```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
import pandas as pd
df=pd.read_csv('/content/sample_data/california_housing_train.csv')
```

```
df
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	med
0	-114.31	34.19	15.0	5612.0	1283.0	1015.0	472.0	1.4936	
1	-114.47	34.40	19.0	7650.0	1901.0	1129.0	463.0	1.8200	
2	-114.56	33.69	17.0	720.0	174.0	333.0	117.0	1.6509	
3	-114.57	33.64	14.0	1501.0	337.0	515.0	226.0	3.1917	
4	-114.57	33.57	20.0	1454.0	326.0	624.0	262.0	1.9250	
...	...	...	...	...	...	...	...	...	
16995	-124.26	40.58	52.0	2217.0	394.0	907.0	369.0	2.3571	
16996	-124.27	40.69	36.0	2349.0	528.0	1194.0	465.0	2.5179	
16997	-124.30	41.84	17.0	2677.0	531.0	1244.0	456.0	3.0313	
16998	-124.30	41.80	19.0	2672.0	552.0	1298.0	478.0	1.9797	
16999	-124.35	40.54	52.0	1820.0	300.0	806.0	270.0	3.0147	

17000 rows × 9 columns

Next steps: Generate code with df View recommended plots

```
df.head()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
0	-114.31	34.19	15.0	5612.0	1283.0	1015.0	472.0	1.4936	66900.0
1	-114.47	34.40	19.0	7650.0	1901.0	1129.0	463.0	1.8200	80100.0
2	-114.56	33.69	17.0	720.0	174.0	333.0	117.0	1.6509	85700.0
3	-114.57	33.64	14.0	1501.0	337.0	515.0	226.0	3.1917	73400.0
4	-114.57	33.57	20.0	1454.0	326.0	624.0	262.0	1.9250	65500.0

Next steps: Generate code with df View recommended plots

```
df.tail()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
16995	-124.26	40.58	52.0	2217.0	394.0	907.0	369.0	2.3571	111400.0
16996	-124.27	40.69	36.0	2349.0	528.0	1194.0	465.0	2.5179	79000.0
16997	-124.30	41.84	17.0	2677.0	531.0	1244.0	456.0	3.0313	103600.0
16998	-124.30	41.80	19.0	2672.0	552.0	1298.0	478.0	1.9797	85800.0
16999	-124.35	40.54	52.0	1820.0	300.0	806.0	270.0	3.0147	94600.0

1.DROP A ROW THAT CONTAINS THE VALUE

```
df=df.drop(df[df['latitude']==33.78].index)
df
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
0	-114.31	34.19	15.0	5612.0	1283.0	1015.0	472.0	1.4936	66900.0
1	-114.47	34.40	19.0	7650.0	1901.0	1129.0	463.0	1.8200	80100.0
2	-114.56	33.69	17.0	720.0	174.0	333.0	117.0	1.6509	85700.0
3	-114.57	33.64	14.0	1501.0	337.0	515.0	226.0	3.1917	73400.0
4	-114.57	33.57	20.0	1454.0	326.0	624.0	262.0	1.9250	65500.0
...	...	...	...	...	...	...	...	...	...
16995	-124.26	40.58	52.0	2217.0	394.0	907.0	369.0	2.3571	111400.0
16996	-124.27	40.69	36.0	2349.0	528.0	1194.0	465.0	2.5179	79000.0
16997	-124.30	41.84	17.0	2677.0	531.0	1244.0	456.0	3.0313	103600.0
16998	-124.30	41.80	19.0	2672.0	552.0	1298.0	478.0	1.9797	85800.0
16999	-124.35	40.54	52.0	1820.0	300.0	806.0	270.0	3.0147	94600.0

16899 rows × 9 columns

Next steps: Generate code with df View recommended plots

2.DISPLAY A COLUMN

```
df['latitude']
```

```
0        34.19
1        34.40
2        33.69
3        33.64
4        33.57
         ...
16995    40.58
16996    40.69
16997    41.84
16998    41.80
16999    40.54
Name: latitude, Length: 16899, dtype: float64
```

3.DISPLAY FIRST 5 ROWS OF THE COLUMN

```
df['longitude'].head()
```

```
0   -114.31
1   -114.47
2   -114.56
3   -114.57
4   -114.57
Name: longitude, dtype: float64
```

4.replace a record

```
df['longitude']=df['longitude'].replace(-114.31,117.31)
df
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
0	117.31	34.19	15.0	5612.0	1283.0	1015.0	472.0	1.4936	66900.0
1	-114.47	34.40	19.0	7650.0	1901.0	1129.0	463.0	1.8200	80100.0
2	-114.56	33.69	17.0	720.0	174.0	333.0	117.0	1.6509	85700.0
3	-114.57	33.64	14.0	1501.0	337.0	515.0	226.0	3.1917	73400.0
4	-114.57	33.57	20.0	1454.0	326.0	624.0	262.0	1.9250	65500.0
...	...	...	...	...	...	...	...	...	...
16995	-124.26	40.58	52.0	2217.0	394.0	907.0	369.0	2.3571	111400.0
16996	-124.27	40.69	36.0	2349.0	528.0	1194.0	465.0	2.5179	79000.0
16997	-124.30	41.84	17.0	2677.0	531.0	1244.0	456.0	3.0313	103600.0
16998	-124.30	41.80	19.0	2672.0	552.0	1298.0	478.0	1.9797	85800.0
16999	-124.35	40.54	52.0	1820.0	300.0	806.0	270.0	3.0147	94600.0

17000 rows × 9 columns

Next steps: Generate code with df View recommended plots

5.DISPLAY NULL VALUES

```
print("count of null values",df.isnull().sum())
print("sum of null values",df.isnull().sum().sum())
```

```
count of null values longitude             0
latitude              0
housing_median_age    0
total_rooms           0
total_bedrooms        0
population            0
households            0
median_income         0
median_house_value    0
dtype: int64
sum of null values 0
```

6.print outliers

```
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3 - Q1

outliers = df[(df < (Q1 - 1.5 * IQR)) | (df > (Q3 + 1.5 * IQR))]
```

7.FIND ALL THE NA VALUES

```
df.dropna()
df
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
0	120.00	34.19	15.0	5612.0	1283.0	1015.0	472.0	1.4936	66900.0
1	-114.47	34.40	19.0	7650.0	1901.0	1129.0	463.0	1.8200	80100.0
2	-114.56	33.69	17.0	720.0	174.0	333.0	117.0	1.6509	85700.0
3	-114.57	33.64	14.0	1501.0	337.0	515.0	226.0	3.1917	73400.0
4	-114.57	33.57	20.0	1454.0	326.0	624.0	262.0	1.9250	65500.0
...	...	...	...	...	...	...	...	...	...
16995	-124.26	40.58	52.0	2217.0	394.0	907.0	369.0	2.3571	111400.0
16996	-124.27	40.69	36.0	2349.0	528.0	1194.0	465.0	2.5179	79000.0
16997	-124.30	41.84	17.0	2677.0	531.0	1244.0	456.0	3.0313	103600.0
16998	-124.30	41.80	19.0	2672.0	552.0	1298.0	478.0	1.9797	85800.0
16999	-124.35	40.54	52.0	1820.0	300.0	806.0	270.0	3.0147	94600.0

16899 rows × 9 columns

Next steps: Generate code with df View recommended plots

8.SET A COLUMN AS THE INDEX

```
df.set_index('population')
df
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
0	120.00	34.19	15.0	5612.0	1283.0	1015.0	472.0	1.4936	66900.0
1	-114.47	34.40	19.0	7650.0	1901.0	1129.0	463.0	1.8200	80100.0
2	-114.56	33.69	17.0	720.0	174.0	333.0	117.0	1.6509	85700.0
3	-114.57	33.64	14.0	1501.0	337.0	515.0	226.0	3.1917	73400.0
4	-114.57	33.57	20.0	1454.0	326.0	624.0	262.0	1.9250	65500.0
...	...	...	...	...	...	...	...	...	...
16995	-124.26	40.58	52.0	2217.0	394.0	907.0	369.0	2.3571	111400.0
16996	-124.27	40.69	36.0	2349.0	528.0	1194.0	465.0	2.5179	79000.0
16997	-124.30	41.84	17.0	2677.0	531.0	1244.0	456.0	3.0313	103600.0
16998	-124.30	41.80	19.0	2672.0	552.0	1298.0	478.0	1.9797	85800.0
16999	-124.35	40.54	52.0	1820.0	300.0	806.0	270.0	3.0147	94600.0

16899 rows × 9 columns

Next steps: Generate code with df View recommended plots

9.SET GROUP OF COLUMNS AS INDEX

```
df.set_index(['median_income','median_income'])
```

		longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_house_
median_income	median_income								
1.4936	**1.4936**	120.00	34.19	15.0	5612.0	1283.0	1015.0	472.0	66
1.8200	**1.8200**	-114.47	34.40	19.0	7650.0	1901.0	1129.0	463.0	80
1.6509	**1.6509**	-114.56	33.69	17.0	720.0	174.0	333.0	117.0	85
3.1917	**3.1917**	-114.57	33.64	14.0	1501.0	337.0	515.0	226.0	73
1.9250	**1.9250**	-114.57	33.57	20.0	1454.0	326.0	624.0	262.0	65
...	**...**	...	...	...	...	...	...	...	
2.3571	**2.3571**	-124.26	40.58	52.0	2217.0	394.0	907.0	369.0	11
2.5179	**2.5179**	-124.27	40.69	36.0	2349.0	528.0	1194.0	465.0	79
3.0313	**3.0313**	-124.30	41.84	17.0	2677.0	531.0	1244.0	456.0	103
1.9797	**1.9797**	-124.30	41.80	19.0	2672.0	552.0	1298.0	478.0	85
3.0147	**3.0147**	-124.35	40.54	52.0	1820.0	300.0	806.0	270.0	94

16899 rows × 8 columns

10.DISPLAY NULL VALUES

```
df.isnull()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False