니다.

```
del dict1['math']
print(dict1)
```

key를 활용해 value를 출력하는 법을 알아봅시다.

```
print(dict1['korean'])
```

key만 출력하는 법을 알아봅시다.

```
print(dict1.keys())
```

value만 출력할땐 이렇게 합니다.

```
print(dict1.values())
```

key와 value를 함께 출력합니다.

```
print(dict1.items())
```

Small Quiz

A = 'fastcampus'

B = 'python'

A ∪ B

A ∩ B

A - B

A Δ B

Set

- 수학 집합 연산을 쉽게 하기 위해 만든 자료형
- 순서없음
- 중복없음

Set

Set 선언

```
ppap = {'pen', 'apple', 'pineapple', 'pen'}
print(ppap)
```

```
'apple' in ppap
'applepen' in ppap
```

```
pineapple = set('pineapple')
pineapple
```

Set

A = 'fastcampus'

B = 'python'

A ∪ B == `A | B`

A ∩ B == `A & B`

A - B == `A - B`

A Δ B == `A ^ B`

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!!

```
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

```
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))
```

List Comprehension

존재하는 리스트를 활용하여 새로운 리스트를 생성하는 방법

비슷한 표현들

- Set Comprehension
- Dictionary Comprehension
- Parallel list Comprehension

List Comprehension

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        doubled_list.append(i * 2)
```

List Comprehension

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        doubled_list.append(i * 2)
```

```
doubled_list = []
```

List Comprehension

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        doubled_list.append(i * 2)
```

```
doubled_list = [i * 2]
```

List Comprehension

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        doubled_list.append(i * 2)
```

```
doubled_list = [i * 2 for i in old_list]
```

List Comprehension - another example

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        if i % 2 == 0:
                doubled_list.append(i * 2)
```

List Comprehension - another example

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        if i % 2 == 0:
                doubled_list.append(i * 2)
```

```
doubled_list = []
```

List Comprehension - another example

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        if i % 2 == 0:
                doubled_list.append(i * 2)
```

```
doubled_list = [i * 2]
```

List Comprehension - another example

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        if i % 2 == 0:
                doubled_list.append(i * 2)
```

```
doubled_list = [i * 2 for i in old_list]
```

List Comprehension - another example

```
old_list = [1, 2, 3, 4, 5,]

doubled_list = []
for i in old_list:
        if i % 2 == 0:
                doubled_list.append(i * 2)
```

```
doubled_list = [i * 2 for i in old_list if i % 2 == 0]
```

Mini Project!

- List Comprehension으로 FizzBuzz 한 줄로 구현하기