```
import pandas as pd
df = pd.DataFrame()


Data ="/content/sample_data/california_housing_test.csv"
df=pd.read_csv(Data)
df.head(2)
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	hou
0	-122.05	37.37	27.0	3885.0	661.0	1537.0	
1	-118.30	34.26	43.0	1510.0	310.0	809.0	

Next steps: Generate code with `df` View recommended plots

```
format(df.shape)
```

```
'(3000, 9)'
```

```
df.describe()
```

	ıl_rooms	total_bedrooms	population	households	median_income	median_house_value
	0.000000	3000.000000	3000.000000	3000.00000	3000.000000	3000.00000
	9.578667	529.950667	1402.798667	489.91200	3.807272	205846.27500
	5.593332	415.654368	1030.543012	365.42271	1.854512	113119.68747
	6.000000	2.000000	5.000000	2.00000	0.499900	22500.00000
	1.000000	291.000000	780.000000	273.00000	2.544000	121200.00000
	6.000000	437.000000	1155.000000	409.50000	3.487150	177650.00000
	9.000000	636.000000	1742.750000	597.25000	4.656475	263975.00000
	0.000000	5419.000000	11935.000000	4930.00000	15.000100	500001.00000

```
print("---------------select the first row \n",df.iloc[0])
print("\n -------------select three row \n",df.iloc[1:4])
print("\n ---------------all rows up to 4 including four \n",df.iloc[:4])
```

```
---------------select the first row
 longitude               -122.0500
latitude                  37.3700
housing_median_age        27.0000
```

```
total_rooms             3885.0000
total_bedrooms           661.0000
population              1537.0000
households               606.0000
median_income              6.6085
median_house_value    344700.0000
Name: 0, dtype: float64

 -------------select three row
    longitude  latitude  housing_median_age  total_rooms  total_bedrooms  \
1    -118.30     34.26                43.0       1510.0           310.0
2    -117.81     33.78                27.0       3589.0           507.0
3    -118.36     33.82                28.0         67.0            15.0

   population  households  median_income  median_house_value
1       809.0       277.0         3.5990            176500.0
2      1484.0       495.0         5.7934            270500.0
3        49.0        11.0         6.1359            330000.0

 ---------------all rows up to 4 including four
    longitude  latitude  housing_median_age  total_rooms  total_bedrooms  \
0    -122.05     37.37                27.0       3885.0           661.0
1    -118.30     34.26                43.0       1510.0           310.0
2    -117.81     33.78                27.0       3589.0           507.0
3    -118.36     33.82                28.0         67.0            15.0

   population  households  median_income  median_house_value
0      1537.0       606.0         6.6085            344700.0
1       809.0       277.0         3.5990            176500.0
2      1484.0       495.0         5.7934            270500.0
3        49.0        11.0         6.1359            330000.0
```

Selecting Rows Based on Conditionals

```
df[df['housing_median_age']==27.0].head(2)
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	populat
longitude						
-122.05	-122.05	37.37	27.0	3885.0	661.0	15
-117.81	-117.81	33.78	27.0	3589.0	507.0	14

```
df['housing_median_age'].replace(27.0,20.1).head(2)
```

```
longitude
-122.05    20.1
-118.30    43.0
Name: housing_median_age, dtype: float64
```

```
df['housing_median_age'].replace([27.0,20.1],[43.0,28.0]).head(2)
```

```
     longitude
     -122.05    43.0
     -118.30    43.0
     Name: housing_median_age, dtype: float64
```

Renaming columns

```
df.rename(columns={'longitude':'diameter'}).head(2)
```

tal_rooms	total_bedrooms	population	households	median_income	median_house_value
3885.0	661.0	1537.0	606.0	6.6085	344700.0
1510.0	310.0	809.0	277.0	3.5990	176500.0

```
print('Maximum: {}'.format(df['housing_median_age'].max()))
print('Minimum: ()'.format(df['housing_median_age'].min()))
print('Mean: {}'.format(df ['housing_median_age'].mean()))
print('Sum: '.format(df ['housing_median_age'].sum()))
print('Count: {}'.format(df ['housing_median_age'].count()))
```

```
     Maximum: 52.0
     Minimum: ()
     Mean: 28.845333333333333
     Sum:
     Count: 3000
```

```
print("Variance: ()\n".format(df.var()))
print("\nStandard Deviation: {}\n".format(df.std()))
print("\nKurtosis: {}\n".format(df.kurt()))
print("\nSkewness:{}\n".format(df.skew()))
```

```
     Variance: ()


     Standard Deviation: longitude                 1.994936
     latitude                    2.129670
     housing_median_age         12.555396
     total_rooms              2155.593332
     total_bedrooms            415.654368
     population               1030.543012
     households                365.422710
     median_income               1.854512
     median_house_value     113119.687470
     dtype: float64
```

```
Kurtosis: longitude             -1.362772
latitude               -1.124372
housing_median_age     -0.803784
total_rooms            32.099941
total_bedrooms         28.537071
population             16.443268
households             26.229361
median_income           5.626184
median_house_value      0.395399
dtype: float64


Skewness:longitude            -0.297858
latitude               0.459816
housing_median_age     0.018513
total_rooms            4.167637
total_bedrooms         3.863393
population             2.949671
households             3.559753
median_income          1.698512
median_house_value     0.989562
dtype: float64
```

```
df['median_income'].unique()
```

```
array([6.6085, 3.599 , 5.7934, ..., 3.3906, 2.2895, 8.5608])
```

```
df.drop('housing_median_age',axis=1).head(2)
```

	longitude	latitude	total_rooms	total_bedrooms	population	households	med
longitude							
-122.05	-122.05	37.37	3885.0	661.0	1537.0	606.0	
-118.30	-118.30	34.26	1510.0	310.0	809.0	277.0	

```
df[df['population']!=310].head(2)
```

tal_rooms	total_bedrooms	population	households	median_income	median_house_value
3885.0	661.0	1537.0	606.0	6.6085	344700.0
1510.0	310.0	809.0	277.0	3.5990	176500.0

Grouping Rows by Values

```
df.groupby('population').mean()
```

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	households	medi
on							
	-114.620	33.62	26.0	18.0	3.0	3.0	
	-117.035	33.26	26.5	11.0	3.0	2.5	
	-117.775	33.35	50.5	28.5	6.0	8.0	
	-122.490	38.00	26.0	48.0	8.0	8.0	
	-118.060	34.03	36.0	21.0	7.0	9.0	
	...	...	...	...	...	...	
	-117.120	33.49	4.0	21988.0	4055.0	3252.0	
	-117.200	33.58	2.0	30450.0	5033.0	3197.0	
	-117.270	33.15	4.0	23915.0	4135.0	3958.0	
	-116.140	34.45	12.0	8796.0	1721.0	1680.0	
	-121.530	38.48	5.0	27870.0	5027.0	4855.0	

‹ 8 columns

```
df.groupby('population')['housing_median_age'].count()
```

```
population
5.0        1
8.0        2
14.0       2
19.0       1
21.0       1
          ..
8824.0     1
9419.0     1
10877.0    1
11139.0    1
11935.0    1
Name: housing_median_age, Length: 1802, dtype: int64
```

```
df.groupby('total_rooms ').apply(lamba x: x.count())
```

Start coding or generate with AI.