

1 2 기 정 규 세 셴

T o B i g ' s 1 1 기

권혜민 임채빈 한재연

Algorithm

두뇌 풀 가동

Problem 1 | 여행

이영전은 여행을 떠나기 위해 최대 X kg 의 짐을 넣을 수 있는 배낭을 샀다. 이영전에게는 N 개의 짐이 있는데, 배낭에 최대한 많은 무게의 짐을 넣을 것이다. 이 때 배낭에 들어간 짐의 무게의 총 합을 구하라.

입력

1 이상 10 이하 자연수 N 과 1 이상 100 이하 자연수 X 가 한 줄에 공백을 구분하여 주어진다. 다음 줄에는 자연수인 각 짐의 무게가 공백을 구분하여 주어진다.

출력

정답을 출력하라.

Problem 1 | 여행

예제 입력1

3 15

1 16 12

출력

13

예제 입력2

6 76

19 42 73 4 55 66

출력

74

Problem 2 | BIG DATA 다루기

 study_HMS.csv	2019-08-28 오...	Microsoft Excel...	2,465,098KB
---	-----------------	--------------------	-------------

2.4 GB의 스터디룸 이용 데이터

Problem 2 | BIG DATA 다루기

변수 설명

study_start_day, study_end_day : 집계 시작일로부터 이용 시작 / 종료 날짜

study_start_time, study_end_time : 날짜를 무시한 이용 시작 / 종료 시각

Hashed : 해당 스터디룸 사용자들의 암호화된 ID, comma(,)로 구분

	study_start_day	study_start_time	study_end_day	study_end_time	hashed
0	1	09:14:58.558	1	09:41:30.200	967393e81d99ce8e577ee130b7ce8e4fd45e3e9cecb560...
1	17	11:05:05.176	17	13:07:42.515	02181a0c962f34f019bc9d5b582fb0ec79b1441f96aa4d...
2	20	02:18:43.172	20	02:28:58.177	86022904c5cf72a54978479c94041f4256d6c3c2a1f71c...
3	22	09:22:01.936	22	09:47:40.192	aafb40d212fe18ff4eafb82fdcf3b53f2161cb3ce59de4...
4	26	06:29:21.182	26	06:50:55.004	c87c2fad141edf323f3787335b54be22945a02fe052448...

Problem 2 | BIG DATA 다루기

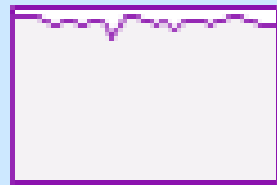
```
df['hashed'][0] = 967393e81d99ce8e577ee130b7ce8e4fd45e3e9cecb560de427ede6ea49e024f,  
a0b6ecbec654b18fe36ebe6230e25a653fb12125733583d012741572134447f4,  
3193ab18168bcadbcb8342c06c4a35fa0d6e58d9619fe805fb811fc4e6562fef
```

0번째 사용기록은 3명의 사용자가 참여함

각 hash값이 가리키는 사용자 마다의 다양한 정보를 알고 싶다면
hashed안의 사용자들을 구분해야 함

Problem 2 | BIG DATA 다루기

```
In [*]: 1 df['hashed'].apply(lambda x : x.split(','))
```



메모리
7.2/7.9GB (91%)

Split만으로 Memory Error가 발생

Problem 2 | BIG DATA 다루기

왜 memory error가 발생한걸까?

voice.csv

	A	B	C	D	E	F	G	H
1	meanfreq	sd	median	Q25	Q75	IQR	skew	kurt
2	0.059781	0.0642413	0.0320269	0.0150715	0.0901934	0.075122	12.863462	274.40
3	0.0660087	0.06731	0.0402287	0.0194139	0.0926662	0.0732523	22.423285	634.61
4	0.0773155	0.0838294	0.0367185	0.0087011	0.131908	0.123207	30.757155	1024.9
5	0.1512281	0.0721106	0.1580112	0.0965817	0.2079553	0.1113735	1.2328313	4.1772
6	0.1351204	0.0791461	0.1246562	0.0787202	0.2060449	0.1273247	1.1011737	4.3337
7	0.1327864	0.0795569	0.1190898	0.067958	0.2095916	0.1416336	1.9325624	8.308
8	0.1507623	0.0744632	0.1601064	0.0928989	0.2057181	0.1128191	1.5306432	5.9874
9	0.1605143	0.0767669	0.1443368	0.1105322	0.2319619	0.1214297	1.3971564	4.7666
10	0.1422394	0.0780185	0.1385874	0.0882063	0.2085874	0.1203812	1.0997462	4.0702
11	0.1343288	0.08035	0.1214513	0.07558	0.2019571	0.1263771	1.1903684	4.7873
12	0.1570205	0.0719429	0.1681602	0.1014299	0.2167398	0.1153098	0.9794423	3.9742

```
In [*]: 1 df['hashed'].apply(lambda x : x.split(','))
```

python dataframe

split

.avi

.hwp

OS

Problem 2 | BIG DATA 다루기

왜 memory error가 발생한걸까?

	A	
1	meanfreq	sc
2	0.059781	0
3	0.0660087	
4	0.0773155	0
5	0.1512281	0
6	0.1351204	0
7	0.1327864	0
8	0.1507623	0
9	0.1605143	0
10	0.1422394	0
11	0.1343288	
12	0.1570205	0

```

MemoryError                                Traceback (most recent call last)
<ipython-input-6-15641bla4809> in <module>()
      1 K5 = nx.convert_node_labels_to_integers(G_fb, first_label=2)
      2 G_fb.add_edges_from(K5.edges())
--> 3 c = list(k_clique_communities(G_fb, 3))
      4 list(c[0])

~\Anaconda3\lib\site-packages\networkx\algorithms\community\kclique.py in k_clique_communities
     69     for adj_clique in _get_adjacent_cliques(clique, membership_dict):
     70         if len(clique.intersection(adj_clique)) >= (k - 1):
--> 71             perc_graph.add_edge(clique, adj_clique)
     72
     73     ..

```

Problem 2 | BIG DATA 다루기

어떻게 Large Dataset을 다룰 수 있을까?

한번에 전체 메모리를 참조 -> 비효율적

한번에 하나의 데이터만 참조 -> 효율적

Problem 2 | BIG DATA 다루기

Generator

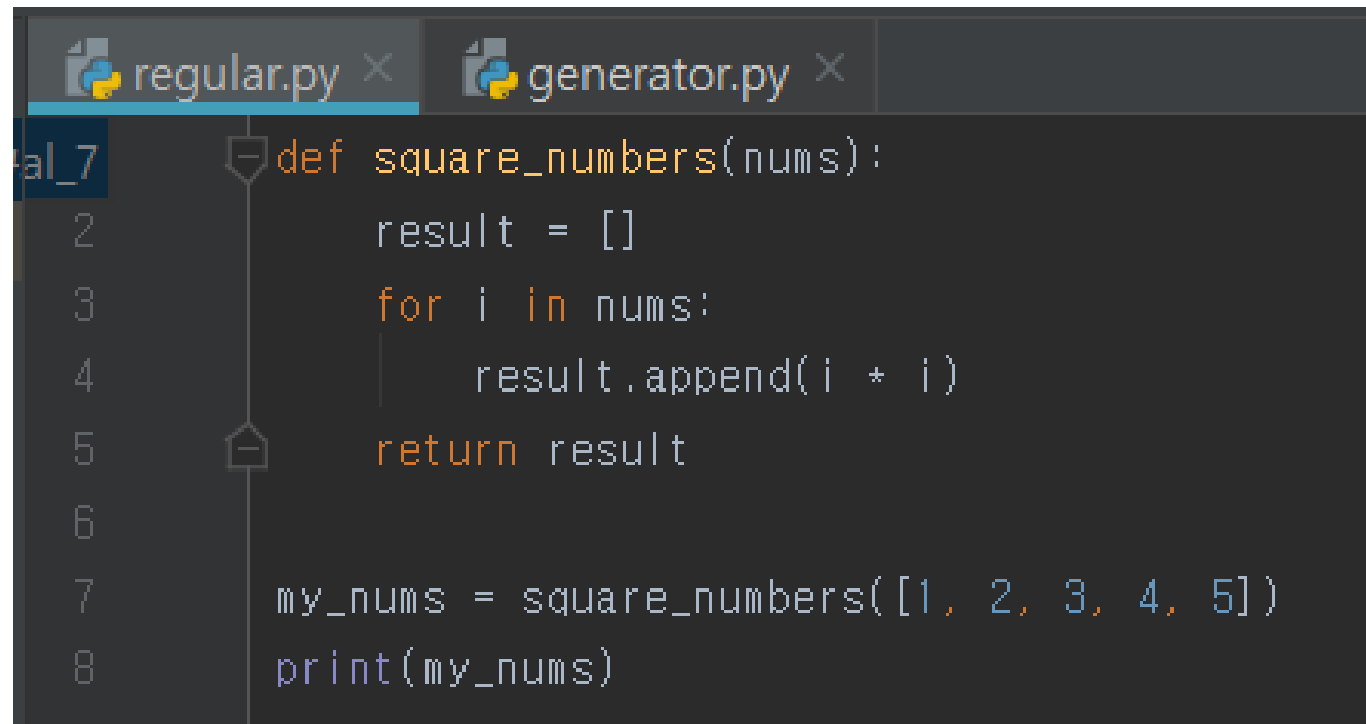
“반복자(iterator)와 같은 루프의 작용을 컨트롤하기 위해 쓰여지는 특별한 함수 또는 루틴 ”

- Iterator란? 반복 가능한 객체, `next()` 함수를 이용해 순차적으로 값을 가져온다.
- Return 대신에 `yield` 구문 -> 한번 호출될 때마다 하나의 값 만을 리턴

작은 메모리를 필요로 한다!

Problem 2 | BIG DATA 다루기

일반적인 코드



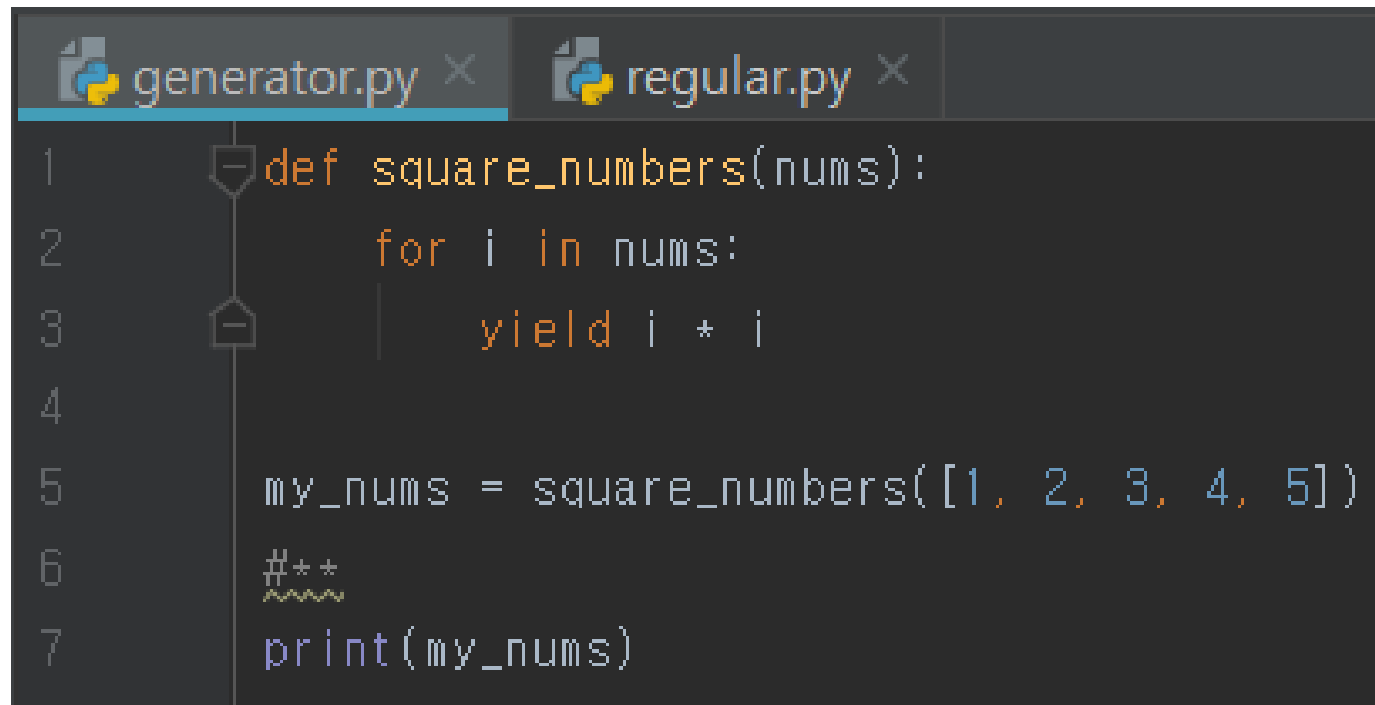
```
regular.py x generator.py x
1 def square_numbers(nums):
2     result = []
3     for i in nums:
4         result.append(i * i)
5     return result
6
7 my_nums = square_numbers([1, 2, 3, 4, 5])
8 print(my_nums)
```

리턴할 모든 값(리스트)를
메모리에 통째로 저장

```
[1, 4, 9, 16, 25]
```

Problem 2 | BIG DATA 다루기

Generator 사용



```
generator.py x regular.py x
1 def square_numbers(nums):
2     for i in nums:
3         yield i * i
4
5 my_nums = square_numbers([1, 2, 3, 4, 5])
6
7 ***
8 print(my_nums)
```

```
<generator object square_numbers at 0x00000018E47DFC1A8>
```

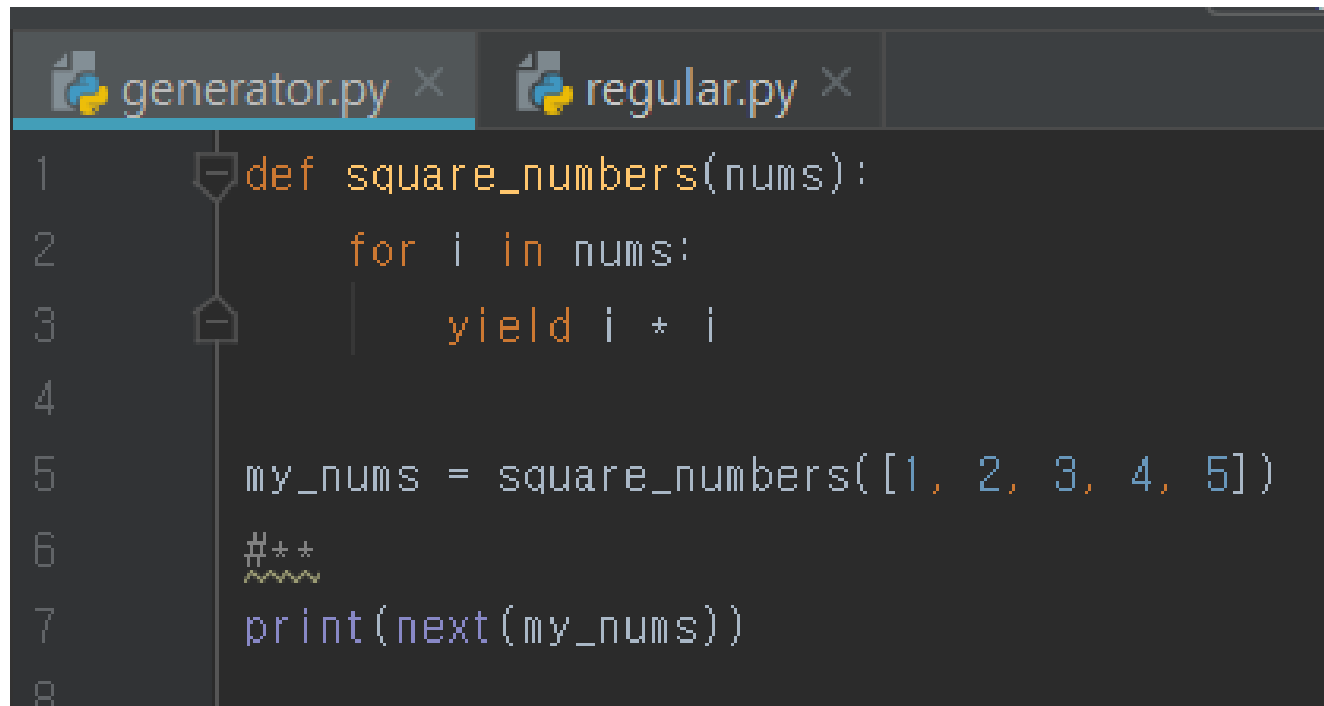
Generator라는 오브젝트가 리턴(모든 값을 저장하지 x)

아직 아무 계산도

안함

Problem 2 | BIG DATA 다루기

Generator사용



```
generator.py x regular.py x
1 def square_numbers(nums):
2     for i in nums:
3         yield i * i
4
5 my_nums = square_numbers([1, 2, 3, 4, 5])
6
7 print(next(my_nums))
8
```

1

Next()함수를 이용하여 다음 값을 확인할 수 있다.

Problem 2 | BIG DATA 다루기

Generator사용

```
generator.py x regular.py x
1 def square_numbers(nums):
2     for i in nums:
3         yield i * i
4
5 my_nums = square_numbers([1, 2, 3, 4, 5])
6
7 print(next(my_nums))
8 print(next(my_nums))
9 print(next(my_nums))
10 print(next(my_nums))
11 print(next(my_nums))
```

```
1
4
9
16
25
```

Problem 2 | BIG DATA 다루기

Generator 사용

For문 싫어요..

regular

```
my_nums = [x*x for x in [1, 2, 3, 4, 5]]

print(my_nums)

for num in my_nums:
    print(num)
```

generator

```
my_nums = (x*x for x in [1, 2, 3, 4, 5])

print(my_nums)

for num in my_nums:
    print(num)
```


Problem 2 | BIG DATA 다루기

Generator 실습코드

```
In [10]: #일반함수
def people_list(num_people):
    result = []
    for i in range(num_people):
        person = {
            'id': i,
            'name': random.choice(names),
            'major': random.choice(majors)
        }
        result.append(person)
    return result
```

```
In [11]: #generator
def people_generator(num_people):
    for i in range(num_people):
        person = {
            'id': i,
            'name': random.choice(names),
            'major': random.choice(majors)
        }
        yield person
```

```
In [14]: print('시작 전 메모리 사용량: {} MB'.format(mem_before))
print('종료 후 메모리 사용량: {} MB'.format(mem_after))
print('총 소요된 시간: {:.6f} 초'.format(total_time))
```

시작 전 메모리 사용량: 47.94140625 MB
종료 후 메모리 사용량: 319.171875 MB
총 소요된 시간: 6.142032 초

```
In [17]: print('시작 전 메모리 사용량: {} MB'.format(mem_before))
print('종료 후 메모리 사용량: {} MB'.format(mem_after))
print('총 소요된 시간: {:.6f} 초'.format(total_time))
```

시작 전 메모리 사용량: 47.94140625 MB
종료 후 메모리 사용량: 48.546875 MB
총 소요된 시간: 0.337646 초

Problem 2 | BIG DATA 다루기

과제설명

study_room_HMS.csv를 이용해서 각 사람별로 파생변수 만들기

Ex) 총 이용 횟수, 총 이용 시간, 평균 이용 시간

총 같이 이용한 인원수, 1회 이용에 참여한 인원수

인맥지도, 집계시작일을 월요일로 가정할 때 요일 별 데이터 등등..

Problem 2 | BIG DATA 다루기

reference

<http://schoolofweb.net/blog/posts/%ED%8C%8C%EC%9D%B4%EC%8D%AC-%EC%A0%9C%EB%84%88%EB%A0%88%EC%9D%B4%ED%84%B0-generator/>

http://www.datamarket.kr/xe/index.php?mid=board_jPWY12&page=2&document_srl=53545

<https://bluese05.tistory.com/56>

<https://pypi.org/project/tqdm/>

Q & A

들어주셔서 감사합니다.