

chapter 1:read files

In [19]:

```

1 #1.1 read csv file
2 import pandas as pd
3 df_csv=pd.read_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pokemon_data
4                      ,usecols=['#','Name','Type 1','HP','Attack','Defense'])
5 df_csv.columns # show columns
6 df_csv.head(4)
7 df_csv.tail(3)
8 print(df_csv[['Name','HP','#']][0:4])

```

		Name	HP	#
0		Bulbasaur	45	1
1		Ivysaur	60	2
2		Venusaur	80	3
3	VenusaurMega	Venusaur	80	3
4		Charmander	39	4
..	
795		Diancie	50	719
796	DiancieMega	Diancie	50	719
797	HoopaHoopa	Confined	80	720
798	HoopaHoopa	Unbound	80	720
799		Volcanion	80	721

[800 rows x 3 columns]

In [6]:

```

1 # 2.read_txt
2 import pandas as pd
3 df_txt=pd.read_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pokemon_data
4                      ,usecols=['#','Name','Type 1','HP','Attack','Defense'])
5 #                      ,delimiter='\t')
6 #df_txt=pd.read_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pokemon_dat
7 print(df_txt.head(3))
8
9 """dataframe.iloc[row, column]"""
10 print(df_csv.iloc[0:-1:2,[0,1,2,3]]) # print all even rows, with spe cols\n"
11 print(df_csv.iloc[:,[0,1,2,4]]) #print all rows

```

	#	Name	Type 1	HP	Attack	Defense
0	1	Bulbasaur	Grass	45	49	49
1	2	Ivysaur	Grass	60	62	63
2	3	Venusaur	Grass	80	82	83

```
In [12]: 1 # 3.read_excel
2
3 """loc is more useful in real world as it specify the extract clo name
4         loc +range(start,end,step)
5 """
6 df_xlsx=pd.read_excel(r'C:\Users\fengs\Desktop\py_project\csv_file\pokemon_d
7                        ,usecols=['#', 'Name', 'Type 1', 'HP', 'Attack', 'Defense'])
8 df_xlsx.loc[range(0,len(df_xlsx),2),['#',"Name","HP"]].head(8)
9 """combine with head() method"""
```

```
Out[12]:
```

	#	Name	HP
0	1	Bulbasaur	45
2	3	Venusaur	80
4	4	Charmander	39
6	6	Charizard	78
8	6	CharizardMega Charizard Y	78
10	8	Wartortle	59
12	9	BlastoiseMega Blastoise	79
14	11	Metapod	50

chapter 2: sort/describe data

```
In [32]: 1 """ df_csv.sort_values(['xx','yy'],ascending=[1,0])
2           1 is ascending, 0 is descending
3           DataFrame.sort_values(by, axis=0, ascending=True, inplace=False
4                                   , kind='quicksort', na_position='last')
5
6                                   #na_postition is null value, default last
7 """
8 #df_csv.sort_values(by=['Name', '#'],ascending=False) #desc
9 df_csv.sort_values(by=['Name', '#'],ascending=[1,0])
10
11 #cannot use head/tail(),need 2 step
12 """IndentationError: unexpected indent: if a space tab in front of df_csv """
```

```
Out[32]: 'IndentationError: unexpected indent: if a space tab in front of df_csv '
```

In [33]:

1 df_csv.sort_values(by=['Name', '#'],ascending=[1,0])

Out[33]:

	#	Name	Type 1	HP	Attack	Defense
510	460	Abomasnow	Grass	90	92	75
511	460	AbomasnowMega Abomasnow	Grass	90	132	105
68	63	Abra	Psychic	25	20	15
392	359	Absol	Dark	65	130	60
393	359	AbsolMega Absol	Dark	65	150	60
...
632	571	Zoroark	Dark	60	105	60
631	570	Zorua	Dark	40	65	40
46	41	Zubat	Poison	40	45	35
695	634	Zweilous	Dark	72	85	70
794	718	Zygarde50% Forme	Dragon	108	100	121

800 rows × 6 columns

chatper 3:change data

In [39]:

1 """3.1 add a column"""
2
3 # df_csv['Total']=df_csv['Attack']+df_csv['Defense']+df_csv["HP"]
4
5 """3.2 DROP A columnm
6 DataFrame.drop(labels=None, axis=0, index=None,
7 columns=None, level=None, inplace=False, errors='raise')[source
8 nplace = True：不创建新的对象，直接对原始对象进行修改；
9 • inplace = False：对数据进行修改，创建并返回新的对象承载其修改结果。
10 默认是False，即创建新的对象进行修改，原对象不变，和深复制和浅复制有些类似。"""
11 df_csv.drop(columns=['Total'],)
12 df_csv.head(4)
13

Out[39]:

	#	Name	Type 1	HP	Attack	Defense	Total
0	1	Bulbasaur	Grass	45	49	49	143
1	2	Ivysaur	Grass	60	62	63	185
2	3	Venusaur	Grass	80	82	83	245
3	3	VenusaurMega Venusaur	Grass	80	100	123	303

In [41]: 1 df_csv.info

Out[41]: <bound method DataFrame.info of # Name Type 1 HP A
 ttack Defense Total
 0 1 Bulbasaur Grass 45 49 49 143
 1 2 Ivysaur Grass 60 62 63 185
 2 3 Venusaur Grass 80 82 83 245
 3 3 VenusaurMega Venusaur Grass 80 100 123 303
 4 4 Charmander Fire 39 52 43 134

 795 719 Diancie Rock 50 100 150 300
 796 719 DiancieMega Diancie Rock 50 160 110 320
 797 720 HoopaHoopa Confined Psychic 80 110 60 250
 798 720 HoopaHoopa Unbound Psychic 80 160 60 300
 799 721 Volcanion Fire 80 110 120 310

[800 rows x 7 columns]>

In [44]: 1 import numpy as np

In [46]: 1 df_csv.describe()

Out[46]:

	#	HP	Attack	Defense	Total
count	800.000000	800.000000	800.000000	800.000000	800.000000
mean	362.813750	69.258750	79.001250	73.842500	222.102500
std	208.343798	25.534669	32.457366	31.183501	68.173834
min	1.000000	1.000000	5.000000	5.000000	55.000000
25%	184.750000	50.000000	55.000000	50.000000	170.000000
50%	364.500000	65.000000	75.000000	70.000000	222.000000
75%	539.250000	80.000000	100.000000	90.000000	269.250000
max	721.000000	255.000000	190.000000	230.000000	440.000000

chapter 4: save file

In [53]:

```

1 """ 4.1 normal way, but can use chain method
2     df_csv_2=df_csv.head(5)
3     df_csv_2.to_excel('xxxxx')
4 """
5 df_csv.head(5).to_excel(r'C:\Users\fengs\Desktop\py_project\csv_file\csv_2.x

```

```
In [58]: 1  """
2  4.2:to txt\csv file
3  """
4  # df_txt.tail(4)
5  # df_csv.tail(5).to_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pk_text
6  """ use this for txt file, default x,y,z, """
7  df_csv.tail(5).to_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pk_text_2
8
```

chapter 5 filtering

```
In [ ]: 1  """5.1:conditions with loc"""
```

```
In [6]: 1  import pandas as pd
2  import numpy as py
3  df_csv=pd.read_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pokemon_data
4  df_csv.columns
```

```
Out[6]: Index(['#', 'Name', 'Type 1', 'Type 2', 'HP', 'Attack', 'Defense', 'Sp. Atk',
              'Sp. Def', 'Speed', 'Generation', 'Legendary'],
              dtype='object')
```

```
In [9]: 1  ""df.info=!df.info""
2  df_csv.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 800 entries, 0 to 799
Data columns (total 12 columns):
#   Column          Non-Null Count  Dtype
---  -
0   #               800 non-null   int64
1   Name            800 non-null   object
2   Type 1          800 non-null   object
3   Type 2          414 non-null   object
4   HP              800 non-null   int64
5   Attack          800 non-null   int64
6   Defense         800 non-null   int64
7   Sp. Atk         800 non-null   int64
8   Sp. Def         800 non-null   int64
9   Speed           800 non-null   int64
10  Generation       800 non-null   int64
11  Legendary        800 non-null   bool
dtypes: bool(1), int64(8), object(3)
memory usage: 69.7+ KB
```

In [46]:

```
1 """fields: var['xxxxxxx']
2     functions: var.fuc[yyyyy] use a outer[]
3 """
4 df_csv.loc[df_csv['Type 1']=='Grass']
```

Out[46]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Leg
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1	
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1	
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1	
3	3	VenusaurMega Venusaur	Grass	Poison	80	100	123	122	120	80	1	
48	43	Oddish	Grass	Poison	45	50	55	75	65	30	1	
...
718	650	Chespin	Grass	NaN	56	61	65	48	45	38	6	
719	651	Quilladin	Grass	NaN	61	78	95	56	58	57	6	
720	652	Chesnaught	Grass	Fighting	88	107	122	74	75	64	6	
740	672	Skiddo	Grass	NaN	66	65	48	62	57	52	6	
741	673	Gogoat	Grass	NaN	123	100	62	97	81	68	6	

70 rows × 12 columns



In [16]:

```
1 """
2     and :&
3     or:|
4     df.loc[()] &() &]
5 """
6 df_csv.loc[(df_csv['Type 1']=='Grass') & (df_csv['#']<50)]
```

Out[16]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Legend
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1	Fal
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1	Fal
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1	Fal
3	3	VenusaurMega Venusaur	Grass	Poison	80	100	123	122	120	80	1	Fal
48	43	Oddish	Grass	Poison	45	50	55	75	65	30	1	Fal
49	44	Gloom	Grass	Poison	60	65	70	85	75	40	1	Fal
50	45	Vileplume	Grass	Poison	75	80	85	110	90	50	1	Fal



```
In [25]: 1 df_txt=pd.read_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pokemon_data
2
```

```
In [20]: 1 """sep='\t useful for to_csv ( )
2         namely file exporting
3         """
4
5 df_txt_2=pd.read_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pokemon_da
6                     ,sep='\t'
7                     )
8
```

```
Out[20]:
```

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Legendary
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1	False
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1	False
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1	False

```
In [27]: 1 """sep='\t' MAKE XXX,YYY,ZZZZ to standard
2
3         """
4 df_csv.loc[(df_csv['Type 1']=='Grass') & (df_csv['#']<50)]\
5           .to_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\new_text.txt',sep=''
```

In [44]:

```
1  """
2      str.contains()=sql like%xxx%
3      Series.str.contains(pat, case=True, flags=0, na=nan, regex=True)
4      not contains add a ~
5  """
6  #df_csv.loc[(df_csv['Name'].str.contains('Mega')) & (df_csv["#"]<20)]
7  #note contaions#
8  df_csv.loc[(~df_csv['Name'].str.contains('Mega|er')) & (df_csv["#"]<10)]
9
10 # """
11 # s2:
12 #     searchfor = ['og', 'at']
13 #     s[s.str.contains('|'.join(searchfor))]
14 # """
```

Out[44]:

	#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	Sp. Def	Speed	Generation	Legendary
0	1	Bulbasaur	Grass	Poison	45	49	49	65	65	45	1	False
1	2	Ivysaur	Grass	Poison	60	62	63	80	80	60	1	False
2	3	Venusaur	Grass	Poison	80	82	83	100	100	80	1	False
5	5	Charmeleon	Fire	NaN	58	64	58	80	65	80	1	False
6	6	Charizard	Fire	Flying	78	84	78	109	85	100	1	False
9	7	Squirtle	Water	NaN	44	48	65	50	64	43	1	False
10	8	Wartortle	Water	NaN	59	63	80	65	80	58	1	False
11	9	Blastoise	Water	NaN	79	83	100	85	105	78	1	False

""where clause "" df_csv.loc[df_csv[""]]

5.2 conditional change

In []:

```
1
```

chapter 6 AG functions

DataFrame.groupby(by=None, axis=0, level=None, as_index=True, sort=True, group_keys=True, squeeze=False, **kwargs)


```
In [51]: 1 import pandas as pd
          2 import numpy as np
          3 #groupby() function
          4 df_csv['count']=1
          5 df_csv.groupby(["Type 1"]).count()['count']
```

```
Out[51]: Type 1
Bug      69
Dark     31
Dragon   32
Electric 44
Fairy    17
Fighting 27
Fire     52
Flying    4
Ghost    32
Grass    70
Ground   32
Ice      24
Normal   98
Poison   28
Psychic  57
Rock     44
Steel    27
Water    112
Name: count, dtype: int64
```

```
In [52]: 1 df_csv['count']=1
          2 df_csv.groupby(["Type 1", 'Type 2']).count()['count']
```

```
Out[52]: Type 1  Type 2
Bug      Electric    2
          Fighting    2
          Fire        2
          Flying     14
          Ghost       1
          ..
Water    Ice         3
          Poison      3
          Psychic     5
          Rock        4
          Steel       1
Name: count, Length: 136, dtype: int64
```

6.2 if large amounts of data

chunksize=5 means each time pass 5 rows records to process

```
In [53]: 1 for df in pd.read_csv(r'C:\Users\fengs\Desktop\py_project\csv_file\pokemon_d
2           ,chunksize=5):
3           print(df)
```

#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	\
0	1	Bulbasaur	Grass	Poison	45	49	49	65
1	2	Ivysaur	Grass	Poison	60	62	63	80
2	3	Venusaur	Grass	Poison	80	82	83	100
3	3	VenusaurMega Venusaur	Grass	Poison	80	100	123	122
4	4	Charmander	Fire	NaN	39	52	43	60
Sp.	Def	Speed	Generation	Legendary				
0	65	45	1	False				
1	80	60	1	False				
2	100	80	1	False				
3	120	80	1	False				
4	50	65	1	False				
#	Name	Type 1	Type 2	HP	Attack	Defense	Sp. Atk	\
5	5	Charmeleon	Fire	NaN	58	64	58	80
6	6	Charizard	Fire	Flying	78	84	78	109
7	6	CharizardMega Charizard X	Fire	Dragon	78	130	111	130
8	6	CharizardMega Charizard Y	Fire	Flying	78	104	78	159
9	7	Sandslash	Ice	NaN	44	46	65	50

```
In [ ]: 1
```