```
pip install seaborn

Requirement already satisfied: seaborn in c:\users\shagu\anaconda3\
lib\site-packages (0.12.2)
Requirement already satisfied: numpy!=1.24.0,>=1.17 in c:\users\shagu\
anaconda3\lib\site-packages (from seaborn) (1.24.3)
Requirement already satisfied: pandas>=0.25 in c:\users\shagu\
anaconda3\lib\site-packages (from seaborn) (1.5.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in c:\users\
shagu\anaconda3\lib\site-packages (from seaborn) (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\shagu\
anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\shagu\
anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\shagu\
anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\shagu\
anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\shagu\
anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(23.0)
Requirement already satisfied: pillow>=6.2.0 in c:\users\shagu\
anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(9.4.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\shagu\
anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\shagu\
anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn)
(2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\shagu\
anaconda3\lib\site-packages (from pandas>=0.25->seaborn) (2022.7)
Requirement already satisfied: six>=1.5 in c:\users\shagu\anaconda3\
lib\site-packages (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.1-
>seaborn) (1.16.0)
Note: you may need to restart the kernel to use updated packages.

import seaborn as sns

import pandas as pd
import matplotlib.pyplot as plt

df=pd.read_csv(r"C:\Users\shagu\Downloads\mtcars (1).csv")
df
```

	model	mpg	cyl	disp	hp	drat	wt	qsec	vs	am
0	Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1
1	Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1
2	Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1
3	Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0
4	Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0
5	Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0
6	Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0
7	Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0
8	Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0
9	Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0
10	Merc 280C	17.8	6	167.6	123	3.92	3.440	18.90	1	0
11	Merc 450SE	16.4	8	275.8	180	3.07	4.070	17.40	0	0
12	Merc 450SL	17.3	8	275.8	180	3.07	3.730	17.60	0	0
13	Merc 450SLC	15.2	8	275.8	180	3.07	3.780	18.00	0	0
14	Cadillac Fleetwood	10.4	8	472.0	205	2.93	5.250	17.98	0	0
15	Lincoln Continental	10.4	8	460.0	215	3.00	5.424	17.82	0	0
16	Chrysler Imperial	14.7	8	440.0	230	3.23	5.345	17.42	0	0
17	Fiat 128	32.4	4	78.7	66	4.08	2.200	19.47	1	1
18	Honda Civic	30.4	4	75.7	52	4.93	1.615	18.52	1	1
19	Toyota Corolla	33.9	4	71.1	65	4.22	1.835	19.90	1	1
20	Toyota Corona	21.5	4	120.1	97	3.70	2.465	20.01	1	0
21	Dodge Challenger	15.5	8	318.0	150	2.76	3.520	16.87	0	0
22	AMC Javelin	15.2	8	304.0	150	3.15	3.435	17.30	0	0
23	Camaro Z28	13.3	8	350.0	245	3.73	3.840	15.41	0	0

24	Pontiac Firebird	19.2	8	400.0	175	3.08	3.845	17.05	0
0									
25	Fiat X1-9	27.3	4	79.0	66	4.08	1.935	18.90	1
1									
26	Porsche 914-2	26.0	4	120.3	91	4.43	2.140	16.70	0
1									
27	Lotus Europa	30.4	4	95.1	113	3.77	1.513	16.90	1
1									
28	Ford Pantera L	15.8	8	351.0	264	4.22	3.170	14.50	0
1									
29	Ferrari Dino	19.7	6	145.0	175	3.62	2.770	15.50	0
1									
30	Maserati Bora	15.0	8	301.0	335	3.54	3.570	14.60	0
1									
31	Volvo 142E	21.4	4	121.0	109	4.11	2.780	18.60	1
1									

	gear	carb
0	4	4
1	4	4
2	4	1
3	3	1
4	3	2
5	3	1
6	3	4
7	4	2
8	4	2
9	4	4
10	4	4
11	3	3
12	3	3
13	3	3
14	3	4
15	3	4
16	3	4
17	4	1
18	4	2
19	4	1
20	3	1
21	3	2
22	3	2
23	3	4
24	3	2
25	4	1
26	5	2
27	5	2
28	5	4
29	5	6

```
30       5       8
31       4       2
```

df.head()

```
              model   mpg  cyl   disp   hp  drat     wt   qsec  vs
am   gear  \
0          Mazda RX4  21.0    6  160.0  110  3.90  2.620  16.46   0
1      4
1      Mazda RX4 Wag  21.0    6  160.0  110  3.90  2.875  17.02   0
1      4
2         Datsun 710  22.8    4  108.0   93  3.85  2.320  18.61   1
1      4
3     Hornet 4 Drive  21.4    6  258.0  110  3.08  3.215  19.44   1
0      3
4  Hornet Sportabout  18.7    8  360.0  175  3.15  3.440  17.02   0
0      3

   carb
0     4
1     4
2     1
3     1
4     2
```

df.tail()

```
             model   mpg  cyl   disp   hp  drat     wt  qsec  vs  am
gear  \
27    Lotus Europa  30.4    4   95.1  113  3.77  1.513  16.9   1   1
5
28  Ford Pantera L  15.8    8  351.0  264  4.22  3.170  14.5   0   1
5
29     Ferrari Dino  19.7    6  145.0  175  3.62  2.770  15.5   0   1
5
30   Maserati Bora  15.0    8  301.0  335  3.54  3.570  14.6   0   1
5
31      Volvo 142E  21.4    4  121.0  109  4.11  2.780  18.6   1   1
4

   carb
27     2
28     4
29     6
30     8
31     2
```

df.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32 entries, 0 to 31
```

```
Data columns (total 12 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   model   32 non-null     object
 1   mpg     32 non-null     float64
 2   cyl     32 non-null     int64
 3   disp    32 non-null     float64
 4   hp      32 non-null     int64
 5   drat    32 non-null     float64
 6   wt      32 non-null     float64
 7   qsec    32 non-null     float64
 8   vs      32 non-null     int64
 9   am      32 non-null     int64
 10  gear    32 non-null     int64
 11  carb    32 non-null     int64
dtypes: float64(5), int64(6), object(1)
memory usage: 3.1+ KB

df.describe()

             mpg        cyl         disp          hp        drat
wt  \
count   32.000000  32.000000    32.000000    32.000000   32.000000
32.000000
mean    20.090625   6.187500   230.721875   146.687500    3.596563
3.217250
std      6.026948   1.785922   123.938694    68.562868    0.534679
0.978457
min     10.400000   4.000000    71.100000    52.000000    2.760000
1.513000
25%     15.425000   4.000000   120.825000    96.500000    3.080000
2.581250
50%     19.200000   6.000000   196.300000   123.000000    3.695000
3.325000
75%     22.800000   8.000000   326.000000   180.000000    3.920000
3.610000
max     33.900000   8.000000   472.000000   335.000000    4.930000
5.424000

             qsec         vs          am        gear       carb
count   32.000000  32.000000   32.000000   32.000000   32.0000
mean    17.848750   0.437500    0.406250    3.687500    2.8125
std      1.786943   0.504016    0.498991    0.737804    1.6152
min     14.500000   0.000000    0.000000    3.000000    1.0000
25%     16.892500   0.000000    0.000000    3.000000    2.0000
50%     17.710000   0.000000    0.000000    4.000000    2.0000
75%     18.900000   1.000000    1.000000    4.000000    4.0000
max     22.900000   1.000000    1.000000    5.000000    8.0000

type(df)
```

```
pandas.core.frame.DataFrame

df.shape

(32, 12)

df.isnull().sum()

model    0
mpg      0
cyl      0
disp     0
hp       0
drat     0
wt       0
qsec     0
vs       0
am       0
gear     0
carb     0
dtype: int64

df.loc[:,"model"].head(10)

0              Mazda RX4
1          Mazda RX4 Wag
2             Datsun 710
3         Hornet 4 Drive
4      Hornet Sportabout
5                Valiant
6             Duster 360
7              Merc 240D
8               Merc 230
9               Merc 280
Name: model, dtype: object

sns.barplot(data = df,x='cyl',y='carb')

<Axes: xlabel='cyl', ylabel='carb'>
```
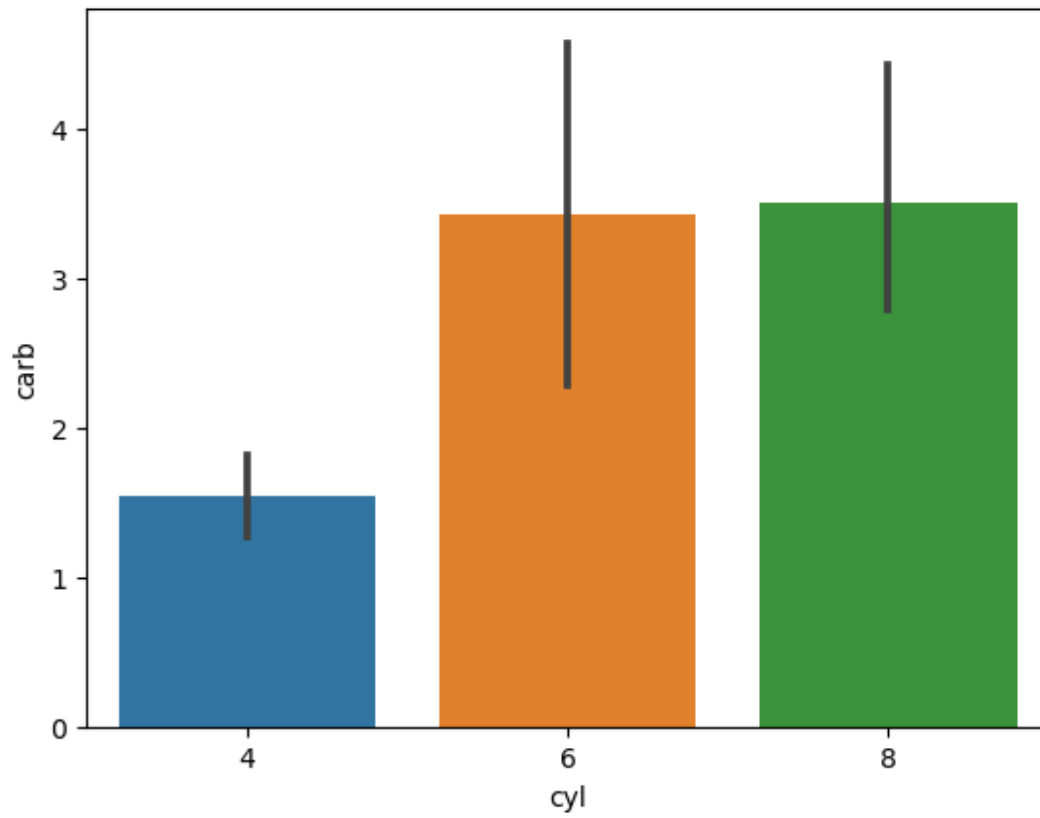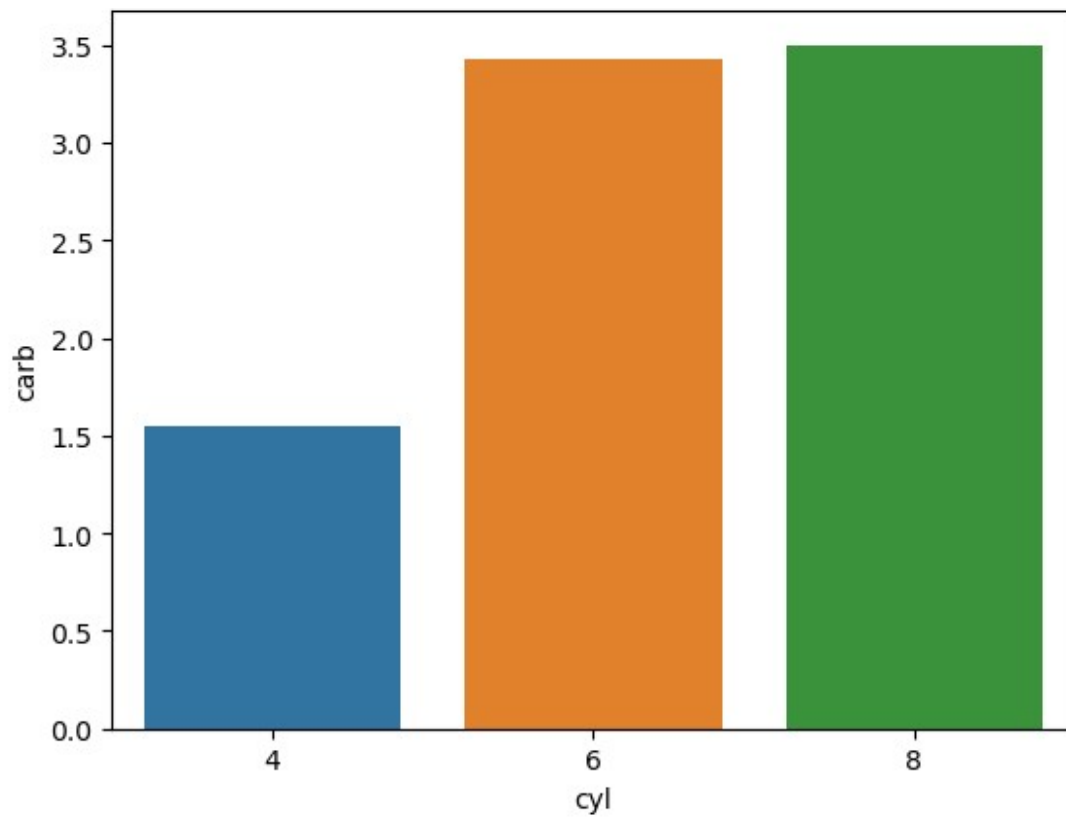
```
sns.barplot(data = df,x='cyl',y='carb', errorbar=None)
<Axes: xlabel='cyl', ylabel='carb'>
```
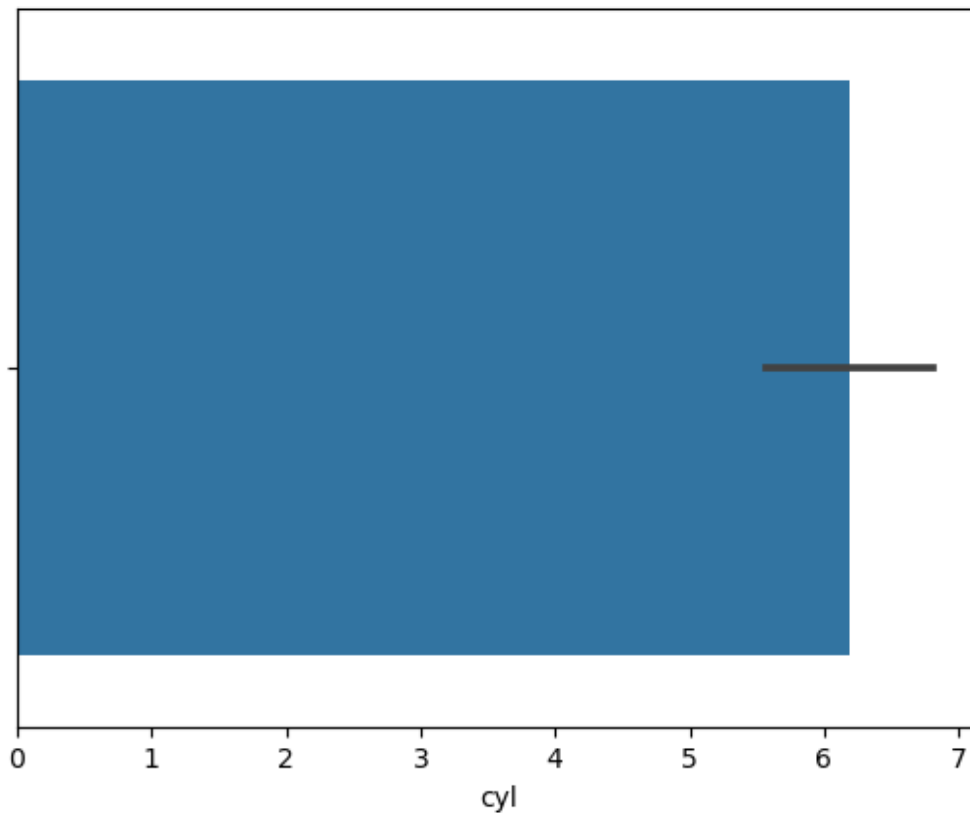
```
sns.barplot(data = df,x='cyl')
```

```
<Axes: xlabel='cyl'>
```

```
import pandas as pd
df  = pd.read_csv(r"C:\Users\shagu\Downloads\supermarket sales Sheet1
(1).csv ")
df
```

	Invoice ID	Branch	City	Customer type	Gender	\
0	750-67-8428	A	Yangon	Member	Female	
1	226-31-3081	C	Naypyitaw	Normal	Female	
2	631-41-3108	A	Yangon	Normal	Male	
3	123-19-1176	A	Yangon	Member	Male	
4	373-73-7910	A	Yangon	Normal	Male	
..	...	...	...	...	...	
995	233-67-5758	C	Naypyitaw	Normal	Male	
996	303-96-2227	B	Mandalay	Normal	Female	
997	727-02-1313	A	Yangon	Member	Male	
998	347-56-2442	A	Yangon	Normal	Male	
999	849-09-3807	A	Yangon	Member	Female	

	Product line	Unit price	Quantity	Tax 5%	Total	\
0	Health and beauty	74.69	7	26.1415	548.9715	
1	Electronic accessories	15.28	5	3.8200	80.2200	
2	Home and lifestyle	46.33	7	16.2155	340.5255	

3	Health and beauty	58.22	8	23.2880	489.0480
4	Sports and travel	86.31	7	30.2085	634.3785
..	...	...	...	...	...
995	Health and beauty	40.35	1	2.0175	42.3675
996	Home and lifestyle	97.38	10	48.6900	1022.4900
997	Food and beverages	31.84	1	1.5920	33.4320
998	Home and lifestyle	65.82	1	3.2910	69.1110
999	Fashion accessories	88.34	7	30.9190	649.2990

```
           Date   Time      Payment     cogs  gross margin percentage  \
0      1/5/2019  13:08      Ewallet   522.83                  4.761905
1      3/8/2019  10:29         Cash    76.40                  4.761905
2      3/3/2019  13:23  Credit card   324.31                  4.761905
3     1/27/2019  20:33      Ewallet   465.76                  4.761905
4      2/8/2019  10:37      Ewallet   604.17                  4.761905
..          ...    ...          ...      ...                       ...
995   1/29/2019  13:46      Ewallet    40.35                  4.761905
996    3/2/2019  17:16      Ewallet   973.80                  4.761905
997    2/9/2019  13:22         Cash    31.84                  4.761905
998   2/22/2019  15:33         Cash    65.82                  4.761905
999   2/18/2019  13:28         Cash   618.38                  4.761905

     gross income  Rating
0         26.1415     9.1
1          3.8200     9.6
2         16.2155     7.4
3         23.2880     8.4
4         30.2085     5.3
..            ...     ...
995        2.0175     6.2
996       48.6900     4.4
997        1.5920     7.7
998        3.2910     4.1
999       30.9190     6.6

[1000 rows x 17 columns]
```