```
import pandas as pd
```

```
#create dataframe from scratch

raw_data = {
    "name": ["Ben","Bam","Boy","Bas","Boss"],
    "age":[33,20,31,43,21],
    "gender":["M","F","M","M","M"]
}
df = pd.DataFrame(raw_data)
```

```
df
```

	name	age	gender
0	Ben	33	M
1	Bam	20	F
2	Boy	31	M
3	Bas	43	M
4	Boss	21	M

```
df["city"] = ['London','London','London','Liverpool','Bristol']
df.shape
```

```
(5, 4)
```

```
# Drop column city
df.drop('city',axis = 1) #drop column
df.drop(2,axis=0) #drop row
```

	name	age	gender	city
0	Ben	33	M	London
1	Bam	20	F	London
3	Bas	43	M	Liverpool
4	Boss	21	M	Bristol

```
# reset index
df = df.reset_index(drop = True)
df
```

	name	age	gender	city
0	Ben	33	M	London
1	Bam	20	F	London
2	Boy	31	M	London
3	Bas	43	M	Liverpool
4	Boss	21	M	Bristol

```
# column name
list(df.columns)
# rename column
df.columns = ['nickname','age','sex','town']
```

```
# create a new series
s1 = pd.Series(['base',15,'F','BKK'],index=['nickname','age','sex','town'])
print(s1)
print(type(s1))
```

```
nickname    base
age           15
sex            F
town         BKK
dtype: object
<class 'pandas.core.series.Series'>
```

```
# append series in record
df = df.append(s1,ignore_index=True)
df
```

	nickname	age	sex	town
0	Ben	33	M	London
1	Bam	20	F	London
2	Boy	31	M	London
3	Bas	43	M	Liverpool
4	Boss	21	M	Bristol
5	base	15	F	BKK

```
# append series with new dimension
s2 = pd.Series(['England','England','England','England','England','Thailand'])
df['Nation'] = s2
df
```

	nickname	age	sex	town	Nation
0	Ben	33	M	London	England
1	Bam	20	F	London	England
2	Boy	31	M	London	England
3	Bas	43	M	Liverpool	England
4	Boss	21	M	Bristol	England
5	base	15	F	BKK	Thailand

```
# write csv file
df.to_csv('mydata.csv')
```

```
# import csv file
df2 = pd.read_csv('Data/demo_export.csv')
```

```
df2
```

	id	movie_name	year	rating	length	genre	score
0	1	The Shawshank Redemption	1994	R	142	Drama	9.3
1	20	Parasite	2019	R	132	Comedy, Drama, Thriller	8.6
2	19	One Flew Over the Cuckoo's Nest	1975	R	133	Drama	8.7
3	35	The Intouchables	2011	R	112	Biography, Comedy, Drama	8.5
4	88	The Hunt	2012	R	115	Drama	8.3

```
#import xlsx file
df3 = pd.read_excel('Data/msa.xlsx')
df3
```

	Unnamed: 0	Paramerter	Result	Area	Date and time	Name	Farm	Country	Consistency	Efficiency	Over Reject	Over Acce
0	NaN	SUD/ASDD	Pass	China	10/2/2022, 10:34:38	Xie jinli	HN-PQ	China	100.0	93.33	0.0667	0.000
1	NaN	Tubular constriction	Pass	China	10/2/2022, 8:25:45	Xie jinli	HN-PQ	China	100.0	96.67	0.0333	0.000
2	NaN	Tubular constriction	Pass	China	10/2/2022, 8:26:23	Xie jinli	HN-PQ	China	100.0	96.67	0.0333	0.000
3	NaN	% Stress test	Pass	China	11/1/2022, 8:30:31	Xie jinli	PQ-HN	China	100.0	100.00	0.0000	0.000
4	NaN	Deform np.	Pass	China	14/1/2022, 8:48:41	Xiejinli	PQ-HN	China	100.0	96.67	0.0333	0.000
5	NaN	Lipid droplet	Pass	China	10/2/2022, 14:25:42	Xie jinli	HN-PQ	China	100.0	96.67	0.0000	0.033
6	NaN	Melanize shell	Pass	China	10/2/2022, 13:03:28	Xie jinli	HN-PQ	China	100.0	90.00	0.0667	0.033

```
#import json file
df4 = pd.read_json('Data/data.json')
df4
```

	ebook	language	amazonRating
0	Getting started with Python	python	4.89
1	Introduction to R	r	4.88
2	SQL for Beginners	sql	4.75

```
penguins = pd.read_csv('Data/penguins.csv')
#preview head
penguins.head()
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	FEMALE

```
#preview tail
penguins.tail()
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
339	Gentoo	Biscoe	NaN	NaN	NaN	NaN	NaN
340	Gentoo	Biscoe	46.8	14.3	215.0	4850.0	FEMALE
341	Gentoo	Biscoe	50.4	15.7	222.0	5750.0	MALE
342	Gentoo	Biscoe	45.2	14.8	212.0	5200.0	FEMALE
343	Gentoo	Biscoe	49.9	16.1	213.0	5400.0	MALE

```
# shape (attribute)
penguins.shape
```

```
(344, 7)
```

```
# information
penguins.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 344 entries, 0 to 343
Data columns (total 7 columns):
 #   Column             Non-Null Count  Dtype  
---  ------             --------------  -----  
 0   species            344 non-null    object 
 1   island             344 non-null    object 
 2   bill_length_mm     342 non-null    float64
 3   bill_depth_mm      342 non-null    float64
 4   flipper_length_mm  342 non-null    float64
```

```
 5   body_mass_g        342 non-null    float64
 6   sex                333 non-null    object
dtypes: float64(4), object(3)
memory usage: 18.9+ KB
```

```
#select column single column
penguins['species']
```

```
penguins.species.head
```

```
<bound method NDFrame.head of 0      Adelie
1      Adelie
2      Adelie
3      Adelie
4      Adelie
        ...
339    Gentoo
340    Gentoo
341    Gentoo
342    Gentoo
343    Gentoo
Name: species, Length: 344, dtype: object>
```

```
penguins[['species','island','sex']].head(3)
```

	species	island	sex
0	Adelie	Torgersen	MALE
1	Adelie	Torgersen	FEMALE
2	Adelie	Torgersen	FEMALE

```
#integer location base indexing(iloc)
penguins.iloc[[0,1,2],[0,2]]
```

	species	bill_length_mm
0	Adelie	39.1
1	Adelie	39.5
2	Adelie	40.3

```
#filters row  by a condition
penguins[penguins['island'] == 'Torgersen']
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	FEMALE
5	Adelie	Torgersen	39.3	20.6	190.0	3650.0	MALE
6	Adelie	Torgersen	38.9	17.8	181.0	3625.0	FEMALE
7	Adelie	Torgersen	39.2	19.6	195.0	4675.0	MALE
8	Adelie	Torgersen	34.1	18.1	193.0	3475.0	NaN
9	Adelie	Torgersen	42.0	20.2	190.0	4250.0	NaN
10	Adelie	Torgersen	37.8	17.1	186.0	3300.0	NaN
11	Adelie	Torgersen	37.8	17.3	180.0	3700.0	NaN
12	Adelie	Torgersen	41.1	17.6	182.0	3200.0	FEMALE
13	Adelie	Torgersen	38.6	21.2	191.0	3800.0	MALE
14	Adelie	Torgersen	34.6	21.1	198.0	4400.0	MALE
15	Adelie	Torgersen	36.6	17.8	185.0	3700.0	FEMALE
16	Adelie	Torgersen	38.7	19.0	195.0	3450.0	FEMALE
17	Adelie	Torgersen	42.5	20.7	197.0	4500.0	MALE
18	Adelie	Torgersen	34.4	18.4	184.0	3325.0	FEMALE
19	Adelie	Torgersen	46.0	21.5	194.0	4200.0	MALE
68	Adelie	Torgersen	35.9	16.6	190.0	3050.0	FEMALE
69	Adelie	Torgersen	41.8	19.4	198.0	4450.0	MALE
70	Adelie	Torgersen	33.5	19.0	190.0	3600.0	FEMALE
71	Adelie	Torgersen	39.7	18.4	190.0	3900.0	MALE
72	Adelie	Torgersen	39.6	17.2	196.0	3550.0	FEMALE
73	Adelie	Torgersen	45.8	18.9	197.0	4150.0	MALE
74	Adelie	Torgersen	35.5	17.5	190.0	3700.0	FEMALE
75	Adelie	Torgersen	42.8	18.5	195.0	4250.0	MALE
76	Adelie	Torgersen	40.9	16.8	191.0	3700.0	FEMALE
77	Adelie	Torgersen	37.2	19.4	184.0	3900.0	MALE
78	Adelie	Torgersen	36.2	16.1	187.0	3550.0	FEMALE
79	Adelie	Torgersen	42.1	19.1	195.0	4000.0	MALE
80	Adelie	Torgersen	34.6	17.2	189.0	3200.0	FEMALE
81	Adelie	Torgersen	42.9	17.6	196.0	4700.0	MALE

82	Adelie	Torgersen	36.7	18.8	187.0	3800.0	FEMALE
83	Adelie	Torgersen	35.1	19.4	193.0	4200.0	MALE

```
penguins[penguins['bill_length_mm']<34]
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
70	Adelie	Torgersen	33.5	19.0	190.0	3600.0	FEMALE
98	Adelie	Dream	33.1	16.1	178.0	2900.0	FEMALE
142	Adelie	Dream	32.1	15.5	188.0	3050.0	FEMALE
122	Adelie	Torgersen	40.2	17.0	176.0	3450.0	FEMALE
123	Adelie	Torgersen	41.4	18.5	202.0	3875.0	MALE

```
# filter more than one criteria/condition  with and
penguins[(penguins['island']=='Torgersen') & (penguins['bill_length_mm']<35)]
# with or
penguins[(penguins['island']=='Torgersen') & (penguins['bill_length_mm']<35)]
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
8	Adelie	Torgersen	34.1	18.1	193.0	3475.0	NaN
14	Adelie	Torgersen	34.6	21.1	198.0	4400.0	MALE
18	Adelie	Torgersen	34.4	18.4	184.0	3325.0	FEMALE
70	Adelie	Torgersen	33.5	19.0	190.0	3600.0	FEMALE
80	Adelie	Torgersen	34.6	17.2	189.0	3200.0	FEMALE

```
# filter with query (method)
penguins.query('island == "Torgersen"')
# more condition
penguins.query('island =="Torgersen" | bill_length_mm <35 ')
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	FEMALE
5	Adelie	Torgersen	39.3	20.6	190.0	3650.0	MALE
6	Adelie	Torgersen	38.9	17.8	181.0	3625.0	FEMALE
7	Adelie	Torgersen	39.2	19.6	195.0	4675.0	MALE
8	Adelie	Torgersen	34.1	18.1	193.0	3475.0	NaN
9	Adelie	Torgersen	42.0	20.2	190.0	4250.0	NaN
10	Adelie	Torgersen	37.8	17.1	186.0	3300.0	NaN
11	Adelie	Torgersen	37.8	17.3	180.0	3700.0	NaN
12	Adelie	Torgersen	41.1	17.6	182.0	3200.0	FEMALE
13	Adelie	Torgersen	38.6	21.2	191.0	3800.0	MALE
14	Adelie	Torgersen	34.6	21.1	198.0	4400.0	MALE
15	Adelie	Torgersen	36.6	17.8	185.0	3700.0	FEMALE
16	Adelie	Torgersen	38.7	19.0	195.0	3450.0	FEMALE
17	Adelie	Torgersen	42.5	20.7	197.0	4500.0	MALE
18	Adelie	Torgersen	34.4	18.4	184.0	3325.0	FEMALE
19	Adelie	Torgersen	46.0	21.5	194.0	4200.0	MALE
54	Adelie	Biscoe	34.5	18.1	187.0	2900.0	FEMALE
68	Adelie	Torgersen	35.9	16.6	190.0	3050.0	FEMALE
69	Adelie	Torgersen	41.8	19.4	198.0	4450.0	MALE
70	Adelie	Torgersen	33.5	19.0	190.0	3600.0	FEMALE
71	Adelie	Torgersen	39.7	18.4	190.0	3900.0	MALE
72	Adelie	Torgersen	39.6	17.2	196.0	3550.0	FEMALE
73	Adelie	Torgersen	45.8	18.9	197.0	4150.0	MALE
74	Adelie	Torgersen	35.5	17.5	190.0	3700.0	FEMALE
75	Adelie	Torgersen	42.8	18.5	195.0	4250.0	MALE
76	Adelie	Torgersen	40.9	16.8	191.0	3700.0	FEMALE
77	Adelie	Torgersen	37.2	19.4	184.0	3900.0	MALE
78	Adelie	Torgersen	36.2	16.1	187.0	3550.0	FEMALE
79	Adelie	Torgersen	42.1	19.1	195.0	4000.0	MALE
80	Adelie	Torgersen	34.6	17.2	189.0	3200.0	FEMALE

81	Adelie	Torgersen	42.9	17.6	196.0	4700.0	MALE
82	Adelie	Torgersen	36.7	18.8	187.0	3800.0	FEMALE

```
#check missing value in each column
penguins.isna().sum()
#filter missing value with Nan
penguins[penguins['sex'].isna()]
#drop NA
clean_penguins = penguins.dropna()
clean_penguins.isna().sum()
```

119	Adelie	Torgersen	41.1	18.6	189.0	3325.0	MALE
120	Adelie	Torgersen	36.2	17.2	187.0	3150.0	FEMALE

```
#sort single column
penguins.sort_values('bill_length_mm',ascending=False)
#sort multiple columns
penguins.sort_values(['island','bill_length_mm'])
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
54	Adelie	Biscoe	34.5	18.1	187.0	2900.0	FEMALE
52	Adelie	Biscoe	35.0	17.9	190.0	3450.0	FEMALE
100	Adelie	Biscoe	35.0	17.9	192.0	3725.0	FEMALE
25	Adelie	Biscoe	35.3	18.9	187.0	3800.0	FEMALE
66	Adelie	Biscoe	35.5	16.2	195.0	3350.0	FEMALE
...	...	...	...	...	...	...	...
131	Adelie	Torgersen	43.1	19.2	197.0	3500.0	MALE
129	Adelie	Torgersen	44.1	18.0	210.0	4000.0	MALE
73	Adelie	Torgersen	45.8	18.9	197.0	4150.0	MALE
19	Adelie	Torgersen	46.0	21.5	194.0	4200.0	MALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN

344 rows × 7 columns

```
#unique values
penguins['island'].unique()
```

```
#count value
penguins['island'].value_counts()
```

```
#count more than one value
penguins[['island','species']].value_counts().reset_index()
```

	island	species	0
0	Biscoe	Gentoo	124
1	Dream	Chinstrap	68
2	Dream	Adelie	56
3	Torgersen	Adelie	52
4	Biscoe	Adelie	44

```
#summarise dataframe
penguins.describe()
```

	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g
count	342.000000	342.000000	342.000000	342.000000
mean	43.921930	17.151170	200.915205	4201.754386
std	5.459584	1.974793	14.061714	801.954536
min	32.100000	13.100000	172.000000	2700.000000
25%	39.225000	15.600000	190.000000	3550.000000
50%	44.450000	17.300000	197.000000	4050.000000
75%	48.500000	18.700000	213.000000	4750.000000
max	59.600000	21.500000	231.000000	6300.000000

```
#summarise df with all column (include text data)
penguins.describe(include='all')
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
count	344	344	342.000000	342.000000	342.000000	342.000000	333
unique	3	3	NaN	NaN	NaN	NaN	2
top	Adelie	Biscoe	NaN	NaN	NaN	NaN	MALE
freq	152	168	NaN	NaN	NaN	NaN	168
mean	NaN	NaN	43.921930	17.151170	200.915205	4201.754386	NaN
std	NaN	NaN	5.459584	1.974793	14.061714	801.954536	NaN
min	NaN	NaN	32.100000	13.100000	172.000000	2700.000000	NaN
25%	NaN	NaN	39.225000	15.600000	190.000000	3550.000000	NaN
50%	NaN	NaN	44.450000	17.300000	197.000000	4050.000000	NaN
75%	NaN	NaN	48.500000	18.700000	213.000000	4750.000000	NaN
max	NaN	NaN	59.600000	21.500000	231.000000	6300.000000	NaN

```
#summarise just average
penguins['bill_length_mm'].mean()
penguins['bill_length_mm'].std()
penguins['bill_length_mm'].median()
```

```
44.45
```

```
#group by + sum/mean
penguins.groupby('species')['bill_length_mm'].mean()
```

```
#groupby aggregation
penguins.groupby('species')['bill_length_mm'].agg(['min','max','median','mean','s
```

	min	max	median	mean	std
species					
Adelie	32.1	46.0	38.80	38.791391	2.663405
Chinstrap	40.9	58.0	49.55	48.833824	3.339256
Gentoo	40.9	59.6	47.30	47.504878	3.081857

```
#groupby more than one column
result = penguins.groupby(['island','species'])['bill_length_mm'].agg(['min','max
#reset index
result = result.reset_index()
result
```

	island	species	min	max
0	Biscoe	Adelie	34.5	45.6
1	Biscoe	Gentoo	40.9	59.6
2	Dream	Adelie	32.1	44.1
3	Dream	Chinstrap	40.9	58.0
4	Torgersen	Adelie	33.5	46.0

```
#write to csv
result.to_csv('result_penguin.csv')
```

```
# if your code lond with \

result = penguins.groupby(['island','species'])['bill_length_mm']\
    .agg(['min','max'])
result
```

		min	max
island	species		
Biscoe	Adelie	34.5	45.6
	Gentoo	40.9	59.6
Dream	Adelie	32.1	44.1
	Chinstrap	40.9	58.0
Torgersen	Adelie	33.5	46.0

```
#map value Male : m , Female : f
penguins['sex'].map({'MALE':'m','FEMALE':'f'}).fillna('other')
```

```
#numpy
import numpy as np
```

```
np.mean(penguins['bill_length_mm'])
```

```
43.9219298245614
```

```
# where with numpy
score = pd.Series([80,92,95,70,64])
grade = np.where(score>=80,"pass","failed")
print(grade)
```

```
['pass' 'pass' 'pass' 'failed' 'failed']
```

```
df = penguins.query("species == 'Adelie'")[['species','island','bill_length_mm']]
```

```
df['new_column'] = np.where(df['bill_length_mm']>40,"true","false")
```

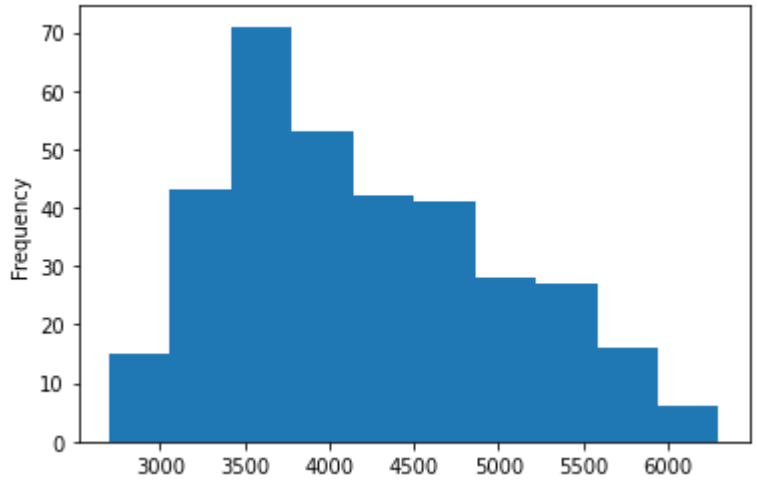
```
left = {
    'key':[1,2,3,4],
    'name':['toy','ben','cat','john'],
    'age':[22,31,42,54]
}

right = {
    'key':[1,2,3,4],
    'city':['BKK','London','New york','Tokyo'],
    'zip':[1001,2032,3452,2314]
}

df_left = pd.DataFrame(left)
df_right = pd.DataFrame(right)
dfmix = pd.merge(df_left,df_right,on='key')
```

```
dfmix
```

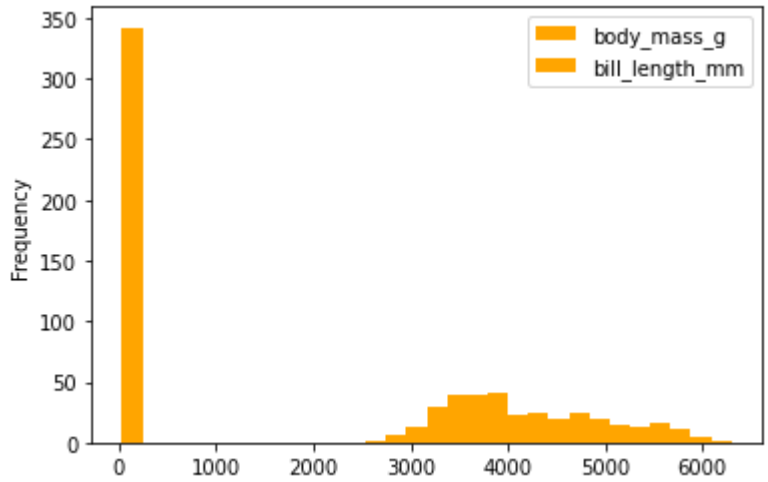
	key	name	age	city	zip
0	1	toy	22	BKK	1001
1	2	ben	31	London	2032
2	3	cat	42	New york	3452
3	4	john	54	Tokyo	2314

```
#pandas plot
#histrogram
penguins['body_mass_g'].plot(kind='hist'); #one dimension
penguins[['body_mass_g','bill_length_mm']].plot(kind='hist',bins=30,color='orange
```

Download



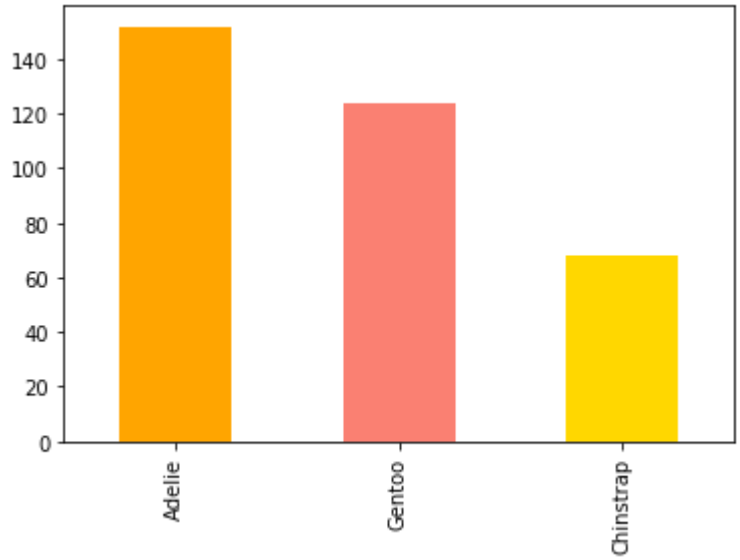
Download




```
#bar plot
penguins['species'].value_counts().plot(kind = 'bar',color=['orange','salmon','go
```

```
<AxesSubplot:>
```

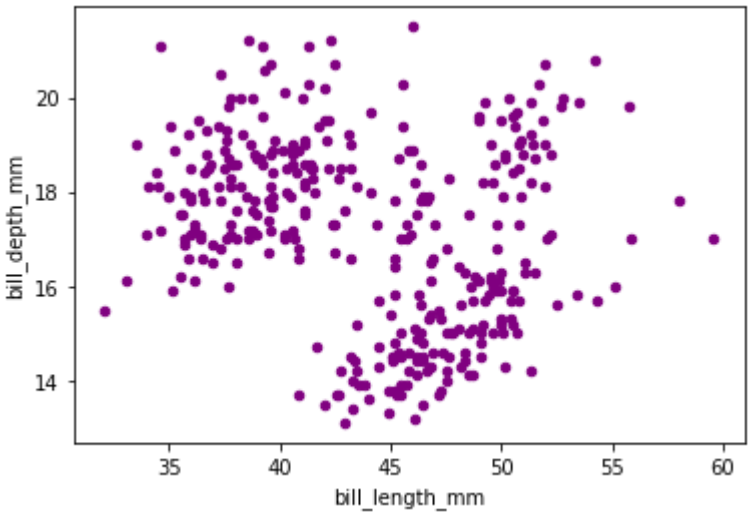
Download




```
#scatter plot
penguins[['bill_length_mm','bill_depth_mm']].plot(x='bill_length_mm',y='bill_dept
```

```
<AxesSubplot:xlabel='bill_length_mm', ylabel='bill_depth_mm'>
```

Download




```
penguins
```

	species	island	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g	sex
0	Adelie	Torgersen	39.1	18.7	181.0	3750.0	MALE
1	Adelie	Torgersen	39.5	17.4	186.0	3800.0	FEMALE
2	Adelie	Torgersen	40.3	18.0	195.0	3250.0	FEMALE
3	Adelie	Torgersen	NaN	NaN	NaN	NaN	NaN
4	Adelie	Torgersen	36.7	19.3	193.0	3450.0	FEMALE
...	...	...	...	...	...	...	...
339	Gentoo	Biscoe	NaN	NaN	NaN	NaN	NaN
340	Gentoo	Biscoe	46.8	14.3	215.0	4850.0	FEMALE
341	Gentoo	Biscoe	50.4	15.7	222.0	5750.0	MALE
342	Gentoo	Biscoe	45.2	14.8	212.0	5200.0	FEMALE
343	Gentoo	Biscoe	49.9	16.1	213.0	5400.0	MALE

344 rows × 7 columns