

Double-click (or enter) to edit

```
ls drive/MyDrive/'bank_train (1).csv'
```

```
↩ 'drive/MyDrive/bank_train (1).csv'
```

#1. Write a Python program to select the 'name' and 'score' columns from the following DataFrame.

```
#Sample DataFrame:
```

```
#exam_data = {'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
```

```
# 'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
```

```
# 'attempts': [1, 3, 4, 3, 5, 3, 6, 1, 7, 1] }
```

```
import pandas as pd
```

```
import numpy as np
```

```
exam_data = {
```

```
    'name': ['Anastasia', 'Dima', 'Katherine', 'James', 'Emily', 'Michael', 'Matthew', 'Laura', 'Kevin', 'Jonas'],
```

```
    'score': [12.5, 9, 16.5, np.nan, 9, 20, 14.5, np.nan, 8, 19],
```

```
    'attempts': [1, 3, 4, 3, 5, 3, 6, 1, 7, 1]
```

```
}
```

```
df = pd.DataFrame(exam_data)
```

```
print(df[['name', 'score']])
```

```
↩
```

	name	score
0	Anastasia	12.5
1	Dima	9.0
2	Katherine	16.5
3	James	NaN
4	Emily	9.0
5	Michael	20.0
6	Matthew	14.5
7	Laura	NaN
8	Kevin	8.0
9	Jonas	19.0

#2. For the above dataframe, Write a program to select the data who's attempt is greater than 3.

```
filtered_df = df[df['attempts'] > 3]
```

```
print(filtered_df)
```

```
↩
```

	name	score	attempts
2	Katherine	16.5	4
4	Emily	9.0	5
6	Matthew	14.5	6
8	Kevin	8.0	7

#Write python code for indexing rows and columns based on the following conditions:

```
#Assume we have the following dataframe:
```

```
data = {'name': ['Alice', 'Bob', 'Charlie', 'Dave'],
```

```
        'age': [25, 35, 40, 28],
```

```
        'gender': ['F', 'M', 'M', 'M'],
```

```
        'salary': [50000, 70000, 60000, 80000]}
```

```
df = pd.DataFrame(data)
```

```
print(df)
```

```
↩
```

	name	age	gender	salary
0	Alice	25	F	50000
1	Bob	35	M	70000
2	Charlie	40	M	60000
3	Dave	28	M	80000

#a. Select rows where age is greater than 30

```
age_df = df[df['age'] > 30]
```

```
print(age_df)
```

```

↕
   name age gender salary
1   Bob  35      M  70000
2 Charlie  40      M  60000

```

```

#b. Select rows where name contains 'e':
fd_df = df[df['name'].str.contains('e')]

```

```

print(fd_df)

```

```

↕
   name age gender salary
0  Alice  25      F  50000
2 Charlie  40      M  60000
3   Dave  28      M  80000

```

```

#c. Select rows where gender is 'M' and salary is greater than 65000:
salary_df = df[(df['gender'] == 'M') & (df['salary'] > 65000)]

```

```

print(salary_df)

```

```

↕
   name age gender salary
1   Bob  35      M  70000
3  Dave  28      M  80000

```

```

#d. Select columns 'name' and 'age'
print(df[['name', 'age']])

```

```

↕
   name age
0  Alice  25
1   Bob  35
2 Charlie  40
3   Dave  28

```

#ACTIVITY 2:

```

df=pd.read_csv("/content/bank_train.csv")

```

```

df.head()

```

```

↕
   age      job marital education default balance housing loan contact day month duration campaign pdays previous p
0   76    retired  married  secondary      no   2302.0      no   no  telephone    5  feb      110         1    87         2
1   66    retired  divorced  unknown      no    53.0      no   no    cellular   12  jul      562         4    -1         0
2   51 management  married   tertiary      no  2455.0     yes   no    cellular   21  jul      553         1    -1         0
3   41 blue-collar  married  secondary      no   356.0     yes   no    cellular   14  may       90         5    -1         0
4   51  technician  married  secondary      no -1944.0     yes   no    cellular    7  may      623         1    -1         0

```

Next steps: [Generate code with df](#) [View recommended plots](#) [New interactive sheet](#)

```

df.tail()

```

```

↕
   age      job marital education default balance housing loan contact day month duration campaign pdays previous p
4461  33 management  married   tertiary      no   133.0     yes   no   unknown   26  may      308         4    -1         0
4462  39  services  divorced  secondary      no   687.0     yes   no    cellular    9  jul      869         1    -1         0
4463  40   admin.   single  secondary      no  2040.0     yes   no    cellular   18  may      906         2   350         2
4464  31  technician  single  secondary      no   628.0     yes   no   unknown   12  may     1083         2    -1         0
4465  70    retired  divorced   primary      no   383.0      no   no    cellular   28  apr       50         2    -1         0

```

```

df.describe()

```



	age	balance	day	duration	campaign	pdays	previous
count	4466.000000	4465.000000	4466.000000	4466.000000	4466.000000	4466.000000	4466.000000
mean	41.100090	1484.334378	15.740484	371.089342	2.484774	52.880878	0.866995
std	11.905566	3253.910473	8.448066	346.904391	2.633638	111.146726	2.381197
min	18.000000	-3058.000000	1.000000	3.000000	1.000000	-1.000000	0.000000
25%	32.000000	107.000000	8.000000	137.000000	1.000000	-1.000000	0.000000
50%	38.000000	539.000000	16.000000	256.000000	2.000000	-1.000000	0.000000
75%	49.000000	1728.000000	22.000000	485.000000	3.000000	64.750000	1.000000
max	93.000000	81204.000000	31.000000	3284.000000	43.000000	828.000000	41.000000



#a. select the rows where clients with primary education have subscribed to a deposit?

```
sub_df = df[(df['education'] == 'primary') & (df['deposit'] == 'yes')]
print(sub_df)
```



	age	job	marital	education	default	balance	housing	loan	\
29	39	blue-collar	divorced	primary	no	1317.0	yes	no	
39	31	unemployed	single	primary	no	163.0	no	no	
56	49	blue-collar	single	primary	no	566.0	yes	no	
66	53	blue-collar	married	primary	yes	-462.0	no	no	
103	42	blue-collar	single	primary	no	4930.0	no	no	
...	
4411	55	housemaid	married	primary	no	0.0	yes	no	
4422	80	retired	married	primary	no	1468.0	no	no	
4451	41	blue-collar	married	primary	no	143.0	yes	yes	
4452	53	blue-collar	married	primary	no	421.0	yes	no	
4458	32	blue-collar	married	primary	no	-454.0	yes	yes	

	contact	day	month	duration	campaign	pdays	previous	poutcome	\
29	cellular	20	nov	543	1	170	4	other	
39	cellular	30	jan	707	2	2	1	other	
56	cellular	25	jul	979	2	-1	0	unknown	
66	cellular	29	jan	470	1	-1	0	unknown	
103	unknown	18	jun	973	1	-1	0	unknown	
...	
4411	cellular	17	jul	1303	2	-1	0	unknown	
4422	cellular	13	jan	330	3	-1	0	unknown	
4451	unknown	2	jun	659	2	-1	0	unknown	
4452	cellular	20	nov	677	1	-1	0	unknown	
4458	cellular	18	may	801	5	355	2	failure	

	deposit
29	yes
39	yes
56	yes
66	yes
103	yes
...	...
4411	yes
4422	yes
4451	yes
4452	yes
4458	yes

[243 rows x 17 columns]

#select the rows where clients who have not subscribed to a deposit?

```
de_df = df[(df['deposit'] == 'no')]
print(de_df)
```



	age	job	marital	education	default	balance	housing	loan	\
0	76	retired	married	secondary	no	2302.0	no	no	
3	41	blue-collar	married	secondary	no	356.0	yes	no	
6	59	retired	married	secondary	no	136.0	no	no	
7	34	blue-collar	married	primary	no	5299.0	yes	no	
9	44	blue-collar	married	secondary	no	879.0	yes	no	
...	
4457	43	management	married	tertiary	no	1336.0	yes	yes	
4460	54	retired	married	secondary	no	522.0	no	yes	
4461	33	management	married	tertiary	no	133.0	yes	no	
4464	31	technician	single	secondary	no	628.0	yes	no	
4465	70	retired	divorced	primary	no	383.0	no	no	
	contact	day	month	duration	campaign	pdays	previous	poutcome	\
0	telephone	5	feb	110	1	87	2	failure	
3	cellular	14	may	90	5	-1	0	unknown	
6	cellular	6	aug	301	4	-1	0	unknown	
7	unknown	26	jun	75	5	-1	0	unknown	
9	cellular	3	apr	383	1	-1	0	unknown	

```

...      ...      ...      ...      ...      ...      ...      ...
4457  cellular  27  may      82      2  309      1  failure
4460  cellular  14  jul      81      3   -1      0  unknown
4461  unknown  26  may     308      4   -1      0  unknown
4464  unknown  12  may    1083      2   -1      0  unknown
4465  cellular  28  apr      50      2   -1      0  unknown

```

```

deposit
0      no
3      no
6      no
7      no
9      no
...
4457   no
4460   no
4461   no
4464   no
4465   no

```

[2354 rows x 17 columns]

```

#c. select the rows where clients who have subscribed to a deposit either have a housing or a personal loan?
pe_df = df[(df['deposit'] == 'yes') & ((df['housing'] == 'yes') | (df['loan'] == 'yes'))]

```

```

print(pe_df)

```

```

➡ age      job      marital  education  default  balance  housing  loan  \
2   51  management  married   tertiary    no   2455.0    yes   no
4   51  technician  married   secondary    no  -1944.0    yes   no
15  37  management   single   tertiary    no    455.0    yes   no
17  24   admin.     single   tertiary    no     0.0    yes   no
21  33   admin.     married   tertiary    no    79.0    yes   no
...  ...      ...      ...      ...      ...      ...      ...
4454 30  blue-collar  single   secondary    no   155.0    yes   yes
4458 32  blue-collar  married   primary     no  -454.0    yes   yes
4459 37  technician   single   secondary    no  3326.0    yes   no
4462 39   services   divorced  secondary    no   687.0    yes   no
4463 40   admin.     single   secondary    no  2040.0    yes   no

```

```

contact  day month  duration  campaign  pdays  previous  poutcome  \
2   cellular  21  jul      553      1    -1      0  unknown
4   cellular   7  may      623      1    -1      0  unknown
15  cellular  13  aug      904      6    -1      0  unknown
17  cellular  27  may      122      2    -1      0  unknown
21  cellular   5  may      389      1   195      4  success
...  ...      ...      ...      ...      ...      ...      ...
4454 cellular   9  jul     1426      3    -1      0  unknown
4458 cellular  18  may      801      5   355      2  failure
4459 unknown  21  may      799      1    -1      0  unknown
4462 cellular   9  jul      869      1    -1      0  unknown
4463 cellular  18  may      906      2   350      2  failure

```

```

deposit
2      yes
4      yes
15     yes
17     yes
21     yes
...
4454   yes
4458   yes
4459   yes
4462   yes
4463   yes

```

[893 rows x 17 columns]

```

#d. select the rows where clients with secondary education who have not subscribed to a deposit?
filtered_df = df[(df['education'] == 'secondary') & (df['deposit'] == 'no')]

```

```

print(filtered_df)

```

```

➡ age      job      marital  education  default  balance  housing  loan  \
0   76  retired  married   secondary    no   2302.0    no   no
3   41  blue-collar  married   secondary    no    356.0    yes   no
6   59  retired  married   secondary    no    136.0    no   no
9   44  blue-collar  married   secondary    no    879.0    yes   no
10  34   services  married   secondary    no   1637.0    yes   no
...  ...      ...      ...      ...      ...      ...      ...
4446 35   services  married   secondary    no     0.0    yes   no
4453 31   services  married   secondary    no    505.0    no   no
4456 54  blue-collar  married   secondary    no   -102.0    yes   no
4460 54   retired  married   secondary    no    522.0    no   yes
4464 31  technician  single   secondary    no    628.0    yes   no

```

	contact	day	month	duration	campaign	pdays	previous	poutcome	\
0	telephone	5	feb	110	1	87	2	failure	
3	cellular	14	may	90	5	-1	0	unknown	
6	cellular	6	aug	301	4	-1	0	unknown	
9	cellular	3	apr	383	1	-1	0	unknown	
10	cellular	21	nov	107	4	-1	0	unknown	
...	
4446	cellular	24	jul	810	1	-1	0	unknown	
4453	cellular	11	jul	773	3	-1	0	unknown	
4456	cellular	27	aug	164	7	-1	0	unknown	
4460	cellular	14	jul	81	3	-1	0	unknown	
4464	unknown	12	may	1083	2	-1	0	unknown	

	deposit
0	no
3	no
6	no
9	no
10	no
...	...
4446	no
4453	no
4456	no
4460	no
4464	no

[1229 rows x 17 columns]

```
#e. select the rows where clients who have subscribed to a term deposit as an outcome of the successful marketing campaign?
filtered_df = df[(df['deposit'] == 'yes') & (df['poutcome'] == 'success')]
```

```
print(filtered_df)
```

	age	job	marital	education	default	balance	housing	loan	\
19	76	self-employed	married	unknown	no	4984.0	no	no	
21	33	admin.	married	tertiary	no	79.0	yes	no	
45	71	retired	divorced	secondary	no	0.0	no	no	
51	68	retired	married	secondary	no	1146.0	no	no	
52	46	management	married	tertiary	no	273.0	yes	no	
...	
4338	38	admin.	divorced	secondary	no	19.0	yes	no	
4372	20	student	single	secondary	no	215.0	no	no	
4376	42	technician	married	secondary	no	994.0	yes	no	
4408	29	housemaid	single	tertiary	no	19.0	no	no	
4448	27	blue-collar	single	secondary	no	535.0	no	no	

	contact	day	month	duration	campaign	pdays	previous	poutcome	\
19	telephone	28	apr	403	1	182	1	success	
21	cellular	5	may	389	1	195	4	success	
45	cellular	26	feb	771	1	171	1	success	
51	cellular	13	may	356	1	71	5	success	
52	cellular	18	mar	910	2	184	4	success	
...	
4338	cellular	5	feb	1130	3	251	2	success	
4372	cellular	24	feb	175	1	92	6	success	
4376	cellular	12	nov	227	3	93	6	success	
4408	cellular	4	may	268	1	88	4	success	
4448	cellular	16	aug	265	3	95	4	success	

	deposit
19	yes
21	yes
45	yes
51	yes
52	yes
...	...
4338	yes
4372	yes
4376	yes
4408	yes
4448	yes

[392 rows x 17 columns]

```
#f. select the rows where unemployed clients who have not subscribed to deposit?
unemployed_no_deposit = df[(df['job'] == 'unemployed') & (df['deposit'] == 'no')]
print(unemployed_no_deposit)
```

	age	job	marital	education	default	balance	housing	loan	\
74	37	unemployed	single	secondary	no	48.0	no	no	
304	48	unemployed	married	secondary	no	855.0	yes	no	
404	57	unemployed	married	primary	no	0.0	yes	no	
464	47	unemployed	divorced	secondary	no	947.0	no	no	
494	45	unemployed	married	tertiary	no	1148.0	no	no	
550	55	unemployed	married	primary	no	8585.0	no	no	
644	35	unemployed	single	secondary	no	2116.0	yes	no	

690	31	unemployed	single	unknown	no	167.0	no	no
705	42	unemployed	divorced	secondary	no	759.0	no	no
811	38	unemployed	married	secondary	no	995.0	no	no
821	43	unemployed	married	secondary	no	1943.0	yes	no
827	52	unemployed	married	secondary	no	1639.0	no	no
856	31	unemployed	married	secondary	no	20.0	no	no
966	42	unemployed	married	secondary	no	-165.0	yes	yes
967	57	unemployed	married	secondary	no	1350.0	no	no
1003	35	unemployed	married	secondary	no	2080.0	yes	no
1123	50	unemployed	married	secondary	no	3478.0	yes	no
1236	43	unemployed	divorced	secondary	no	1854.0	no	no
1351	38	unemployed	divorced	secondary	no	189.0	yes	yes
1424	40	unemployed	divorced	secondary	no	262.0	yes	no
1633	41	unemployed	single	tertiary	no	4517.0	yes	no
1828	31	unemployed	single	secondary	no	209.0	yes	no
1848	41	unemployed	married	primary	no	183.0	yes	no
1850	30	unemployed	married	secondary	no	142.0	yes	no
1961	58	unemployed	married	secondary	no	610.0	yes	no
2060	33	unemployed	single	secondary	no	233.0	yes	no
2184	41	unemployed	divorced	secondary	no	271.0	yes	no
2292	40	unemployed	married	secondary	no	1289.0	no	no
2308	50	unemployed	married	secondary	no	297.0	yes	no
2364	43	unemployed	married	secondary	no	553.0	no	no
2464	55	unemployed	married	primary	no	1221.0	no	yes
2513	55	unemployed	married	secondary	no	512.0	no	no
2603	33	unemployed	single	secondary	no	682.0	no	no
2681	40	unemployed	married	tertiary	no	2430.0	no	no
2699	52	unemployed	married	tertiary	no	2133.0	no	yes
2754	55	unemployed	married	tertiary	no	5345.0	no	no
2762	43	unemployed	married	secondary	no	775.0	no	no
2827	54	unemployed	single	secondary	no	3611.0	yes	no
2913	44	unemployed	married	primary	no	97.0	yes	no
2929	40	unemployed	married	secondary	no	1077.0	yes	yes
3145	36	unemployed	married	secondary	no	439.0	yes	no
3263	47	unemployed	single	secondary	no	4819.0	no	no
3313	45	unemployed	single	secondary	no	382.0	yes	yes
3556	40	unemployed	married	secondary	no	219.0	yes	no
3704	59	unemployed	single	secondary	no	865.0	no	no
3733	46	unemployed	divorced	secondary	no	0.0	no	no
3745	58	unemployed	single	tertiary	no	2094.0	no	no
3905	33	unemployed	married	tertiary	no	3335.0	no	no
3949	46	unemployed	married	secondary	no	2940.0	yes	no
4027	36	unemployed	single	tertiary	no	221.0	no	no
4066	42	unemployed	married	tertiary	no	0.0	no	no
4102	30	unemployed	single	secondary	no	0.0	yes	no
4125	27	unemployed	single	tertiary	no	3060.0	no	no
4130	34	unemployed	married	secondary	no	200.0	yes	no
4140	37	unemployed	married	secondary	no	4769.0	no	no
4179	43	unemployed	divorced	secondary	no	0.0	yes	no

```
#g) select columns 'education' and 'balance' where age is less than or equal to 30
young_clients = df.loc[df['age'] <= 30, ['education', 'balance']]
print(young_clients)
```

```
↔
  education  balance
17    tertiary    0.0
22    primary   544.0
26    secondary   30.0
27    secondary  195.0
40    secondary  743.0
...      ...      ...
4440   tertiary   674.0
4448   secondary   535.0
4449   secondary    81.0
4454   secondary   155.0
4455   tertiary   265.0
```

```
[809 rows x 2 columns]
```