Sheet

Python-Pandas cheat sheet: 30 functions-methods

```
import pandas as pd
```

```
# 1. Loading the data from a csv file
df = pd.read_csv("Airlines.csv")
```

```
# 2. Shape of a dataframe
df.shape
```

```
(539383, 9)
```

```
# 3. Head and Tail of the data frame
df.head(n=10)
df.tail(n=10)
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay
539373	539374	B6	480	LAX	BOS	5	1435	320	1
539374	539375	DL	2354	LAX	ATL	5	1435	255	0
539375	539376	FL	58	LAX	ATL	5	1435	250	0
539376	539377	B6	717	JFK	SJU	5	1439	220	1
539377	539378	B6	739	JFK	PSE	5	1439	223	1
539378	539379	CO	178	OGG	SNA	5	1439	326	0
539379	539380	FL	398	SEA	ATL	5	1439	305	0
539380	539381	FL	609	SFO	MKE	5	1439	255	0
539381	539382	UA	78	HNL	SFO	5	1439	313	1
539382	539383	US	1442	LAX	PHL	5	1439	301	1

```
# 4. data types of the columns
df.dtypes
```

```
# 5. Getting column names
df.columns.tolist()
```

```
['id',
 'Airline',
 'Flight',
 'AirportFrom',
 'AirportTo',
 'DayOfWeek',
 'Time',
 'Length',
 'Delay']
```

```
# 6. Summary stats
df.describe()
```

	id	Flight	DayOfWeek	Time	Length	Delay
count	539383.000000	539383.000000	539383.000000	539383.000000	539383.000000	539383.000000
mean	269692.000000	2427.928630	3.929668	802.728963	132.202007	0.445442
std	155706.604461	2067.429837	1.914664	278.045911	70.117016	0.497015
min	1.000000	1.000000	1.000000	10.000000	0.000000	0.000000
25%	134846.500000	712.000000	2.000000	565.000000	81.000000	0.000000
50%	269692.000000	1809.000000	4.000000	795.000000	115.000000	0.000000
75%	404537.500000	3745.000000	5.000000	1035.000000	162.000000	1.000000
max	539383.000000	7814.000000	7.000000	1439.000000	655.000000	1.000000

```
# 7. Checking NA values in columns
df.isna().sum()
```

```
# 8. Selecting columns with data type as object
df.select_dtypes(include = 'object').columns
```

```
Index(['Airline', 'AirportFrom', 'AirportTo'], dtype='object')
```

```
# 9. Getting value counts from the columns
df['Airline'].value_counts(ascending=True)
```

```
# 10. Getting unique names of values in a column
df['Airline'].unique()
```

```
# 11. Select a few columns from df

df[['id', 'Airline', 'Flight']]
```

	id	Airline	Flight
0	1	CO	269
1	2	US	1558
2	3	AA	2400
3	4	AA	2466
4	5	AS	108
...	...	...	...
539378	539379	CO	178
539379	539380	FL	398
539380	539381	FL	609
539381	539382	UA	78
539382	539383	US	1442

539383 rows × 3 columns

```
# 12. Select a few rows
df.iloc[:10,]
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay
0	1	CO	269	SFO	IAH	3	15	205	1
1	2	US	1558	PHX	CLT	3	15	222	1
2	3	AA	2400	LAX	DFW	3	20	165	1
3	4	AA	2466	SFO	DFW	3	20	195	1
4	5	AS	108	ANC	SEA	3	30	202	0
5	6	CO	1094	LAX	IAH	3	30	181	1
6	7	DL	1768	LAX	MSP	3	30	220	0
7	8	DL	2722	PHX	DTW	3	30	228	0
8	9	DL	2606	SFO	MSP	3	35	216	1
9	10	AA	2538	LAS	ORD	3	40	200	1

```
# 13. Select a few rows and columns
df.loc[:5, ['id', 'Airline', 'Flight']]
```

	id	Airline	Flight
0	1	CO	269
1	2	US	1558
2	3	AA	2400
3	4	AA	2466
4	5	AS	108
5	6	CO	1094

```
# 14. Filter the data using a column
df[df['Airline'] == 'US']
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay
1	2	US	1558	PHX	CLT	3	15	222	1
15	16	US	498	DEN	CLT	3	55	179	0
24	25	US	122	ANC	PHX	3	113	327	1
31	32	US	1011	EWR	CLT	3	300	111	0
32	33	US	1983	BOS	CLT	3	300	135	0
...	...	...	...	...	...	...	...	...	...
539353	539354	US	31	OGG	PHX	5	1410	344	0
539365	539366	US	119	KOA	PHX	5	1425	349	1
539366	539367	US	258	PHX	PHL	5	1425	254	0
539369	539370	US	125	HNL	PHX	5	1430	362	0
539382	539383	US	1442	LAX	PHL	5	1439	301	1

34500 rows × 9 columns

```
# 15. Filter the data using multiple columns
df[(df['Airline'] == 'US') & (df['AirportFrom'] == 'PHX') & (df['DayOfWeek'] == 1)]
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay
85191	85192	US	1558	PHX	CLT	1	15	222	1
86311	86312	US	680	PHX	CLT	1	390	225	0
87775	87776	US	1540	PHX	CLT	1	465	216	0
88067	88068	US	254	PHX	PHL	1	480	263	0
88069	88070	US	540	PHX	DFW	1	480	138	0
...	...	...	...	...	...	...	...	...	...
468485	468486	US	83	PHX	SEA	1	1422	182	0
468496	468497	US	258	PHX	PHL	1	1425	254	1
468497	468498	US	417	PHX	SFO	1	1425	120	1
468498	468499	US	640	PHX	ONT	1	1426	70	1
468503	468504	US	194	PHX	SAN	1	1431	68	0

693 rows × 9 columns

```
# 16. Filter data using OR conditions
df[(df['Airline'] == 'US') | (df['AirportFrom'] == 'PHX')]
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay
1	2	US	1558	PHX	CLT	3	15	222	1
7	8	DL	2722	PHX	DTW	3	30	228	0
11	12	DL	1646	PHX	ATL	3	50	212	1
15	16	US	498	DEN	CLT	3	55	179	0
24	25	US	122	ANC	PHX	3	113	327	1
...	...	...	...	...	...	...	...	...	...
539357	539358	CO	434	PHX	EWR	5	1420	259	1
539365	539366	US	119	KOA	PHX	5	1425	349	1
539366	539367	US	258	PHX	PHL	5	1425	254	0
539369	539370	US	125	HNL	PHX	5	1430	362	0
539382	539383	US	1442	LAX	PHL	5	1439	301	1

44815 rows × 9 columns

```
# 17. Filter data using a list
airline_list = ['DL','US']

df[df['Airline'].isin(airline_list)]
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay
1	2	US	1558	PHX	CLT	3	15	222	1
6	7	DL	1768	LAX	MSP	3	30	220	0
7	8	DL	2722	PHX	DTW	3	30	228	0
8	9	DL	2606	SFO	MSP	3	35	216	1
11	12	DL	1646	PHX	ATL	3	50	212	1
...	...	...	...	...	...	...	...	...	...
539365	539366	US	119	KOA	PHX	5	1425	349	1
539366	539367	US	258	PHX	PHL	5	1425	254	0
539369	539370	US	125	HNL	PHX	5	1430	362	0
539374	539375	DL	2354	LAX	ATL	5	1435	255	0
539382	539383	US	1442	LAX	PHL	5	1439	301	1

95440 rows × 9 columns

```
# 18. Filter data not in list
airline_list = ['DL','US']

df[~df['Airline'].isin(airline_list)]
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay
0	1	CO	269	SFO	IAH	3	15	205	1
2	3	AA	2400	LAX	DFW	3	20	165	1
3	4	AA	2466	SFO	DFW	3	20	195	1
4	5	AS	108	ANC	SEA	3	30	202	0
5	6	CO	1094	LAX	IAH	3	30	181	1
...	...	...	...	...	...	...	...	...	...
539377	539378	B6	739	JFK	PSE	5	1439	223	1
539378	539379	CO	178	OGG	SNA	5	1439	326	0
539379	539380	FL	398	SEA	ATL	5	1439	305	0
539380	539381	FL	609	SFO	MKE	5	1439	255	0
539381	539382	UA	78	HNL	SFO	5	1439	313	1

443943 rows × 9 columns

```
# 19. Sort the data
df.sort_values(by='Airline',ascending=False)
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay
319156	319157	YV	7250	ORD	SAT	7	776	176	0
144301	144302	YV	2651	PHX	BFL	4	660	93	0
345479	345480	YV	1040	HNL	ITO	1	1195	49	1
345480	345481	YV	2680	CLT	CHS	1	1195	63	1
345481	345482	YV	7257	ORD	CHS	1	1195	119	0
...	...	...	...	...	...	...	...	...	...
45169	45170	9E	4349	MEM	ICT	5	825	98	0
45168	45169	9E	4147	DTW	BUF	5	825	75	1
45167	45168	9E	3823	MSP	PIT	5	825	126	0
303169	303170	9E	4250	IND	JFK	6	780	152	1
498329	498330	9E	3788	CLE	MSP	3	914	131	0

539383 rows × 9 columns

```
# 20. Rename a column
df.rename(columns={"Airline": "Airline_Code", "AirportFrom":"Airport_From"})
```

	id	Airline_Code	Flight	Airport_From	AirportTo	DayOfWeek	Time	Length	Delay
0	1	CO	269	SFO	IAH	3	15	205	1
1	2	US	1558	PHX	CLT	3	15	222	1
2	3	AA	2400	LAX	DFW	3	20	165	1
3	4	AA	2466	SFO	DFW	3	20	195	1
4	5	AS	108	ANC	SEA	3	30	202	0
...	...	...	...	...	...	...	...	...	...
539378	539379	CO	178	OGG	SNA	5	1439	326	0
539379	539380	FL	398	SEA	ATL	5	1439	305	0
539380	539381	FL	609	SFO	MKE	5	1439	255	0
539381	539382	UA	78	HNL	SFO	5	1439	313	1
539382	539383	US	1442	LAX	PHL	5	1439	301	1

539383 rows × 9 columns

```
# 21. Summarise using groupby
df.groupby(['Airline','AirportFrom','AirportTo'], as_index=False)['id'].agg('count')
```

	Airline	AirportFrom	AirportTo	id
0	9E	ABE	DTW	85
1	9E	ABR	MSP	2
2	9E	ALB	ATL	41
3	9E	ALB	DTW	90
4	9E	ALB	JFK	31
...	...	...	...	...
6831	YV	SYR	IAD	47
6832	YV	SYR	ORD	35
6833	YV	TEX	PHX	27
6834	YV	TUS	PHX	266
6835	YV	YUM	PHX	188

6836 rows × 4 columns

```
# 22. Summarise and sort
df_summ = df.groupby(['Airline','AirportFrom','AirportTo'], as_index=False)['id'].agg('count')

df_summ.sort_values(by='id', ascending = False)
```

	Airline	AirportFrom	AirportTo	id
3028	HA	OGG	HNL	762
3009	HA	HNL	OGG	731
5361	WN	DAL	HOU	701
5466	WN	HOU	DAL	698
4380	OO	SAN	LAX	573
...	...	...	...	...
2529	EV	ORD	PWM	1
2213	EV	ATL	ORD	1
3611	OH	DTW	RDU	1
2356	EV	DTW	XNA	1
4209	OO	MSP	CVG	1

6836 rows × 4 columns

```
# 23. Summarise for multiple values
df.groupby(['Airline','AirportFrom','AirportTo'])['Time'].agg(['sum','count']).reset_index()
```

	Airline	AirportFrom	AirportTo	sum	count
0	9E	ABE	DTW	58432	85
1	9E	ABR	MSP	820	2
2	9E	ALB	ATL	21734	41
3	9E	ALB	DTW	70685	90
4	9E	ALB	JFK	11295	31
...	...	...	...	...	...
6831	YV	SYR	IAD	36137	47
6832	YV	SYR	ORD	16542	35
6833	YV	TEX	PHX	22818	27
6834	YV	TUS	PHX	215305	266
6835	YV	YUM	PHX	148083	188

6836 rows × 5 columns

```
# 24. Summarise for multiple columns and values
df.groupby(['Airline','AirportFrom','AirportTo']).aggregate({'id':'count','Time':'sum'}).reset_index()
```

	Airline	AirportFrom	AirportTo	id	Time
0	9E	ABE	DTW	85	58432
1	9E	ABR	MSP	2	820
2	9E	ALB	ATL	41	21734
3	9E	ALB	DTW	90	70685
4	9E	ALB	JFK	31	11295
...	...	...	...	...	...
6831	YV	SYR	IAD	47	36137
6832	YV	SYR	ORD	35	16542
6833	YV	TEX	PHX	27	22818
6834	YV	TUS	PHX	266	215305
6835	YV	YUM	PHX	188	148083

6836 rows × 5 columns

```
# 25. Adding a new column
df['Country'] = 'USA'

df.head()
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay	Country
0	1	CO	269	SFO	IAH	3	15	205	1	USA
1	2	US	1558	PHX	CLT	3	15	222	1	USA
2	3	AA	2400	LAX	DFW	3	20	165	1	USA
3	4	AA	2466	SFO	DFW	3	20	195	1	USA
4	5	AS	108	ANC	SEA	3	30	202	0	USA

```
# 26. Adding a column using existing columns
df['CO_SFO'] = (df['Airline'] =='CO') & (df['AirportFrom'] == 'SFO')
df.head()
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay	Country	CO_SFO
0	1	CO	269	SFO	IAH	3	15	205	1	USA	True
1	2	US	1558	PHX	CLT	3	15	222	1	USA	False
2	3	AA	2400	LAX	DFW	3	20	165	1	USA	False
3	4	AA	2466	SFO	DFW	3	20	195	1	USA	False
4	5	AS	108	ANC	SEA	3	30	202	0	USA	False

```
# 27. Dropping a Column
df.drop(['CO_SFO'], axis = 1)
```

	id	Airline	Flight	AirportFrom	AirportTo	DayOfWeek	Time	Length	Delay	Country
0	1	CO	269	SFO	IAH	3	15	205	1	USA
1	2	US	1558	PHX	CLT	3	15	222	1	USA
2	3	AA	2400	LAX	DFW	3	20	165	1	USA
3	4	AA	2466	SFO	DFW	3	20	195	1	USA
4	5	AS	108	ANC	SEA	3	30	202	0	USA
...	...	...	...	...	...	...	...	...	...	...
539378	539379	CO	178	OGG	SNA	5	1439	326	0	USA
539379	539380	FL	398	SEA	ATL	5	1439	305	0	USA
539380	539381	FL	609	SFO	MKE	5	1439	255	0	USA
539381	539382	UA	78	HNL	SFO	5	1439	313	1	USA
539382	539383	US	1442	LAX	PHL	5	1439	301	1	USA

539383 rows × 10 columns

```
# 28. Summarise Using pivot_table

df.pivot_table(index = ['Airline','AirportFrom','AirportTo'],
               values = ['Time'], aggfunc=['sum','count']).reset_index(col_level=1)
```

	Airline	Airpor	
0	9E	ABE	
1	9E	ABR	
2	9E	ALB	
3	9E	ALB	
4	9E	ALB	
...	...	...	
6831	YV	SYR	
6832	YV	SYR	
6833	YV	TEX	
6834	YV	TUS	
6835	YV	YUM	

6836 rows × 5 columns

```
# 29. Pivot data

df.pivot_table(i
```

DayOfWeek	Airline
0	9E
1	AA
2	AS
3	B6
4	CO
5	DL
6	EV
7	F9
8	FL
9	HA
10	MQ
11	OH
12	OO
13	UA
14	US
15	WN
16	XE
17	YV

```
# 30. Summarise
df1 = df.groupby

df1.pivot(index
```

DayOfWeek	Airline
0	9E
1	AA
2	AS
3	B6
4	CO
5	DL
6	EV
7	F9
8	FL
9	HA
10	MQ
11	OH
12	OO
13	UA
14	US
15	WN
16	XE
17	YV