Import and Inspect|Data Frame|Pandas|Python|Solved Exercise Practice Exercise.

In [1]:
```
import pandas as pd
```

Problem 1: Import MPG dataset as car datsframe

In [3]:
```
car=pd.read_csv("https://github.com/YBI-Foundation/Dataset/raw/main/MPG.csv")
```

Problem 2: Print car dataframe

In [4]:
```
car
```

Out[4]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	nam
0	18.0	8	307.0	130.0	3504	12.0	70	usa	chevrole chevell malib
1	15.0	8	350.0	165.0	3693	11.5	70	usa	buic skylar 32
2	18.0	8	318.0	150.0	3436	11.0	70	usa	plymout satellit
3	16.0	8	304.0	150.0	3433	12.0	70	usa	am rebel ss
4	17.0	8	302.0	140.0	3449	10.5	70	usa	for torin
...	...	...	...	...	...	...	...	...	.
393	27.0	4	140.0	86.0	2790	15.6	82	usa	for mustan g
394	44.0	4	97.0	52.0	2130	24.6	82	europe	vw picku
395	32.0	4	135.0	84.0	2295	11.6	82	usa	dodg rampag
396	28.0	4	120.0	79.0	2625	18.6	82	usa	for range
397	31.0	4	119.0	82.0	2720	19.4	82	usa	chevy s 1

398 rows × 9 columns

Problem 3: Inspect first 10 rows

In [5]:
```
car.head(10)
```

Out[5]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	name

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	name
0	18.0	8	307.0	130.0	3504	12.0	70	usa	chevrolet chevelle malibu
1	15.0	8	350.0	165.0	3693	11.5	70	usa	buick skylark 320
2	18.0	8	318.0	150.0	3436	11.0	70	usa	plymouth satellite
3	16.0	8	304.0	150.0	3433	12.0	70	usa	amc rebel sst
4	17.0	8	302.0	140.0	3449	10.5	70	usa	ford torino
5	15.0	8	429.0	198.0	4341	10.0	70	usa	ford galaxie 500
6	14.0	8	454.0	220.0	4354	9.0	70	usa	chevrolet impala
7	14.0	8	440.0	215.0	4312	8.5	70	usa	plymouth fury iii
8	14.0	8	455.0	225.0	4425	10.0	70	usa	pontiac catalina
9	15.0	8	390.0	190.0	3850	8.5	70	usa	amc ambassador dpl

Problem 4: Inspect last 5 rows

In [6]:
```
car.tail()
```

Out[6]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	name
393	27.0	4	140.0	86.0	2790	15.6	82	usa	ford mustang gl
394	44.0	4	97.0	52.0	2130	24.6	82	europe	vw pickup
395	32.0	4	135.0	84.0	2295	11.6	82	usa	dodge rampage
396	28.0	4	120.0	79.0	2625	18.6	82	usa	ford ranger
397	31.0	4	119.0	82.0	2720	19.4	82	usa	chevy s-10

Problem 5: View all rows

In [7]:
```
pd.options.display.max_rows=400
```

In [8]:
```
car
```

Out[8]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
0	18.0	8	307.0	130.0	3504	12.0	70	usa	chevr chev ma
1	15.0	8	350.0	165.0	3693	11.5	70	usa	bı skylark
2	18.0	8	318.0	150.0	3436	11.0	70	usa	plymc sate
3	16.0	8	304.0	150.0	3433	12.0	70	usa	amc re
4	17.0	8	302.0	140.0	3449	10.5	70	usa	ford to
5	15.0	8	429.0	198.0	4341	10.0	70	usa	ford gal
6	14.0	8	454.0	220.0	4354	9.0	70	usa	chevr imp
7	14.0	8	440.0	215.0	4312	8.5	70	usa	plymc fur
8	14.0	8	455.0	225.0	4425	10.0	70	usa	pon cata
9	15.0	8	390.0	190.0	3850	8.5	70	usa	a ambassa
10	15.0	8	383.0	170.0	3563	10.0	70	usa	do challer
11	14.0	8	340.0	160.0	3609	8.0	70	usa	plymc 'cuda
12	15.0	8	400.0	150.0	3761	9.5	70	usa	chevr monte c
13	14.0	8	455.0	225.0	3086	10.0	70	usa	buick es wagon
14	24.0	4	113.0	95.0	2372	15.0	70	japan	toy cor ma
15	22.0	6	198.0	95.0	2833	15.5	70	usa	plymc du
16	18.0	6	199.0	97.0	2774	15.5	70	usa	amc hoı
17	21.0	6	200.0	85.0	2587	16.0	70	usa	f mave
18	27.0	4	97.0	88.0	2130	14.5	70	japan	dat pl
19	26.0	4	97.0	46.0	1835	20.5	70	europe	volkswa 1131 del se
20	25.0	4	110.0	87.0	2672	17.5	70	europe	peug
21	24.0	4	107.0	90.0	2430	14.5	70	europe	audi 1C

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
22	25.0	4	104.0	95.0	2375	17.5	70	europe	saab
23	26.0	4	121.0	113.0	2234	12.5	70	europe	bmw 2
24	21.0	6	199.0	90.0	2648	15.0	70	usa	a grer
25	10.0	8	360.0	215.0	4615	14.0	70	usa	ford f
26	10.0	8	307.0	200.0	4376	15.0	70	usa	chevy
27	11.0	8	318.0	210.0	4382	13.5	70	usa	dodge d
28	9.0	8	304.0	193.0	4732	18.5	70	usa	hi 12
29	27.0	4	97.0	88.0	2130	14.5	71	japan	dat pl
30	28.0	4	140.0	90.0	2264	15.5	71	usa	chevr vega 2
31	25.0	4	113.0	95.0	2228	14.0	71	japan	toy cor
32	25.0	4	98.0	NaN	2046	19.0	71	usa	ford p
33	19.0	6	232.0	100.0	2634	13.0	71	usa	a grer
34	16.0	6	225.0	105.0	3439	15.5	71	usa	plymc sate cust
35	17.0	6	250.0	100.0	3329	15.5	71	usa	chevr chev ma
36	19.0	6	250.0	88.0	3302	15.5	71	usa	ford to
37	18.0	6	232.0	100.0	3288	15.5	71	usa	a mata
38	14.0	8	350.0	165.0	4209	12.0	71	usa	chevr imp
39	14.0	8	400.0	175.0	4464	11.5	71	usa	pon cata brougl
40	14.0	8	351.0	153.0	4154	13.5	71	usa	ford gal
41	14.0	8	318.0	150.0	4096	13.0	71	usa	plymc fur
42	12.0	8	383.0	180.0	4955	11.5	71	usa	do mon
43	13.0	8	400.0	170.0	4746	12.0	71	usa	f cou squire
44	13.0	8	400.0	175.0	5140	12.0	71	usa	pon safari

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
45	18.0	6	258.0	110.0	2962	13.5	71	usa	amc hor sportab
46	22.0	4	140.0	72.0	2408	19.0	71	usa	chevr vega
47	19.0	6	250.0	100.0	3282	15.0	71	usa	pon firel
48	18.0	6	250.0	88.0	3139	14.5	71	usa	must
49	23.0	4	122.0	86.0	2220	14.0	71	usa	merc capri 2
50	28.0	4	116.0	90.0	2123	14.0	71	europe	opel 1
51	30.0	4	79.0	70.0	2074	19.5	71	europe	peug
52	30.0	4	88.0	76.0	2065	14.5	71	europe	fiat 1
53	31.0	4	71.0	65.0	1773	19.0	71	japan	toy corolla 1
54	35.0	4	72.0	69.0	1613	18.0	71	japan	dat 1
55	27.0	4	97.0	60.0	1834	19.0	71	europe	volkswa model
56	26.0	4	91.0	70.0	1955	20.5	71	usa	plymo cri
57	24.0	4	113.0	95.0	2278	15.5	72	japan	toy cor hard
58	25.0	4	97.5	80.0	2126	17.0	72	usa	dodge hard
59	23.0	4	97.0	54.0	2254	23.5	72	europe	volkswa typ
60	20.0	4	140.0	90.0	2408	19.5	72	usa	chevr v
61	21.0	4	122.0	86.0	2226	16.5	72	usa	ford p runab
62	13.0	8	350.0	165.0	4274	12.0	72	usa	chevr imp
63	14.0	8	400.0	175.0	4385	12.0	72	usa	pon cata
64	15.0	8	318.0	150.0	4135	13.5	72	usa	plymo fur
65	14.0	8	351.0	153.0	4129	13.0	72	usa	ford gal
66	17.0	8	304.0	150.0	3672	11.5	72	usa	ambassa

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
67	11.0	8	429.0	208.0	4633	11.0	72	usa	merc marc
68	13.0	8	350.0	155.0	4502	13.5	72	usa	bu lesa cust
69	12.0	8	350.0	160.0	4456	13.5	72	usa	oldsmo delta roy
70	13.0	8	400.0	190.0	4422	12.5	72	usa	chry newp r
71	19.0	3	70.0	97.0	2330	13.5	72	japan	mazda co
72	15.0	8	304.0	150.0	3892	12.5	72	usa	a mata
73	13.0	8	307.0	130.0	4098	14.0	72	usa	chevr chev conco
74	13.0	8	302.0	140.0	4294	16.0	72	usa	ford g torino
75	14.0	8	318.0	150.0	4077	14.0	72	usa	plymo sate custom
76	18.0	4	121.0	112.0	2933	14.5	72	europe	volvo 1
77	22.0	4	121.0	76.0	2511	18.0	72	europe	volkswa 411
78	21.0	4	120.0	87.0	2979	19.5	72	europe	peug 504
79	26.0	4	96.0	69.0	2189	18.0	72	europe	renaul
80	22.0	4	122.0	86.0	2395	16.0	72	usa	ford p
81	28.0	4	97.0	92.0	2288	17.0	72	japan	datsun
82	23.0	4	120.0	97.0	2506	14.5	72	japan	toyo cor mark ii
83	28.0	4	98.0	80.0	2164	15.0	72	usa	dodge
84	27.0	4	97.0	88.0	2100	16.5	72	japan	toy corolla 1
85	13.0	8	350.0	175.0	4100	13.0	73	usa	bu century

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
86	14.0	8	304.0	150.0	3672	11.5	73	usa	a mata
87	13.0	8	350.0	145.0	3988	13.0	73	usa	chevr ma
88	14.0	8	302.0	137.0	4042	14.5	73	usa	ford g to
89	15.0	8	318.0	150.0	3777	12.5	73	usa	do corc cust
90	12.0	8	429.0	198.0	4952	11.5	73	usa	merc marc brough
91	13.0	8	400.0	150.0	4464	12.0	73	usa	chevr cap cla
92	13.0	8	351.0	158.0	4363	13.0	73	usa	forc
93	14.0	8	318.0	150.0	4237	14.5	73	usa	plymc fury g se
94	13.0	8	440.0	215.0	4735	11.0	73	usa	chry new yo brough
95	12.0	8	455.0	225.0	4951	11.0	73	usa	b electra cust
96	13.0	8	360.0	175.0	3821	11.0	73	usa	a ambassa brough
97	18.0	6	225.0	105.0	3121	16.5	73	usa	plymc val
98	16.0	6	250.0	100.0	3278	18.0	73	usa	chevr n cust
99	18.0	6	232.0	100.0	2945	16.0	73	usa	amc hor
100	18.0	6	250.0	88.0	3021	16.5	73	usa	f mave
101	23.0	6	198.0	95.0	2904	16.0	73	usa	plymc du
102	26.0	4	97.0	46.0	1950	21.0	73	europe	volkswa su be
103	11.0	8	400.0	150.0	4997	14.0	73	usa	chevr imp
104	12.0	8	400.0	167.0	4906	12.5	73	usa	f cou
105	13.0	8	360.0	170.0	4654	13.0	73	usa	plymc cust sub

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
106	12.0	8	350.0	180.0	4499	12.5	73	usa	oldsmo vista cru
107	18.0	6	232.0	100.0	2789	15.0	73	usa	a grer
108	20.0	4	97.0	88.0	2279	19.0	73	japan	toy ca
109	21.0	4	140.0	72.0	2401	19.5	73	usa	chevr v
110	22.0	4	108.0	94.0	2379	16.5	73	japan	datsun
111	18.0	3	70.0	90.0	2124	13.5	73	japan	maxda
112	19.0	4	122.0	85.0	2310	18.5	73	usa	ford p
113	21.0	6	155.0	107.0	2472	14.0	73	usa	merc capr
114	26.0	4	98.0	90.0	2265	15.5	73	europe	fiat sport co
115	15.0	8	350.0	145.0	4082	13.0	73	usa	chevr monte c
116	16.0	8	400.0	230.0	4278	9.5	73	usa	pon grand
117	29.0	4	68.0	49.0	1867	19.5	73	europe	fiat
118	24.0	4	116.0	75.0	2158	15.5	73	europe	opel ma
119	20.0	4	114.0	91.0	2582	14.0	73	europe	audi 1
120	19.0	4	121.0	112.0	2868	15.5	73	europe	volvo 14
121	15.0	8	318.0	150.0	3399	11.0	73	usa	dodge cust
122	24.0	4	121.0	110.0	2660	14.0	73	europe	saab 9
123	20.0	6	156.0	122.0	2807	13.5	73	japan	toyota m
124	11.0	8	350.0	180.0	3664	11.0	73	usa	oldsmo om
125	20.0	6	198.0	95.0	3102	16.5	74	usa	plymc du
126	21.0	6	200.0	NaN	2875	17.0	74	usa	f mave
127	19.0	6	232.0	100.0	2901	16.0	74	usa	amc hor
128	15.0	6	250.0	100.0	3336	17.0	74	usa	chevr n
129	31.0	4	79.0	67.0	1950	19.0	74	japan	dat b
130	26.0	4	122.0	80.0	2451	16.5	74	usa	ford p
131	32.0	4	71.0	65.0	1836	21.0	74	japan	toy corolla 1

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
132	25.0	4	140.0	75.0	2542	17.0	74	usa	chevr v
133	16.0	6	250.0	100.0	3781	17.0	74	usa	chevr chev ma cla
134	16.0	6	258.0	110.0	3632	18.0	74	usa	a mata
135	18.0	6	225.0	105.0	3613	16.5	74	usa	plymo sate seb
136	16.0	8	302.0	140.0	4141	14.0	74	usa	ford g to
137	13.0	8	350.0	150.0	4699	14.5	74	usa	bu cen luxus
138	14.0	8	318.0	150.0	4457	13.5	74	usa	do corc custom
139	14.0	8	302.0	140.0	4638	16.0	74	usa	ford g torino
140	14.0	8	304.0	150.0	4257	15.5	74	usa	a mata
141	29.0	4	98.0	83.0	2219	16.5	74	europe	audi
142	26.0	4	79.0	67.0	1963	15.5	74	europe	volkswa das
143	26.0	4	97.0	78.0	2300	14.5	74	europe	opel ma
144	31.0	4	76.0	52.0	1649	16.5	74	japan	toy cor
145	32.0	4	83.0	61.0	2003	19.0	74	japan	datsun
146	28.0	4	90.0	75.0	2125	14.5	74	usa	dodge
147	24.0	4	90.0	75.0	2108	15.5	74	europe	fiat
148	26.0	4	116.0	75.0	2246	14.0	74	europe	fiat 12
149	24.0	4	120.0	97.0	2489	15.0	74	japan	honda
150	26.0	4	108.0	93.0	2391	15.5	74	japan	sub
151	31.0	4	79.0	67.0	2000	16.0	74	europe	fiat
152	19.0	6	225.0	95.0	3264	16.0	75	usa	plymo val cust
153	18.0	6	250.0	105.0	3459	16.0	75	usa	chevr n
154	15.0	6	250.0	72.0	3432	21.0	75	usa	merc mona

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
155	15.0	6	250.0	72.0	3158	19.5	75	usa	f mave
156	16.0	8	400.0	170.0	4668	11.5	75	usa	pon cata
157	15.0	8	350.0	145.0	4440	14.0	75	usa	chevr be
158	16.0	8	318.0	150.0	4498	14.5	75	usa	plymc grand
159	14.0	8	351.0	148.0	4657	13.5	75	usa	forc
160	17.0	6	231.0	110.0	3907	21.0	75	usa	bı cen
161	16.0	6	250.0	105.0	3897	18.5	75	usa	chevr chev ma
162	15.0	6	258.0	110.0	3730	19.0	75	usa	a mata
163	18.0	6	225.0	95.0	3785	19.0	75	usa	plymc
164	21.0	6	231.0	110.0	3039	15.0	75	usa	bı skyh
165	20.0	8	262.0	110.0	3221	13.5	75	usa	chevr monza
166	13.0	8	302.0	129.0	3169	12.0	75	usa	f mustar
167	29.0	4	97.0	75.0	2171	16.0	75	japan	toy cor
168	23.0	4	140.0	83.0	2639	17.0	75	usa	ford p
169	20.0	6	232.0	100.0	2914	16.0	75	usa	a grer
170	23.0	4	140.0	78.0	2592	18.5	75	usa	pon a
171	24.0	4	134.0	96.0	2702	13.5	75	japan	toy cor
172	25.0	4	90.0	71.0	2223	16.5	75	europe	volkswa das
173	24.0	4	119.0	97.0	2545	17.0	75	japan	datsun
174	18.0	6	171.0	97.0	2984	14.5	75	usa	ford p
175	29.0	4	90.0	70.0	1937	14.0	75	europe	volkswa ral
176	19.0	6	232.0	90.0	3211	17.0	75	usa	amc pa
177	23.0	4	115.0	95.0	2694	15.0	75	europe	audi 1
178	23.0	4	120.0	88.0	2957	17.0	75	europe	peug
179	22.0	4	121.0	98.0	2945	14.5	75	europe	volvo 24

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
180	25.0	4	121.0	115.0	2671	13.5	75	europe	saab 9
181	33.0	4	91.0	53.0	1795	17.5	75	japan	honda c c
182	28.0	4	107.0	86.0	2464	15.5	76	europe	fiat
183	25.0	4	116.0	81.0	2220	16.9	76	europe	opel 1
184	25.0	4	140.0	92.0	2572	14.9	76	usa	cap
185	26.0	4	98.0	79.0	2255	17.7	76	usa	dodge
186	27.0	4	101.0	83.0	2202	15.3	76	europe	renault
187	17.5	8	305.0	140.0	4215	13.0	76	usa	chevr chev ma cla
188	16.0	8	318.0	150.0	4190	13.0	76	usa	do corc brougl
189	15.5	8	304.0	120.0	3962	13.9	76	usa	a mata
190	14.5	8	351.0	152.0	4215	12.8	76	usa	ford g to
191	22.0	6	225.0	100.0	3233	15.4	76	usa	plymo val
192	22.0	6	250.0	105.0	3353	14.5	76	usa	chevr n
193	24.0	6	200.0	81.0	3012	17.6	76	usa	f mave
194	22.5	6	232.0	90.0	3085	17.6	76	usa	amc hor
195	29.0	4	85.0	52.0	2035	22.2	76	usa	chevr chev
196	24.5	4	98.0	60.0	2164	22.1	76	usa	chevr wo
197	29.0	4	90.0	70.0	1937	14.2	76	europe	vw ral
198	33.0	4	91.0	53.0	1795	17.4	76	japan	honda c
199	20.0	6	225.0	100.0	3651	17.7	76	usa	do asper
200	18.0	6	250.0	78.0	3574	21.0	76	usa	f gran g
201	18.5	6	250.0	110.0	3645	16.2	76	usa	pon ventur
202	17.5	6	258.0	95.0	3193	17.8	76	usa	amc pa
203	29.5	4	97.0	71.0	1825	12.2	76	europe	volkswa ral

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
204	32.0	4	85.0	70.0	1990	17.0	76	japan	datsui
205	28.0	4	97.0	75.0	2155	16.4	76	japan	toy cor
206	26.5	4	140.0	72.0	2565	13.6	76	usa	ford p
207	20.0	4	130.0	102.0	3150	15.7	76	europe	volvo
208	13.0	8	318.0	150.0	3940	13.2	76	usa	plymc vo premie
209	19.0	4	120.0	88.0	3270	21.9	76	europe	peug
210	19.0	6	156.0	108.0	2930	15.5	76	japan	toyota m
211	16.5	6	168.0	120.0	3820	16.7	76	europe	merced benz 2
212	16.5	8	350.0	180.0	4380	12.1	76	usa	cad se
213	13.0	8	350.0	145.0	4055	12.0	76	usa	chevy
214	13.0	8	302.0	130.0	3870	15.0	76	usa	ford f
215	13.0	8	318.0	150.0	3755	14.0	76	usa	dodge d
216	31.5	4	98.0	68.0	2045	18.5	77	japan	ho accord (
217	30.0	4	111.0	80.0	2155	14.8	77	usa	buick c is del
218	36.0	4	79.0	58.0	1825	18.6	77	europe	renault 5
219	25.5	4	122.0	96.0	2300	15.5	77	usa	plymc arrov
220	33.5	4	85.0	70.0	1945	16.8	77	japan	datsun f hatchb
221	17.5	8	305.0	145.0	3880	12.5	77	usa	chevr cap cla
222	17.0	8	260.0	110.0	4060	19.0	77	usa	oldsmo cut supre
223	15.5	8	318.0	145.0	4140	13.7	77	usa	do mon brougl
224	15.0	8	302.0	130.0	4295	14.9	77	usa	merc cou brougl
225	17.5	6	250.0	110.0	3520	16.4	77	usa	chevr conco
226	20.5	6	231.0	105.0	3425	16.9	77	usa	bı sky

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
227	19.0	6	225.0	100.0	3630	17.7	77	usa	plymo vo cust
228	18.5	6	250.0	98.0	3525	19.0	77	usa	f gran
229	16.0	8	400.0	180.0	4220	11.1	77	usa	pon grand pr
230	15.5	8	350.0	170.0	4165	11.4	77	usa	chevr monte c lan
231	15.5	8	400.0	190.0	4325	12.2	77	usa	chry cord
232	16.0	8	351.0	149.0	4335	14.5	77	usa	f thunder
233	29.0	4	97.0	78.0	1940	14.5	77	europe	volkswa ra cust
234	24.5	4	151.0	88.0	2740	16.0	77	usa	pon sun co
235	26.0	4	97.0	75.0	2265	18.2	77	japan	toy cor liftb
236	25.5	4	140.0	89.0	2755	15.8	77	usa	f mustar 2
237	30.5	4	98.0	63.0	2051	17.0	77	usa	chevr chev
238	33.5	4	98.0	83.0	2075	15.9	77	usa	dodge n
239	30.0	4	97.0	67.0	1985	16.4	77	japan	subar
240	30.5	4	97.0	78.0	2190	14.1	77	europe	volkswa das
241	22.0	6	146.0	97.0	2815	14.5	77	japan	datsun
242	21.5	4	121.0	110.0	2600	12.8	77	europe	bmw 3
243	21.5	3	80.0	110.0	2720	13.5	77	japan	mazda
244	43.1	4	90.0	48.0	1985	21.5	78	europe	volkswa ra cust di
245	36.1	4	98.0	66.0	1800	14.4	78	usa	ford fi
246	32.8	4	78.0	52.0	1985	19.4	78	japan	mazda del
247	39.4	4	85.0	70.0	2070	18.6	78	japan	dat b21
248	36.1	4	91.0	60.0	1800	16.4	78	japan	honda c c

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
249	19.9	8	260.0	110.0	3365	15.5	78	usa	oldsmo cut sa brougl
250	19.4	8	318.0	140.0	3735	13.2	78	usa	do diplo
251	20.2	8	302.0	139.0	3570	12.8	78	usa	merc mona
252	19.2	6	231.0	105.0	3535	19.2	78	usa	pon phoer
253	20.5	6	200.0	95.0	3155	18.2	78	usa	chevr ma
254	20.2	6	200.0	85.0	2965	15.8	78	usa	fairm (a
255	25.1	4	140.0	88.0	2720	15.4	78	usa	fairm (n
256	20.5	6	225.0	100.0	3430	17.2	78	usa	plymc vo
257	19.4	6	232.0	90.0	3210	17.2	78	usa	conc
258	20.6	6	231.0	105.0	3380	15.8	78	usa	b cen spe
259	20.8	6	200.0	85.0	3070	16.7	78	usa	merc zep
260	18.6	6	225.0	110.0	3620	18.7	78	usa	do as
261	18.1	6	258.0	120.0	3410	15.1	78	usa	concord
262	19.2	8	305.0	145.0	3425	13.2	78	usa	chevr monte c lan
263	17.7	6	231.0	165.0	3445	13.4	78	usa	buick re sport co (tu
264	18.1	8	302.0	139.0	3205	11.2	78	usa	ford fu
265	17.5	8	318.0	140.0	4080	13.7	78	usa	do magnun
266	30.0	4	98.0	68.0	2155	16.5	78	usa	chevr chev
267	27.5	4	134.0	95.0	2560	14.2	78	japan	toy cor
268	27.2	4	119.0	97.0	2300	14.7	78	japan	datsun
269	30.9	4	105.0	75.0	2230	14.5	78	usa	dodge o

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
270	21.1	4	134.0	95.0	2515	14.8	78	japan	toy celic liftb
271	23.2	4	156.0	105.0	2745	16.7	78	usa	plymc sapp
272	23.8	4	151.0	85.0	2855	17.6	78	usa	oldsmo starfir
273	23.9	4	119.0	97.0	2405	14.9	78	japan	datsun 2
274	20.3	5	131.0	103.0	2830	15.9	78	europe	audi 5
275	17.0	6	163.0	125.0	3140	13.6	78	europe	volvo 26
276	21.6	4	121.0	115.0	2795	15.7	78	europe	saab 9
277	16.2	6	163.0	133.0	3410	15.8	78	europe	peug 6
278	31.5	4	89.0	71.0	1990	14.9	78	europe	volkswa scirc
279	29.5	4	98.0	68.0	2135	16.6	78	japan	ho accor
280	21.5	6	231.0	115.0	3245	15.4	79	usa	pon leman
281	19.8	6	200.0	85.0	2990	18.2	79	usa	merc zeph
282	22.3	4	140.0	88.0	2890	17.3	79	usa	f fairmo
283	20.2	6	232.0	90.0	3265	18.2	79	usa	a concor
284	20.6	6	225.0	110.0	3360	16.6	79	usa	do aspe
285	17.0	8	305.0	130.0	3840	15.4	79	usa	chevr cap cla
286	17.6	8	302.0	129.0	3725	13.4	79	usa	ford lan
287	16.5	8	351.0	138.0	3955	13.2	79	usa	merc gr marc
288	18.2	8	318.0	135.0	3830	15.2	79	usa	dodg r
289	16.9	8	350.0	155.0	4360	14.9	79	usa	buick es wagon
290	15.5	8	351.0	142.0	4054	14.3	79	usa	f cou squire
291	19.2	8	267.0	125.0	3605	15.0	79	usa	chevr ma classic

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
292	18.5	8	360.0	150.0	3940	13.0	79	usa	chry leba towi country
293	31.9	4	89.0	71.0	1925	14.0	79	europe	vw ral cus
294	34.1	4	86.0	65.0	1975	15.2	79	japan	maxda del
295	35.7	4	98.0	80.0	1915	14.4	79	usa	dodge hatchb cus
296	27.4	4	121.0	80.0	2670	15.0	79	usa	amc spir
297	25.4	5	183.0	77.0	3530	20.1	79	europe	merce benz 3
298	23.0	8	350.0	125.0	3900	17.4	79	usa	cad eldor
299	27.2	4	141.0	71.0	3190	24.8	79	europe	peug
300	23.9	8	260.0	90.0	3420	22.2	79	usa	oldsmo cut sa brougl
301	34.2	4	105.0	70.0	2200	13.2	79	usa	plymo hori
302	34.5	4	105.0	70.0	2150	14.9	79	usa	plymo horizon
303	31.8	4	85.0	65.0	2020	19.2	79	japan	datsun
304	37.3	4	91.0	69.0	2130	14.7	79	europe	fiat str cus
305	28.4	4	151.0	90.0	2670	16.0	79	usa	bı sky lim
306	28.8	6	173.0	115.0	2595	11.3	79	usa	chevr cita
307	26.8	6	173.0	115.0	2700	12.9	79	usa	oldsmo om brougl
308	33.5	4	151.0	90.0	2556	13.2	79	usa	pon pho
309	41.5	4	98.0	76.0	2144	14.7	80	europe	vw ral
310	38.1	4	89.0	60.0	1968	18.8	80	japan	toy cor te
311	32.1	4	98.0	70.0	2120	15.5	80	usa	chevr chev
312	37.2	4	86.0	65.0	2019	16.4	80	japan	datsun

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
313	28.0	4	151.0	90.0	2678	16.5	80	usa	chevr cita
314	26.4	4	140.0	88.0	2870	18.1	80	usa	f fairm
315	24.3	4	151.0	90.0	3003	20.1	80	usa	a conc
316	19.1	6	225.0	90.0	3381	18.7	80	usa	do as
317	34.3	4	97.0	78.0	2188	15.8	80	europe	audi 4
318	29.8	4	134.0	90.0	2711	15.5	80	japan	toy cor liftb
319	31.3	4	120.0	75.0	2542	17.5	80	japan	mazda
320	37.0	4	119.0	92.0	2434	15.0	80	japan	datsun hatchb
321	32.2	4	108.0	75.0	2265	15.2	80	japan	toy cor
322	46.6	4	86.0	65.0	2110	17.9	80	japan	mazda
323	27.9	4	156.0	105.0	2800	14.4	80	usa	dodge
324	40.8	4	85.0	65.0	2110	19.2	80	japan	datsun
325	44.3	4	90.0	48.0	2085	21.7	80	europe	vw rabb (die
326	43.4	4	90.0	48.0	2335	23.7	80	europe	vw das (die
327	36.4	5	121.0	67.0	2950	19.9	80	europe	audi 50 (die
328	30.0	4	146.0	67.0	3250	21.8	80	europe	merced benz 2
329	44.6	4	91.0	67.0	1850	13.8	80	japan	honda c 150
330	40.9	4	85.0	NaN	1835	17.3	80	europe	ren lecar del
331	33.8	4	97.0	67.0	2145	18.0	80	japan	subar
332	29.8	4	89.0	62.0	1845	15.3	80	europe	vokswa ral
333	32.7	6	168.0	132.0	2910	11.4	80	japan	datsun 2
334	23.7	3	70.0	100.0	2420	12.5	80	japan	mazda
335	35.0	4	122.0	88.0	2500	15.1	80	europe	triumph co
336	23.6	4	140.0	NaN	2905	14.3	80	usa	f must cc

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
337	32.4	4	107.0	72.0	2290	17.0	80	japan	ho acc
338	27.2	4	135.0	84.0	2490	15.7	81	usa	plymc rel
339	26.6	4	151.0	84.0	2635	16.4	81	usa	b sky
340	25.8	4	156.0	92.0	2620	14.4	81	usa	dodge a wagon
341	23.5	6	173.0	110.0	2725	12.6	81	usa	chevr cita
342	30.0	4	135.0	84.0	2385	12.9	81	usa	plymc rel
343	39.1	4	79.0	58.0	1755	16.9	81	japan	toy sta
344	39.0	4	86.0	64.0	1875	16.4	81	usa	plymc cha
345	35.1	4	81.0	60.0	1760	16.1	81	japan	honda 1
346	32.3	4	97.0	67.0	2065	17.8	81	japan	sub
347	37.0	4	85.0	65.0	1975	19.4	81	japan	datsun r
348	37.7	4	89.0	62.0	2050	17.3	81	japan	toy te
349	34.1	4	91.0	68.0	1985	16.0	81	japan	mazda g
350	34.7	4	105.0	63.0	2215	14.9	81	usa	plymc horizo
351	34.4	4	98.0	65.0	2045	16.2	81	usa	ford es
352	29.9	4	98.0	65.0	2380	20.7	81	usa	ford es
353	33.0	4	105.0	74.0	2190	14.2	81	europe	volkswa j
354	34.5	4	100.0	NaN	2320	15.8	81	europe	renault
355	33.7	4	107.0	75.0	2210	14.4	81	japan	ho prel
356	32.4	4	108.0	75.0	2350	16.8	81	japan	toy cor
357	32.9	4	119.0	100.0	2615	14.8	81	japan	dat 20
358	31.6	4	120.0	74.0	2635	18.3	81	japan	mazda
359	28.1	4	141.0	80.0	3230	20.4	81	europe	peug 505s tu di
360	30.7	6	145.0	76.0	3160	19.6	81	europe	volvo di

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
361	25.4	6	168.0	116.0	2900	12.6	81	japan	toy cres
362	24.2	6	146.0	120.0	2930	13.8	81	japan	datsun max
363	22.4	6	231.0	110.0	3415	15.8	81	usa	b cen
364	26.6	8	350.0	105.0	3725	19.0	81	usa	oldsmo cutla
365	20.2	6	200.0	88.0	3060	17.1	81	usa	f granad
366	17.6	6	225.0	85.0	3465	16.6	81	usa	chry leba sa
367	28.0	4	112.0	88.0	2605	19.6	82	usa	chevr cava
368	27.0	4	112.0	88.0	2640	18.6	82	usa	chevr cava wa
369	34.0	4	112.0	88.0	2395	18.0	82	usa	chevr cavalie d
370	31.0	4	112.0	85.0	2575	16.2	82	usa	pon j200 hatchb
371	29.0	4	135.0	84.0	2525	16.0	82	usa	dodge a
372	27.0	4	151.0	90.0	2735	18.0	82	usa	pon pho
373	24.0	4	140.0	92.0	2865	16.4	82	usa	f fairm fu
374	23.0	4	151.0	NaN	3035	20.5	82	usa	a concor
375	36.0	4	105.0	74.0	1980	15.3	82	europe	volkswa rab
376	37.0	4	91.0	68.0	2025	18.2	82	japan	mazda custo
377	31.0	4	91.0	68.0	1970	17.6	82	japan	mazda cus
378	38.0	4	105.0	63.0	2125	14.7	82	usa	plymo hori m
379	36.0	4	98.0	70.0	2125	17.3	82	usa	merc ly
380	36.0	4	120.0	88.0	2160	14.5	82	japan	nis stanz

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	na
381	36.0	4	107.0	75.0	2205	14.5	82	japan	ho acc
382	34.0	4	108.0	70.0	2245	16.9	82	japan	toy cor
383	38.0	4	91.0	67.0	1965	15.0	82	japan	honda c
384	32.0	4	91.0	67.0	1965	15.7	82	japan	honda c (a
385	38.0	4	91.0	67.0	1995	16.2	82	japan	datsun
386	25.0	6	181.0	110.0	2945	16.4	82	usa	b cen lim
387	38.0	6	262.0	85.0	3015	17.0	82	usa	oldsmo cutlass c (die
388	26.0	4	156.0	92.0	2585	14.5	82	usa	chry leba medal
389	22.0	6	232.0	112.0	2835	14.7	82	usa	f grana
390	32.0	4	144.0	96.0	2665	13.9	82	japan	toy celic
391	36.0	4	135.0	84.0	2370	13.0	82	usa	do charger
392	27.0	4	151.0	90.0	2950	17.3	82	usa	chevr cam
393	27.0	4	140.0	86.0	2790	15.6	82	usa	f mustan
394	44.0	4	97.0	52.0	2130	24.6	82	europe	vw pic
395	32.0	4	135.0	84.0	2295	11.6	82	usa	do ramp
396	28.0	4	120.0	79.0	2625	18.6	82	usa	ford rar
397	31.0	4	119.0	82.0	2720	19.4	82	usa	chevy s

Problem 6: How many missing values.

In [9]:

```
car.isna().sum()
```

Out[9]:

```
mpg             0
cylinders       0
displacement    0
horsepower      6
weight          0
acceleration    0
model_year      0
origin          0
name            0
dtype: int64
```

Problem 7: Drop all missing values

In [10]:
```
car=car.dropna()
```

In [11]:
```
car.isna().sum()
```

Out[11]:
```
mpg             0
cylinders       0
displacement    0
horsepower      0
weight          0
acceleration    0
model_year      0
origin          0
name            0
dtype: int64
```

Problem 8: Summary statistics

In [12]:
```
car.describe()
```

Out[12]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year
count	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000	392.000000
mean	23.445918	5.471939	194.411990	104.469388	2977.584184	15.541327	75.979592
std	7.805007	1.705783	104.644004	38.491160	849.402560	2.758864	3.683737
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000
25%	17.000000	4.000000	105.000000	75.000000	2225.250000	13.775000	73.000000
50%	22.750000	4.000000	151.000000	93.500000	2803.500000	15.500000	76.000000
75%	29.000000	8.000000	275.750000	126.000000	3614.750000	17.025000	79.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000

Problem 9: Data Type of each column

In [13]:
```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 392 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   mpg           392 non-null    float64
 1   cylinders     392 non-null    int64  
 2   displacement  392 non-null    float64
 3   horsepower    392 non-null    float64
 4   weight        392 non-null    int64  
 5   acceleration  392 non-null    float64
 6   model_year    392 non-null    int64  
 7   origin        392 non-null    object 
 8   name          392 non-null    object 
dtypes: float64(4), int64(3), object(2)
memory usage: 30.6+ KB
```

Problem 10: Shape of Dataframe

In [14]:
```
car.shape
```

Out[14]:
```
(392, 9)
```

In [ ]:

Analysing Columns|Dataframe|Pandas|Python|Solved Exercise Practice Exercise

In [1]:
```
import pandas as pd
```

Problem 1: Import MPG Dataset and store as the pandas dataframe with name mpg

In [2]:
```
mpg=pd.read_csv('https://github.com/YBI-Foundation/Dataset/raw/main/MPG.csv')
```

In [3]:
```
mpg
```

Out[3]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	nam
0	18.0	8	307.0	130.0	3504	12.0	70	usa	chevrole chevell malib
1	15.0	8	350.0	165.0	3693	11.5	70	usa	buic skylar 32
2	18.0	8	318.0	150.0	3436	11.0	70	usa	plymout satellit
3	16.0	8	304.0	150.0	3433	12.0	70	usa	am rebel ss
4	17.0	8	302.0	140.0	3449	10.5	70	usa	for torin
...	...	...	...	...	...	...	...	...	.
393	27.0	4	140.0	86.0	2790	15.6	82	usa	for mustan g
394	44.0	4	97.0	52.0	2130	24.6	82	europe	vv picku
395	32.0	4	135.0	84.0	2295	11.6	82	usa	dodg rampag
396	28.0	4	120.0	79.0	2625	18.6	82	usa	for range
397	31.0	4	119.0	82.0	2720	19.4	82	usa	chevy s 1

398 rows × 9 columns

Problem 2: Copy mpg datset as car

In [4]:
```
car=mpg.copy()
```

In [5]:
```
car
```

Out[5]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	nam
0	18.0	8	307.0	130.0	3504	12.0	70	usa	chevrole chevell malib
1	15.0	8	350.0	165.0	3693	11.5	70	usa	buic skylar 32
2	18.0	8	318.0	150.0	3436	11.0	70	usa	plymout satellit
3	16.0	8	304.0	150.0	3433	12.0	70	usa	am rebel ss
4	17.0	8	302.0	140.0	3449	10.5	70	usa	for torin
...	...	...	...	...	...	...	...	...	.
393	27.0	4	140.0	86.0	2790	15.6	82	usa	for mustan
394	44.0	4	97.0	52.0	2130	24.6	82	europe	vv picku
395	32.0	4	135.0	84.0	2295	11.6	82	usa	dodg rampag
396	28.0	4	120.0	79.0	2625	18.6	82	usa	for range
397	31.0	4	119.0	82.0	2720	19.4	82	usa	chevy s 1

398 rows × 9 columns

Problem 3: Drop column name cylinders from original dataframe(mpg) and inspect what happens to copy(car)

In [7]:

```
mpg=mpg.drop('cylinders',axis=1)
```

In [8]:

```
mpg.columns
```

Out[8]:

```
Index(['mpg', 'displacement', 'horsepower', 'weight', 'acceleration',
       'model_year', 'origin', 'name'],
      dtype='object')
```

In [9]:

```
car.columns
```

Out[9]:

```
Index(['mpg', 'cylinders', 'displacement', 'horsepower', 'weight',
       'acceleration', 'model_year', 'origin', 'name'],
      dtype='object')
```

Problem 4: Analyze dataframe car

In [10]:

```
car.info()
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 398 entries, 0 to 397
Data columns (total 9 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   mpg           398 non-null    float64
 1   cylinders     398 non-null    int64  
 2   displacement  398 non-null    float64
 3   horsepower    392 non-null    float64
 4   weight        398 non-null    int64  
 5   acceleration  398 non-null    float64
 6   model_year    398 non-null    int64  
 7   origin        398 non-null    object 
 8   name          398 non-null    object 
dtypes: float64(4), int64(3), object(2)
memory usage: 28.1+ KB
```

In [11]:

```
car.describe()
```

Out[11]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year
count	398.000000	398.000000	398.000000	392.000000	398.000000	398.000000	398.000000
mean	23.514573	5.454774	193.425879	104.469388	2970.424623	15.568090	76.010050
std	7.815984	1.701004	104.269838	38.491160	846.841774	2.757689	3.697627
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000
25%	17.500000	4.000000	104.250000	75.000000	2223.750000	13.825000	73.000000
50%	23.000000	4.000000	148.500000	93.500000	2803.500000	15.500000	76.000000
75%	29.000000	8.000000	262.000000	126.000000	3608.000000	17.175000	79.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000

Problem 5: Provide unique values in each columns cylinders and origin

In [12]:

```
car[['cylinders','origin']].value_counts()
```

Out[12]:

```
cylinders  origin
8          usa       103
6          usa        74
4          usa        72
           japan      69
           europe     63
6          japan       6
3          japan       4
6          europe      4
5          europe      3
dtype: int64
```

Problem 6: Provide unique values of column origin

In [13]:

```
car[['origin']].value_counts()
```

Out[13]:

```
origin
usa       249
japan      79
europe     70
dtype: int64
```

In [14]:
```
car['origin'].unique()
```

Out[14]:
```
array(['usa', 'japan', 'europe'], dtype=object)
```

In [15]:
```
car['origin'].nunique()
```

Out[15]:
```
3
```

Problem 7: Sort value of car dataframe as per displacement column

In [16]:
```
car.sort_values('displacement')
```

Out[16]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	nan
117	29.0	4	68.0	49.0	1867	19.5	73	europe	fiat 1:
71	19.0	3	70.0	97.0	2330	13.5	72	japan	mazda r: coup
111	18.0	3	70.0	90.0	2124	13.5	73	japan	maxda r:
334	23.7	3	70.0	100.0	2420	12.5	80	japan	mazda r 7
131	32.0	4	71.0	65.0	1836	21.0	74	japan	toyo coro 12(
...	...	...	...	...	...	...	...	...	
94	13.0	8	440.0	215.0	4735	11.0	73	usa	chrysl ne york brougha
6	14.0	8	454.0	220.0	4354	9.0	70	usa	chevrol impa
95	12.0	8	455.0	225.0	4951	11.0	73	usa	bui elect 2: custo
8	14.0	8	455.0	225.0	4425	10.0	70	usa	ponti catali
13	14.0	8	455.0	225.0	3086	10.0	70	usa	bui esta wagc (s

398 rows × 9 columns

Problem 8: Sort value of car dataframe as per displacement column in descending order.

In [17]:
```
car.sort_values('displacement',ascending=False)
```

Out[17]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	nam

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	nam
8	14.0	8	455.0	225.0	4425	10.0	70	usa	pontia catalin
95	12.0	8	455.0	225.0	4951	11.0	73	usa	buic electr 22 custor
13	14.0	8	455.0	225.0	3086	10.0	70	usa	buic estat wago (sw
6	14.0	8	454.0	220.0	4354	9.0	70	usa	chevrole impal
7	14.0	8	440.0	215.0	4312	8.5	70	usa	plymout fury i
...	...	...	...	...	...	...	...	...	.
131	32.0	4	71.0	65.0	1836	21.0	74	japan	toyot coroll 120
111	18.0	3	70.0	90.0	2124	13.5	73	japan	maxd rx
71	19.0	3	70.0	97.0	2330	13.5	72	japan	mazd rx coup
334	23.7	3	70.0	100.0	2420	12.5	80	japan	mazd rx-7 g
117	29.0	4	68.0	49.0	1867	19.5	73	europe	fiat 12

398 rows × 9 columns

Problem 9: Sort value of car dataframe as per displacement and weight columns in descending order

In [18]:
```
car.sort_values(['displacement','weight'],ascending=False)
```

Out[18]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	nan
95	12.0	8	455.0	225.0	4951	11.0	73	usa	bui elect 2: custo
8	14.0	8	455.0	225.0	4425	10.0	70	usa	ponti catalii
13	14.0	8	455.0	225.0	3086	10.0	70	usa	bui esta wago (s
6	14.0	8	454.0	220.0	4354	9.0	70	usa	chevrol impa

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year	origin	nan
94	13.0	8	440.0	215.0	4735	11.0	73	usa	chrysl ne york brougha
...	...	...	...	...	...	...	...	...	
53	31.0	4	71.0	65.0	1773	19.0	71	japan	toyo coro 12(
334	23.7	3	70.0	100.0	2420	12.5	80	japan	mazda r 7
71	19.0	3	70.0	97.0	2330	13.5	72	japan	mazda r: cou|
111	18.0	3	70.0	90.0	2124	13.5	73	japan	maxda r:
117	29.0	4	68.0	49.0	1867	19.5	73	europe	fiat 1:

398 rows × 9 columns

Problem 10: Summary status of all columns

In [19]:

```
car.describe(include='all')
```

Out[19]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model_year
count	398.000000	398.000000	398.000000	392.000000	398.000000	398.000000	398.000000
unique	NaN	NaN	NaN	NaN	NaN	NaN	NaN
top	NaN	NaN	NaN	NaN	NaN	NaN	NaN
freq	NaN	NaN	NaN	NaN	NaN	NaN	NaN
mean	23.514573	5.454774	193.425879	104.469388	2970.424623	15.568090	76.010050
std	7.815984	1.701004	104.269838	38.491160	846.841774	2.757689	3.697627
min	9.000000	3.000000	68.000000	46.000000	1613.000000	8.000000	70.000000
25%	17.500000	4.000000	104.250000	75.000000	2223.750000	13.825000	73.000000
50%	23.000000	4.000000	148.500000	93.500000	2803.500000	15.500000	76.000000
75%	29.000000	8.000000	262.000000	126.000000	3608.000000	17.175000	79.000000
max	46.600000	8.000000	455.000000	230.000000	5140.000000	24.800000	82.000000

Problem 11: Transpose dataframe

In [20]:

```
car.T
```

Out[20]:

	0	1	2	3	4	5	6	7	8
mpg	18.0	15.0	18.0	16.0	17.0	15.0	14.0	14.0	14.0

	0	1	2	3	4	5	6	7	8	
cylinders	8	8	8	8	8	8	8	8	8	
displacement	307.0	350.0	318.0	304.0	302.0	429.0	454.0	440.0	455.0	
horsepower	130.0	165.0	150.0	150.0	140.0	198.0	220.0	215.0	225.0	
weight	3504	3693	3436	3433	3449	4341	4354	4312	4425	
acceleration	12.0	11.5	11.0	12.0	10.5	10.0	9.0	8.5	10.0	
model_year	70	70	70	70	70	70	70	70	70	
origin	usa	usa	usa	usa	usa	usa	usa	usa	usa	
name	chevrolet chevelle malibu	buick skylark 320	plymouth satellite	amc rebel sst	ford torino	ford galaxie 500	chevrolet impala	plymouth fury iii	pontiac catalina	ambas

9 rows × 398 columns

```
In [ ]:
```

Indexing & Slicing|Dataframe|Pandas|Python|Solved Exercise Practice Exercise

In [1]:

```
import pandas as pd
```

Problem 1: Import Titanic dataset and store as the pandas dataframe with name titanic

In [2]:

```
titanic= pd.read_csv('https://github.com/YBI-Foundation/Dataset/raw/main/Titanic.csv
```

Problem 2: Print info of titanic

In [4]:

```
titanic.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1309 entries, 0 to 1308
Data columns (total 14 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   pclass     1309 non-null   int64  
 1   survived   1309 non-null   int64  
 2   name       1309 non-null   object 
 3   sex        1309 non-null   object 
 4   age        1046 non-null   float64
 5   sibsp      1309 non-null   int64  
 6   parch      1309 non-null   int64  
 7   ticket     1309 non-null   object 
 8   fare       1308 non-null   float64
 9   cabin      295 non-null    object 
 10  embarked   1307 non-null   object 
 11  boat       486 non-null    object 
 12  body       121 non-null    float64
 13  home.dest  745 non-null    object 
dtypes: float64(3), int64(4), object(7)
memory usage: 143.3+ KB
```

Problem 3: Print column labels

In [5]:

```
titanic.columns
```

Out[5]:

```
Index(['pclass', 'survived', 'name', 'sex', 'age', 'sibsp', 'parch', 'ticket',
       'fare', 'cabin', 'embarked', 'boat', 'body', 'home.dest'],
      dtype='object')
```

Problem 4: Select passengers name column

In [6]:

```
titanic.name
```

Out[6]:

```
0                           Allen, Miss. Elisabeth Walton
1                          Allison, Master. Hudson Trevor
2                            Allison, Miss. Helen Loraine
3                    Allison, Mr. Hudson Joshua Creighton
4         Allison, Mrs. Hudson J C (Bessie Waldo Daniels)
                              ...                        
1304                                 Zabour, Miss. Hileni
1305                                Zabour, Miss. Thamine
1306                            Zakarian, Mr. Mapriededer
```

```
1307                             Zakarian, Mr. Ortin
1308                              Zimmerman, Mr. Leo
Name: name, Length: 1309, dtype: object
```

In [7]:
```
type(titanic.name)
```

Out[7]:
```
pandas.core.series.Series
```

Problem 5: Select passengers name column as pandas series and save an name

In [8]:
```
titanic['name']
```

Out[8]:
```
0                          Allen, Miss. Elisabeth Walton
1                         Allison, Master. Hudson Trevor
2                           Allison, Miss. Helen Loraine
3                   Allison, Mr. Hudson Joshua Creighton
4        Allison, Mrs. Hudson J C (Bessie Waldo Daniels)
                              ...
1304                                Zabour, Miss. Hileni
1305                               Zabour, Miss. Thamine
1306                           Zakarian, Mr. Mapriededer
1307                                 Zakarian, Mr. Ortin
1308                                  Zimmerman, Mr. Leo
Name: name, Length: 1309, dtype: object
```

In [9]:
```
name=titanic['name']
```

In [10]:
```
name
```

Out[10]:
```
0                          Allen, Miss. Elisabeth Walton
1                         Allison, Master. Hudson Trevor
2                           Allison, Miss. Helen Loraine
3                   Allison, Mr. Hudson Joshua Creighton
4        Allison, Mrs. Hudson J C (Bessie Waldo Daniels)
                              ...
1304                                Zabour, Miss. Hileni
1305                               Zabour, Miss. Thamine
1306                           Zakarian, Mr. Mapriededer
1307                                 Zakarian, Mr. Ortin
1308                                  Zimmerman, Mr. Leo
Name: name, Length: 1309, dtype: object
```

In [11]:
```
type(name)
```

Out[11]:
```
pandas.core.series.Series
```

In [12]:
```
name.shape
```

Out[12]:
```
(1309,)
```

Problem 6: Select passengers name column and save as pandas dataframe

In [13]:
```
name=titanic[['name']]
```

In [14]:
```
name
```

Out[14]:

	name
0	Allen, Miss. Elisabeth Walton
1	Allison, Master. Hudson Trevor
2	Allison, Miss. Helen Loraine
3	Allison, Mr. Hudson Joshua Creighton
4	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)
...	...
1304	Zabour, Miss. Hileni
1305	Zabour, Miss. Thamine
1306	Zakarian, Mr. Mapriededer
1307	Zakarian, Mr. Ortin
1308	Zimmerman, Mr. Leo

1309 rows × 1 columns

In [15]:
```
type(name)
```

Out[15]:
```
pandas.core.frame.DataFrame
```

In [16]:
```
name.shape
```

Out[16]:
```
(1309, 1)
```

Problem 7: Select 100th row and all columns with iloc function

In [17]:
```
titanic.iloc[100,:]
```

Out[17]:
```
pclass                                                1
survived                                              1
name         Duff Gordon, Sir. Cosmo Edmund ("Mr Morgan")
sex                                                male
age                                                49.0
sibsp                                                 1
parch                                                 0
ticket                                         PC 17485
fare                                            56.9292
cabin                                               A20
embarked                                              C
boat                                                  1
body                                                NaN
home.dest                                London / Paris
Name: 100, dtype: object
```

Problem 8: Select 100tn row with loc function

In [18]:
```
titanic.loc[100,:]
```

```
pclass                                                1
```

Out[18]:
```
survived                                        1
name         Duff Gordon, Sir. Cosmo Edmund ("Mr Morgan")
sex                                          male
age                                          49.0
sibsp                                           1
parch                                           0
ticket                                   PC 17485
fare                                      56.9292
cabin                                         A20
embarked                                        C
boat                                            1
body                                          NaN
home.dest                          London / Paris
Name: 100, dtype: object
```

Problem 9: Select all rows with column label name and fare column with iloc function

In [19]:
```python
titanic.iloc[:,[2,8]]
```

Out[19]:

	name	fare
0	Allen, Miss. Elisabeth Walton	211.3375
1	Allison, Master. Hudson Trevor	151.5500
2	Allison, Miss. Helen Loraine	151.5500
3	Allison, Mr. Hudson Joshua Creighton	151.5500
4	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	151.5500
...	...	...
1304	Zabour, Miss. Hileni	14.4542
1305	Zabour, Miss. Thamine	14.4542
1306	Zakarian, Mr. Mapriededer	7.2250
1307	Zakarian, Mr. Ortin	7.2250
1308	Zimmerman, Mr. Leo	7.8750

1309 rows × 2 columns

Problem 10: Select all rows with loc function and column label name and fare

In [20]:
```python
titanic.loc[:,['name','fare']]
```

Out[20]:

	name	fare
0	Allen, Miss. Elisabeth Walton	211.3375
1	Allison, Master. Hudson Trevor	151.5500
2	Allison, Miss. Helen Loraine	151.5500
3	Allison, Mr. Hudson Joshua Creighton	151.5500
4	Allison, Mrs. Hudson J C (Bessie Waldo Daniels)	151.5500
...	...	...
1304	Zabour, Miss. Hileni	14.4542

	name	fare
1305	Zabour, Miss. Thamine	14.4542
1306	Zakarian, Mr. Mapriededer	7.2250
1307	Zakarian, Mr. Ortin	7.2250
1308	Zimmerman, Mr. Leo	7.8750

1309 rows × 2 columns

Problem 11:Select row number 50th,25th,15th and column label passenger class,fare,age with both loc and iloc function

In [21]:

```
titanic.loc[[50,25,15],['pclass','fare','age']]
```

Out[21]:

	pclass	fare	age
50	1	512.3292	58.0
25	1	26.0000	25.0
15	1	25.9250	NaN

In [22]:

```
titanic.iloc[[50,25,15],[0,8,4]]
```

Out[22]:

	pclass	fare	age
50	1	512.3292	58.0
25	1	26.0000	25.0
15	1	25.9250	NaN

Problem 12: Select rows from 10th,25th and column label passenger class,fare,age with both loc and iloc function

In [23]:

```
titanic.loc[10:25,['pclass','fare','age']]
```

Out[23]:

	pclass	fare	age
10	1	227.5250	47.0
11	1	227.5250	18.0
12	1	69.3000	24.0
13	1	78.8500	26.0
14	1	30.0000	80.0
15	1	25.9250	NaN
16	1	247.5208	24.0
17	1	247.5208	50.0
18	1	76.2917	32.0
19	1	75.2417	36.0

	pclass	fare	age
20	1	52.5542	37.0
21	1	52.5542	47.0
22	1	30.0000	26.0
23	1	227.5250	42.0
24	1	221.7792	29.0
25	1	26.0000	25.0

In [24]:

```
titanic.iloc[10:26,[0,8,4]]
```

Out[24]:

	pclass	fare	age
10	1	227.5250	47.0
11	1	227.5250	18.0
12	1	69.3000	24.0
13	1	78.8500	26.0
14	1	30.0000	80.0
15	1	25.9250	NaN
16	1	247.5208	24.0
17	1	247.5208	50.0
18	1	76.2917	32.0
19	1	75.2417	36.0
20	1	52.5542	37.0
21	1	52.5542	47.0
22	1	30.0000	26.0
23	1	227.5250	42.0
24	1	221.7792	29.0
25	1	26.0000	25.0

Probelm 13: Select rows from 10th to 15th columns from passenger class to age with both loc and iloc function

In [25]:

```
titanic.loc[10:15,'pclass':'age']
```

Out[25]:

	pclass	survived	name	sex	age
10	1	0	Astor, Col. John Jacob	male	47.0
11	1	1	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	female	18.0
12	1	1	Aubart, Mme. Leontine Pauline	female	24.0
13	1	1	Barber, Miss. Ellen "Nellie"	female	26.0
14	1	1	Barkworth, Mr. Algernon Henry Wilson	male	80.0

	pclass	survived	name	sex	age
15	1	0	Baumann, Mr. John D	male	NaN

In [26]:
```
titanic.iloc[10:16,0:5]
```

Out[26]:

	pclass	survived	name	sex	age
10	1	0	Astor, Col. John Jacob	male	47.0
11	1	1	Astor, Mrs. John Jacob (Madeleine Talmadge Force)	female	18.0
12	1	1	Aubart, Mme. Leontine Pauline	female	24.0
13	1	1	Barber, Miss. Ellen "Nellie"	female	26.0
14	1	1	Barkworth, Mr. Algernon Henry Wilson	male	80.0
15	1	0	Baumann, Mr. John D	male	NaN

Problem 14: Select all passengers with age equal to and more than 35 years

In [27]:
```
titanic[titanic['age']>=35]
```

Out[27]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked
5	1	1	Anderson, Mr. Harry	male	48.0	0	0	19952	26.5500	E12	S
6	1	1	Andrews, Miss. Kornelia Theodosia	female	63.0	1	0	13502	77.9583	D7	S
7	1	0	Andrews, Mr. Thomas Jr	male	39.0	0	0	112050	0.0000	A36	S
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	53.0	2	0	11769	51.4792	C101	S
9	1	0	Artagaveytia, Mr. Ramon	male	71.0	0	0	PC 17609	49.5042	NaN	C
...	...	...	...	...	...	...	...	...	...	...	...
1286	3	1	Whabee, Mrs. George Joseph (Shawneene Abi-Saab)	female	38.0	0	0	2688	7.2292	NaN	C
1287	3	0	Widegren, Mr. Carl/Charles Peter	male	51.0	0	0	347064	7.7500	NaN	S
1290	3	1	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarked
1298	3	0	Wittevrongel, Mr. Camille	male	36.0	0	0	345771	9.5000	NaN	S
1301	3	0	Youseff, Mr. Gerious	male	45.5	0	0	2628	7.2250	NaN	C

345 rows × 14 columns

Problem 15: Select all passengers with age equal to and more than 35 years and column with label passenger class to age

In [28]:

```
titanic.loc[(titanic['age']>=35),'pclass':'age']
```

Out[28]:

	pclass	survived	name	sex	age
5	1	1	Anderson, Mr. Harry	male	48.0
6	1	1	Andrews, Miss. Kornelia Theodosia	female	63.0
7	1	0	Andrews, Mr. Thomas Jr	male	39.0
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	53.0
9	1	0	Artagaveytia, Mr. Ramon	male	71.0
...	...	...	...	...	...
1286	3	1	Whabee, Mrs. George Joseph (Shawneene Abi-Saab)	female	38.0
1287	3	0	Widegren, Mr. Carl/Charles Peter	male	51.0
1290	3	1	Wilkes, Mrs. James (Ellen Needs)	female	47.0
1298	3	0	Wittevrongel, Mr. Camille	male	36.0
1301	3	0	Youseff, Mr. Gerious	male	45.5

345 rows × 5 columns

Problem 16: Select all female passengers with age equal to or more than 35 years

In [29]:

```
titanic.loc[(titanic['age']>=35) & (titanic['sex']=='female')]
```

Out[29]:

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarke
6	1	1	Andrews, Miss. Kornelia Theodosia	female	63.0	1	0	13502	77.9583	D7	
8	1	1	Appleton, Mrs. Edward Dale (Charlotte Lamson)	female	53.0	2	0	11769	51.4792	C101	

	pclass	survived	name	sex	age	sibsp	parch	ticket	fare	cabin	embarke
17	1	1	Baxter, Mrs. James (Helene DeLaudeniere Chaput)	female	50.0	0	1	PC 17558	247.5208	B58 B60	
21	1	1	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	female	47.0	1	1	11751	52.5542	D35	
23	1	1	Bidois, Miss. Rosalie	female	42.0	0	0	PC 17757	227.5250	NaN	
...	...	...	...	...	...	...	...	...	...	...	
1158	3	0	Rosblom, Mrs. Viktor (Helena Wilhelmina)	female	41.0	0	2	370129	20.2125	NaN	
1211	3	0	Skoog, Mrs. William (Anna Bernhardina Karlsson)	female	45.0	1	4	347088	27.9000	NaN	
1261	3	1	Turkula, Mrs. (Hedwig)	female	63.0	0	0	4134	9.5875	NaN	
1286	3	1	Whabee, Mrs. George Joseph (Shawneene Abi-Saab)	female	38.0	0	0	2688	7.2292	NaN	
1290	3	1	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	

125 rows × 14 columns

In [ ]:

Calculated Columns|Dataframe|Pandas|Python|Solved Exercise Practice Exercise

In [1]:
```
import pandas as pd
```

Problem 1: Import Tips dataset and store as the pandas dataframe with name tips

In [2]:
```
tips=pd.read_csv('https://github.com/YBI-Foundation/Dataset/raw/main/Tips%20Payment%
```

Problem 2: Display the first 5 rows of tips dataframe

In [3]:
```
tips.head()
```

Out[3]:

	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	Payer Name	CC Number	Payme
0	16.99	1.01	Female	No	Sun	Dinner	2	8.49	Christy Cunningham	3560325168603410	Sun29
1	10.34	1.66	Male	No	Sun	Dinner	3	3.45	Douglas Tucker	4478071379779230	Sun46
2	21.01	3.50	Male	No	Sun	Dinner	3	7.00	Travis Walters	6011812112971320	Sun44
3	23.68	3.31	Male	No	Sun	Dinner	2	11.84	Nathaniel Harris	4676137647685990	Sun52
4	24.59	3.61	Female	No	Sun	Dinner	4	6.15	Tonya Carter	4832732618637220	Sun22

Problem 3: Calculate percentage of tip to total bill

In [4]:
```
tips['Tip']/tips['Total Bill']*100
```

Out[4]:
```
0       5.944673
1      16.054159
2      16.658734
3      13.978041
4      14.680765
         ...
239    20.392697
240     7.358352
241     8.822232
242     9.820426
243    15.974441
Length: 244, dtype: float64
```

Problem 4: Create a new column of percentage tip

In [5]:
```
tip_percentage= tips['Tip']/tips['Total Bill']*100
```

In [6]:
```
tip_percentage
```

Out[6]:
```
0       5.944673
1      16.054159
2      16.658734
3      13.978041
4      14.680765
         ...
239    20.392697
240     7.358352
241     8.822232
242     9.820426
243    15.974441
Length: 244, dtype: float64
```

Problem 5: Insert percentage tip in existing tips dataframe

In [7]:
```
tips['tip_percentage']=tips['Tip']/tips['Total Bill']*100
```

In [8]:
```
tips.head()
```

Out[8]:

	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	Payer Name	CC Number	Payme
0	16.99	1.01	Female	No	Sun	Dinner	2	8.49	Christy Cunningham	3560325168603410	Sun29
1	10.34	1.66	Male	No	Sun	Dinner	3	3.45	Douglas Tucker	4478071379779230	Sun46
2	21.01	3.50	Male	No	Sun	Dinner	3	7.00	Travis Walters	6011812112971320	Sun44
3	23.68	3.31	Male	No	Sun	Dinner	2	11.84	Nathaniel Harris	4676137647685990	Sun52
4	24.59	3.61	Female	No	Sun	Dinner	4	6.15	Tonya Carter	4832732618637220	Sun22

Problem 6: Round upto one decimal place the tip_percentage column values

In [9]:
```
tips['tip_percentage']=tips['tip_percentage'].round(1)
```

In [10]:
```
tips.head()
```

Out[10]:

	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	Payer Name	CC Number	Payme
0	16.99	1.01	Female	No	Sun	Dinner	2	8.49	Christy Cunningham	3560325168603410	Sun29
1	10.34	1.66	Male	No	Sun	Dinner	3	3.45	Douglas Tucker	4478071379779230	Sun46
2	21.01	3.50	Male	No	Sun	Dinner	3	7.00	Travis Walters	6011812112971320	Sun44
3	23.68	3.31	Male	No	Sun	Dinner	2	11.84	Nathaniel Harris	4676137647685990	Sun52

	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	Payer Name	CC Number	Payme
4	24.59	3.61	Female	No	Sun	Dinner	4	6.15	Tonya Carter	4832732618637220	Sun22

Problem 7: Drop column payer number

In [11]:
```
tips=tips.drop(['Payer Name'],axis=1)
```

In [12]:
```
tips.head()
```

Out[12]:

	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	CC Number	Payment ID	tip_percer
0	16.99	1.01	Female	No	Sun	Dinner	2	8.49	3560325168603410	Sun2959	
1	10.34	1.66	Male	No	Sun	Dinner	3	3.45	4478071379779230	Sun4608	
2	21.01	3.50	Male	No	Sun	Dinner	3	7.00	6011812112971320	Sun4458	
3	23.68	3.31	Male	No	Sun	Dinner	2	11.84	4676137647685990	Sun5260	
4	24.59	3.61	Female	No	Sun	Dinner	4	6.15	4832732618637220	Sun2251	

Problem 8: Index tips dataframe as per Payment ID

In [13]:
```
tips.set_index('Payment ID')
```

Out[13]:

Payment ID	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	CC Number	tip_percenta
Sun2959	16.99	1.01	Female	No	Sun	Dinner	2	8.49	3560325168603410	5
Sun4608	10.34	1.66	Male	No	Sun	Dinner	3	3.45	4478071379779230	16
Sun4458	21.01	3.50	Male	No	Sun	Dinner	3	7.00	6011812112971320	16
Sun5260	23.68	3.31	Male	No	Sun	Dinner	2	11.84	4676137647685990	14
Sun2251	24.59	3.61	Female	No	Sun	Dinner	4	6.15	4832732618637220	14
...	...	...	...	...	...	...	...	...	...	
Sat2657	29.03	5.92	Male	No	Sat	Dinner	3	9.68	5296068606052840	20
Sat1766	27.18	2.00	Female	Yes	Sat	Dinner	2	13.59	3506806155565400	7
Sat3880	22.67	2.00	Male	Yes	Sat	Dinner	2	11.34	6011891618747190	8
Sat17	17.82	1.75	Male	No	Sat	Dinner	2	8.91	4375220550950	9
Thur672	18.78	3.00	Female	No	Thur	Dinner	2	9.39	3511451626698130	16

244 rows × 10 columns

In [14]:
```
tips.head()
```

Out[14]:

	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	CC Number	Payment ID	tip_percer
0	16.99	1.01	Female	No	Sun	Dinner	2	8.49	3560325168603410	Sun2959	
1	10.34	1.66	Male	No	Sun	Dinner	3	3.45	4478071379779230	Sun4608	
2	21.01	3.50	Male	No	Sun	Dinner	3	7.00	6011812112971320	Sun4458	
3	23.68	3.31	Male	No	Sun	Dinner	2	11.84	4676137647685990	Sun5260	
4	24.59	3.61	Female	No	Sun	Dinner	4	6.15	4832732618637220	Sun2251	

Problem 9 : Change index tips dataframe as per Payment ID

In [15]:
```
tips=tips.set_index('Payment ID')
```

In [16]:
```
tips.head()
```

Out[16]:

	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	CC Number	tip_percentag
Payment ID										
Sun2959	16.99	1.01	Female	No	Sun	Dinner	2	8.49	3560325168603410	5
Sun4608	10.34	1.66	Male	No	Sun	Dinner	3	3.45	4478071379779230	16
Sun4458	21.01	3.50	Male	No	Sun	Dinner	3	7.00	6011812112971320	16
Sun5260	23.68	3.31	Male	No	Sun	Dinner	2	11.84	4676137647685990	14
Sun2251	24.59	3.61	Female	No	Sun	Dinner	4	6.15	4832732618637220	14

Problem 10: Reset index of tips dataframe to row index

In [17]:
```
tips=tips.reset_index()
```

In [18]:
```
tips.head()
```

Out[18]:

	Payment ID	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	CC Number	tip_percer
0	Sun2959	16.99	1.01	Female	No	Sun	Dinner	2	8.49	3560325168603410	
1	Sun4608	10.34	1.66	Male	No	Sun	Dinner	3	3.45	4478071379779230	
2	Sun4458	21.01	3.50	Male	No	Sun	Dinner	3	7.00	6011812112971320	

	Payment ID	Total Bill	Tip	Gender	Smoker	Day	Time	Size	Bill Per Person	CC Number	tip_percer
3	Sun5260	23.68	3.31	Male	No	Sun	Dinner	2	11.84	4676137647685990	
4	Sun2251	24.59	3.61	Female	No	Sun	Dinner	4	6.15	4832732618637220	

In [ ]: