



برنامه نویسی با پایتون

سعید محققى

06: Work with Files and Dictionaries

■ پروژه 6: محاسبه معدل از فایل‌های نمرات (average from files)

What will we learn? (Keywords)

- Get List of Files in a Directory
- Read / Write Text Files
- Dictionary Data Type
- Counter (in collections module)

1. خواندن فایل‌های متنی حاوی نمرات

2. محاسبه معدل هر فایل

3. نمایش معدل‌ها به ترتیب نزولی

4. ذخیره نام و معدل دانش‌آموزان در یک فایل متنی

کار با فایل متنی در پایتون

■ خواندن فایل متنی

روش پیشنهادی

```
with open("file.txt", "r") as f:  
    for line in f:  
        print(line.strip())
```

روش نامناسب

```
1. f = open("file.txt", "r")  
  
2. for line in f:  
    print(line.strip())  
  
3. f.close() → اجباری
```

بازگشت به ابتدای فایل `f.seek(0)`

| | | | |
|--------------------------------|------------------|-----|-------------|
| <code>s = f.read()</code> | خواندن کل فایل | >>> | خروجی: متن |
| <code>s = f.readline()</code> | خواندن یک خط | >>> | خروجی: متن |
| <code>s = f.readlines()</code> | خواندن تمام خطوط | >>> | خروجی: لیست |

کار با فایل متنی در پایتون

■ نوشتن در فایل متنی

روش پیشنهادی

روش نامناسب

```
1. f = open("file.txt", "w")  
2. f.write("some text...")  
3. f.close() → اجباری
```

```
with open("file.txt", "w") as f:  
    f.write("some text...")
```

`f.write("string")`

نوشتن کل متن در فایل

`f.writelines([list])`

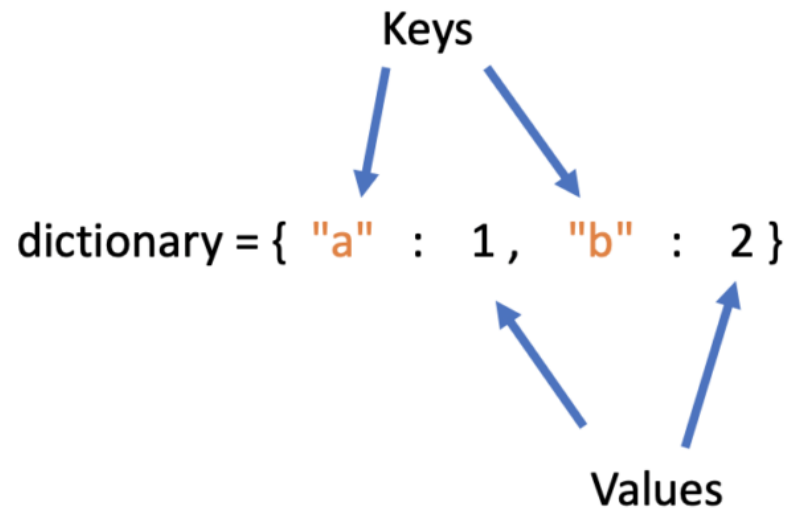
نوشتن آیتم‌های لیست در فایل → `f.writelines(["Line1", "\nLine2"])`

حالت‌های باز کردن فایل

| Mode | Description | Action |
|------|--|---|
| 'r' | Read >>> فقط خواندن (حالت پیش‌فرض) | اگر فایل موجود نباشد، خطا می‌دهد |
| 'w' | Write >>> نوشتن (بازنویسی محتوای فایل) | اگر فایل موجود نباشد، آن را می‌سازد |
| 'a' | Append >>> نوشتن (اضافه کردن به انتهای فایل) | اگر فایل موجود نباشد، آن را می‌سازد |
| 'r+' | Read/Write >>> خواندن و نوشتن | اگر فایل موجود نباشد، خطا می‌دهد |
| 'x' | Exclusive >>> ایجاد فایل و نوشتن | اگر فایل موجود <u>باشد</u> ، خطا می‌دهد |

نوع داده Dictionary در پایتون

- هر آیتم در دیکشنری شامل یک کلید (Key) و یک مقدار (Value) است
- Keys → Duplicate is not allowed / Only "str" or "int" or "tuple"
- Values → Duplicate is allowed / Any type of data



d = {'a':1, 'b':2}

d.keys()
↓
dict_keys(['a', 'b'])

d.values()
↓
dict_values([1, 2])

- تابع Counter در ماژول collections

```
from collections import Counter
```

- شمارش آیتم‌ها در list و tuple و dictionary

- مرتب‌سازی (sort) آیتم‌ها از بیشترین به کمترین

- List, Tuple → Counter(['B', 'B', 'A', 'B', 'C', 'A', 'B', 'B', 'A', 'C'])

Output → Counter({'B': 5, 'A': 3, 'C': 2})

- Dictionary → Counter({'A':2, 'B':4, 'C':3}).most_common()

Output → [('B': 4, 'C': 3, 'A': 2)]

<https://www.geeksforgeeks.org/counters-in-python-set-1/>

<https://www.geeksforgeeks.org/counters-in-python-set-2-accessing-counters/>