



(<https://github.com/MKarimi21/University-of-Bojnurd/tree/master/Data-Mining>).



(<https://github.com/MKarimi21/University-of-Bojnurd>).



(<https://github.com/MKarimi21/University-of-Bojnurd/tree/master/Data-Mining>).

Licence MIT (<https://github.com/MKarimi21/University-of-Bojnurd/blob/master/LICENSE>)

Create-by MKarimi (<https://www.mr-karimi.ir>)

Data Mining

درس داده کاوی

تمرین های درس داده کاوی، الگوریتم ها و کاربرد ها - دکتر مربی

دانشجو: مصطفی کریمی

#===== تمرین اول =====#

خواندن ورودی فایل text و یا نوشتن در آن.

```
In [3]: f = open("File/Data.txt", "r")
        print(f.read())
```

```
Data: Importan data
Made: 2/9. 99
Part: 1 - DM
```

```
In [10]: f = open("File/text_output.txt", "w")
        text = " This is test. "

        for i in range(0, 10):
            f.write("This is for loop: is [# {} #]\n".format(str(i + 1)))

        f.write("\nOut Put >> [ {} ]".format(str(text)))
```

Out[10]: 31

```
In [14]: f = open("File/text_append_mood.txt", "a")

f.write("This is text: first line\n")

for i in range(0, 5):
    f.write("\nThis is append [ {} ]\n".format(str(i + 1)))
```

در این روش می‌توان اول با ترمینال آدرس فایل در سیستم را به دست آورد یا با گزینه های پیش فرض ویندوز در قسمت "Home > Copy path" استفاده نمود

```
import os
os.getcwd()
```

```
In [26]: # data from folder

import os

data_list = os.listdir("File/list_data")

print(data_list)

data_even = []

for i in range(len(data_list)):
    data_even.append(data_list[i][0:-4])

print(data_even)

['This-01.txt', 'This-02.txt', 'This-03.txt', 'This-04.txt']
['This-01', 'This-02', 'This-03', 'This-04']
```

```
In [30]: with open("File/text_output.txt") as f:
        data = f.read()

print(f)

print(data)

<_io.TextIOWrapper name='File/text_output.txt' mode='r' encoding='UTF-8'>
This is for loop: is [# 1 #]
This is for loop: is [# 2 #]
This is for loop: is [# 3 #]
This is for loop: is [# 4 #]
This is for loop: is [# 5 #]
This is for loop: is [# 6 #]
This is for loop: is [# 7 #]
This is for loop: is [# 8 #]
This is for loop: is [# 9 #]
This is for loop: is [# 10 #]

Out Put >> [ This is test. ]
```

ورودی دیتا های Excel و CSV
برای مطالعه بیشتر با این library به لینک زیر مراجعه کنید:

Pandas-DataFrame: [Page-Link \(https://pandas.pydata.org/pandas-docs/stable/getting_started/dsintro.html#dataframe\)](https://pandas.pydata.org/pandas-docs/stable/getting_started/dsintro.html#dataframe)

```
In [1]: import pandas as pd

housing = pd.read_csv("File/housing.csv")

#print(housing)
#housing.info()
#print(housing.iloc[20:35, ].to_string())
```

```
In [2]: import pandas as pd

excel = pd.read_excel("File/Data_Excel.xlsx")

# print(excel)
#excel.info()
#print(excel.iloc[10:20, 0:2].to_string())
```

دریافت اطلاعات از سازمان بورس
برای مثال صنایع مس ایران (فملی) در بازار اول (تابلوی اصلی) بورس

TseTmc (Tehran Securities Exchange Technology Management Co): [Home-Page \(http://www.tsetmc.com/\)](http://www.tsetmc.com/)

Famely page's = [Famely Page \(http://www.tsetmc.com/Loader.aspx?ParTree=151311&i=35425587644337450#\)](http://www.tsetmc.com/Loader.aspx?ParTree=151311&i=35425587644337450#)

```
In [16]: import pandas as pd
url_famely = "http://www.tsetmc.com/tsev2/data/Export-txt.aspx?t=i&a=1&b=0&i=35425587644337450"
data_famely = pd.read_csv(url_famely)
print(data_famely)
```

<TICKER>	<DTYYYYMMDD>	<FIRST>	<HIGH>	<LOW>	<CLOSE>	\
0	S*I..N..C..Ind.	20200426	11164.0	11164.0	11164.0	11164.0
1	S*I..N..C..Ind.	20200425	10639.0	10648.0	10278.0	10633.0
2	S*I..N..C..Ind.	20200422	10250.0	10250.0	9800.0	10141.0
3	S*I..N..C..Ind.	20200421	9963.0	10279.0	9795.0	10052.0
4	S*I..N..C..Ind.	20200420	9500.0	9999.0	9250.0	9839.0
...
2902	S*I..N..C..Ind.	20070210	4008.0	4008.0	4008.0	4008.0
2903	S*I..N..C..Ind.	20070207	3930.0	3930.0	3930.0	3930.0
2904	S*I..N..C..Ind.	20070206	3853.0	3853.0	3853.0	3853.0
2905	S*I..N..C..Ind.	20070205	3778.0	3778.0	3778.0	3778.0
2906	S*I..N..C..Ind.	20070204	4000.0	4000.0	3600.0	3704.0

	<VALUE>	<VOL>	<OPENINT>	<PER>	<OPEN>	<LAST>
0	282509552584	25305406	1415	D	10633.0	11164.0
1	1650627235222	155239880	18749	D	10141.0	10648.0
2	1586848417818	156478248	19504	D	10052.0	10170.0
3	836027130920	83170132	13661	D	9839.0	10129.0
4	1072967189616	109046931	16587	D	9534.0	9888.0
...
2902	162907276224	40645528	4984	D	3930.0	4008.0
2903	122171316570	31086849	4247	D	3853.0	3930.0
2904	103453562449	26850133	4118	D	3778.0	3853.0
2905	40902501888	10826496	1709	D	3704.0	3778.0
2906	640374172370	172898994	2489	D	4000.0	3704.0

[2907 rows x 12 columns]

رشته، دیکشنری و json

```

In [23]: import json

data_dic = {
    "name": "Mostafa",
    "age": 24,
    "Married": False,
    "Children": None,
    "Student": True,
}

data_json = '''
{
    "name": "MMM",
    "age": 29,
    "Married": false,
    "Children": null,
    "Student": true
}
'''

dict_json = json.dumps(data_dic)
print(dict_json)

print(data_dic["name"])

with open("File/dict2json.json", "w") as da:
    json.dump(dict_json, da)

json_data = json.loads(data_json)
print(json_data)

with open("File/dict2json.json", "r") as da:
    j2d = json.load(da)

print(j2d)

{"name": "Mostafa", "age": 24, "Married": false, "Children": null, "Student": true}
Mostafa
{'name': 'MMM', 'age': 29, 'Married': False, 'Children': None, 'Student': True}
{"name": "Mostafa", "age": 24, "Married": false, "Children": null, "Student": true}

```

کار با api ها

api = Application Programming Interface

```
In [28]: import requests

c19_response = requests.get("https://api.covid19api.com/country/iran?from=2020-04-01T00:00:00Z&to=2020-04-24T00:00:00Z")

print(c19_response)
# print(c19_response.json())

a = c19_response.json()[0]['Confirmed']
b = c19_response.json()[0]['Deaths']
print("In first day Confirmed is {} and Deaths is {}".format(a, b))
print("*"*21, "####", "*" * 21)

iter_co = len(c19_response.json())

for i in range(0, iter_co):
    print("In day {} Confirmed by Col9-test is {} and Deaths is {}".format(i+1,
c19_response.json()[i]['Confirmed'], c19_response.json()[i]['Deaths']))

<Response [200]>
In first day Confirmed is 47593 and Deaths is 3036
***** #### *****
In day 1 Confirmed by Col9-test is 47593 and Deaths is 3036
In day 2 Confirmed by Col9-test is 50468 and Deaths is 3160
In day 3 Confirmed by Col9-test is 53183 and Deaths is 3294
In day 4 Confirmed by Col9-test is 55743 and Deaths is 3452
In day 5 Confirmed by Col9-test is 58226 and Deaths is 3603
In day 6 Confirmed by Col9-test is 60500 and Deaths is 3739
In day 7 Confirmed by Col9-test is 62589 and Deaths is 3872
In day 8 Confirmed by Col9-test is 64586 and Deaths is 3993
In day 9 Confirmed by Col9-test is 66220 and Deaths is 4110
In day 10 Confirmed by Col9-test is 68192 and Deaths is 4232
In day 11 Confirmed by Col9-test is 70029 and Deaths is 4357
In day 12 Confirmed by Col9-test is 71686 and Deaths is 4474
In day 13 Confirmed by Col9-test is 73303 and Deaths is 4585
In day 14 Confirmed by Col9-test is 74877 and Deaths is 4683
In day 15 Confirmed by Col9-test is 76389 and Deaths is 4777
In day 16 Confirmed by Col9-test is 77995 and Deaths is 4869
In day 17 Confirmed by Col9-test is 79494 and Deaths is 4958
In day 18 Confirmed by Col9-test is 80868 and Deaths is 5031
In day 19 Confirmed by Col9-test is 82211 and Deaths is 5118
In day 20 Confirmed by Col9-test is 83505 and Deaths is 5209
In day 21 Confirmed by Col9-test is 84802 and Deaths is 5297
In day 22 Confirmed by Col9-test is 85996 and Deaths is 5391
In day 23 Confirmed by Col9-test is 87026 and Deaths is 5481
In day 24 Confirmed by Col9-test is 88194 and Deaths is 5574
```

```
In [39]: import requests

bit_response = requests.get("https://api.coinbase.com/v2/prices/buy?currency=USD")

print(bit_response)
a = bit_response.json()
b = bit_response.json()['data']['amount']
print(a, "\n", b)

print("Today per BitCoins amount in USD is {} dollars".format(b))

<Response [200]>
{'data': {'base': 'BTC', 'currency': 'USD', 'amount': '8726.74'}}
8726.74
Today per BitCoins amount in USD is 8726.74 dollars
```

```
In [21]: import requests
import re
from bs4 import BeautifulSoup

felan = input('chi mikhay ?')
req = requests.get('https://www.digikala.com/search/?q={}'.format(felan))
soup = BeautifulSoup(req.text , "html.parser")

# title = soup.find_all('div' , attrs={'class':'c-product-box__title'})
# cost = soup.find_all('div', attrs={'class':'c-price__value-wrapper'})
# len_page = int(len(title))

# title_val = re.sub(r'\s+', ' ', title[2].text)
# cost_val = re.sub(r'\s+', ' ', cost[2].text)

# print("Title : {} and cost is {}".format(title_val, cost_val))

# with open("File/Resul_di.txt", "w") as res:
#     for i in range(0, len_page - 1 ):
#         res.write("Title: {} -" .format(re.sub(r'\s+', ' ', title[i].text)))
#         res.write("- Cost: {} \n".format(re.sub(r'\s+', ' ', cost[i].text)))

content = soup.find_all('div', attrs={'class':'c-product-box__content'})

for i in range(len(content)):

    print(re.sub(r'\s+', ' ', content[i].find('div', attrs={'class': 'c-product-box__title'}).text), " - ", re.sub(r'\s+', ' ', content[i].find('div', attrs={'class': 'c-product-box__row c-product-box__row--price'}).text))
```

سبد سیب زمینی و پیاز تلم مدل 01-40-022 - ۹۸,۲۸۰ تومان
 جعبه دستمال کاغذی تلم مدل 77088 - ۸۵,۰۰۰ تومان
 سرویس غذاخوری 98 پارچه چینی زرین ایران سری کواترو مدل مون بلان درجه یک - ۱,۳۰۰ تومان
 سرویس غذاخوری 102 پارچه چینی زرین ایران سری ایتالیا اف مدل مون بلان درجه یک - ۱,۱۸۱,۰۰۰ تومان
 سرویس غذاخوری 27 پارچه چینی زرین ایران سری کواترو مدل مون بلان درجه عالی - ۴۱,۰۰۰ تومان
 تومان ۸۸,۵۰۰ - K1381-2 تخته گوشت تلم مدل
 تومان ۵۳,۰۰۰ - K1381 پایه رول دستمال تلم مدل
 تومان ۹۷,۱۰۰ - Flan قالب کیک گواردینی سری کرامیا مدل
 ست جای ادویه 7 پارچه تلم مدل 17-1381 - ۱۱۵,۰۰۰ تومان
 تومان ۷۷,۵۰۰ - K1381-10 تخته گوشت تلم مدل
 کد 02 - ۷۸,۰۰۰ تومان Oliver جاقاشقی تلم مدل
 تومان ۴,۴۰۰,۰۰۰ - Felen 2-3 چادر 2 نفره هاسکی مدل
 حجم 300 میلی لیتر - ۲۵,۰۰۰ %۴۰ Anti Pollution مدل Protective شامپو گلان سری
 ۵,۰۰۰ تومان حذف ۲ عدد ۳ عدد ۴ عدد ۵ عدد ۶ عدد ۷ عدد ۸ عدد ۹ عدد ۱۰ عدد ۱۱ عدد ۱۲ عدد
 ۳ عدد ۱۴ عدد ۱۵ عدد ۱۶ عدد ۱۷ عدد ۱۸ عدد
 شامپو بدن گلان کد 002 حجم 750 میلی لیتر - ۱۹,۸۷۰ تومان حذف ۲ عدد ۳ عدد ۴ عدد
 ۵ عدد ۶ عدد ۷ عدد ۸ عدد ۹ عدد ۱۰ عدد ۱۱ عدد
 تومان ۱۰,۴۰۰ - T02 پایه نگهدارنده گوشی موبایل تلم مدل
 سرویس غذاخوری 102 پارچه چینی زرین ایران سری ایتالیا اف مدل مون بلان درجه عالی - ۱,۲۹۰,۰۰۰ تومان
 سرویس غذاخوری 27 پارچه چینی زرین ایران سری کواترو مدل مون بلان درجه عالی - ۶۰,۰۰۰ تومان
 حجم 50 میلی لیتر - ناموجود Jeu d'Amour Félin ادو پرفیوم زنانه کنزو مدل
 سرویس غذاخوری 98 پارچه چینی زرین ایران سری کواترو مدل مون بلان درجه عالی - ۱,۶۰۰,۰۰۰ تومان
 شعله پخش کن تلم مدل 111233332 - ۲۳,۵۰۰ تومان
 زیر قابلمه ای تلم طرح توت فرنگی - ۱۹,۹۰۰ تومان
 سرویس غذاخوری 27 پارچه چینی زرین ایران سری کواترو مدل مون بلان درجه یک - ۳۸,۰۰۰ تومان
 تومان ۸۳,۰۰۰ - K1412-11 جاقاشقی تلم مدل
 تومان ۵۹,۰۰۰ - K13811 پایه رول دستمال تلم مدل
 کد 01 - ناموجود Melan شلوارک زنانه ساروک مدل
 تومان ۹۹,۰۰۰ - M تخته گوشت تلم مدل گرد
 کتاب سوتله دلان 6 اثر سیدعباس میرحسینی - ۱۰,۰۰۰ تومان
 تومان ۱۸۹,۰۰۰ - ABS1100 میز اتو تلم مدل
 تومان ۱۷۵,۰۰۰ - ABS1104 میز اتو تلم مدل
 ست جای ادویه 2 پارچه تلم مدل - ۳۱,۳۰۰ تومان
 آویز لباس تلم مدل سیب - ۲۷,۰۰۰ تومان
 کتاب سوتله دلان 2 اثر سید عباس میرحسینی - ۳۰,۰۰۰ تومان
 ناموجود FELA جا کفشی مدل
 حجم 300 میلی لیتر - ۲۵,۰۰۰ تومان Anti Pollution مدل Protective شامپو گلان سری
 حذف ۲ عدد ۳ عدد ۴ عدد ۵ عدد ۶ عدد ۷ عدد ۸ عدد ۹ عدد ۱۰ عدد ۱۱ عدد ۱۲ عدد ۱۳ عدد ۱۴ عدد
 ۱۵ عدد ۱۶ عدد ۱۷ عدد ۱۸ عدد ۱۹ عدد ۲۰ عدد ۲۱ عدد ۲۲ عدد ۲۳ عدد ۲۴ عدد
 تومان ۱۹۸,۰۰۰ - BR1412-03 بند رخت تلم مدل
 تومان ۱۱۵,۰۰۰ - circle L1381-18 تخته گوشت تلم مدل

Any Question:

- [Email \(mailto:mkarimi21@hotmail.com\)](mailto:mkarimi21@hotmail.com), [LinkedIn \(https://www.linkedin.com/in/mkarimi21/\)](https://www.linkedin.com/in/mkarimi21/), [Telegram \(https://telegram.me/mkarimi21\)](https://telegram.me/mkarimi21).



(<https://mr-karimi.ir/>)

Mostafa Karimi

```
In [22]: !jupyter nbconvert --to html MKarimi_Scrap.ipynb
```

```
[NbConvertApp] Converting notebook MKarimi_Scrap.ipynb to html  
[NbConvertApp] Writing 309894 bytes to MKarimi_Scrap.html
```

```
In [ ]:
```