Model building

To predict price of the land in some areas

```
1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 import sklearn
```

```
1 df=pd.read_csv("10_USA_Housing.csv")
2 df
```

	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price	Address
0	79545.458574	5.682861	7.009188	4.09	23086.800503	1.059034e+06	208 Michael Ferry Apt. 674\nLaurabury, NE 3701...
1	79248.642455	6.002900	6.730821	3.09	40173.072174	1.505891e+06	188 Johnson Views Suite 079\nLake Kathleen, CA...
2	61287.067179	5.865890	8.512727	5.13	36882.159400	1.058988e+06	9127 Elizabeth Stravenue\nDanieltown, WI 06482...
3	63345.240046	7.188236	5.586729	3.26	34310.242831	1.260617e+06	USS Barnett\nFPO AP 44820
4	59982.197226	5.040555	7.839388	4.23	26354.109472	6.309435e+05	USNS Raymond\nFPO AE 09386
...	...	...	...	...	...	...	...
4995	60567.944140	7.830362	6.137356	3.46	22837.361035	1.060194e+06	USNS Williams\nFPO AP 30153-7653
4996	78491.275435	6.999135	6.576763	4.02	25616.115489	1.482618e+06	PSC 9258, Box 8489\nAPO AA 42991-3352
4997	63390.686886	7.250591	4.805081	2.13	33266.145490	1.030730e+06	4215 Tracy Garden Suite 076\nJoshualand, VA 01...
4998	68001.331235	5.534388	7.130144	5.44	42625.620156	1.198657e+06	USS Wallace\nFPO AE 73316
4999	65510.581804	5.992305	6.792336	4.07	46501.283803	1.298950e+06	37778 George Ridges Apt. 509\nEast Holly, NV 2...

```
1 df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5000 entries, 0 to 4999
Data columns (total 7 columns):
 #   Column                       Non-Null Count  Dtype  
---  ------                       --------------  -----  
 0   Avg. Area Income             5000 non-null   float64
 1   Avg. Area House Age          5000 non-null   float64
 2   Avg. Area Number of Rooms    5000 non-null   float64
 3   Avg. Area Number of Bedrooms 5000 non-null   float64
 4   Area Population              5000 non-null   float64
 5   Price                        5000 non-null   float64
 6   Address                      5000 non-null   object 
dtypes: float64(6), object(1)
memory usage: 273.6+ KB
```

```
1 df.describe()
```

	Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price
count	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5.000000e+03
mean	68583.108984	5.977222	6.987792	3.981330	36163.516039	1.232073e+06
std	10657.991214	0.991456	1.005833	1.234137	9925.650114	3.531176e+05
min	17796.631190	2.644304	3.236194	2.000000	172.610686	1.593866e+04
25%	61480.562388	5.322283	6.299250	3.140000	29403.928702	9.975771e+05
50%	68804.286404	5.970429	7.002902	4.050000	36199.406689	1.232669e+06
75%	75783.338666	6.650808	7.665871	4.490000	42861.290769	1.471210e+06
max	107701.748378	9.519088	10.759588	6.500000	69621.713378	2.469066e+06

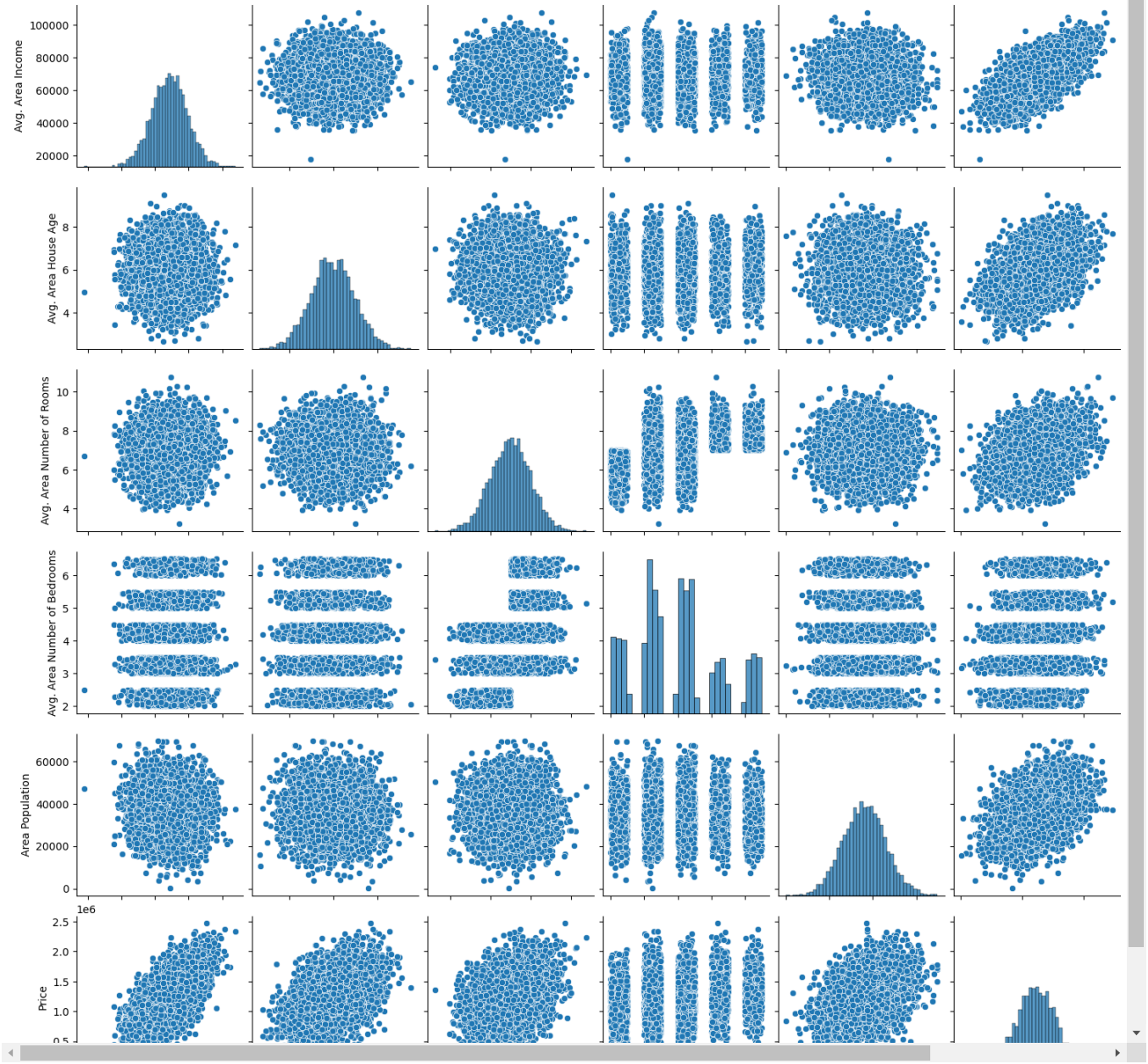
```
1 df.columns
```

```
Index(['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms',
       'Avg. Area Number of Bedrooms', 'Area Population', 'Price', 'Address'],
      dtype='object')
```

EDA

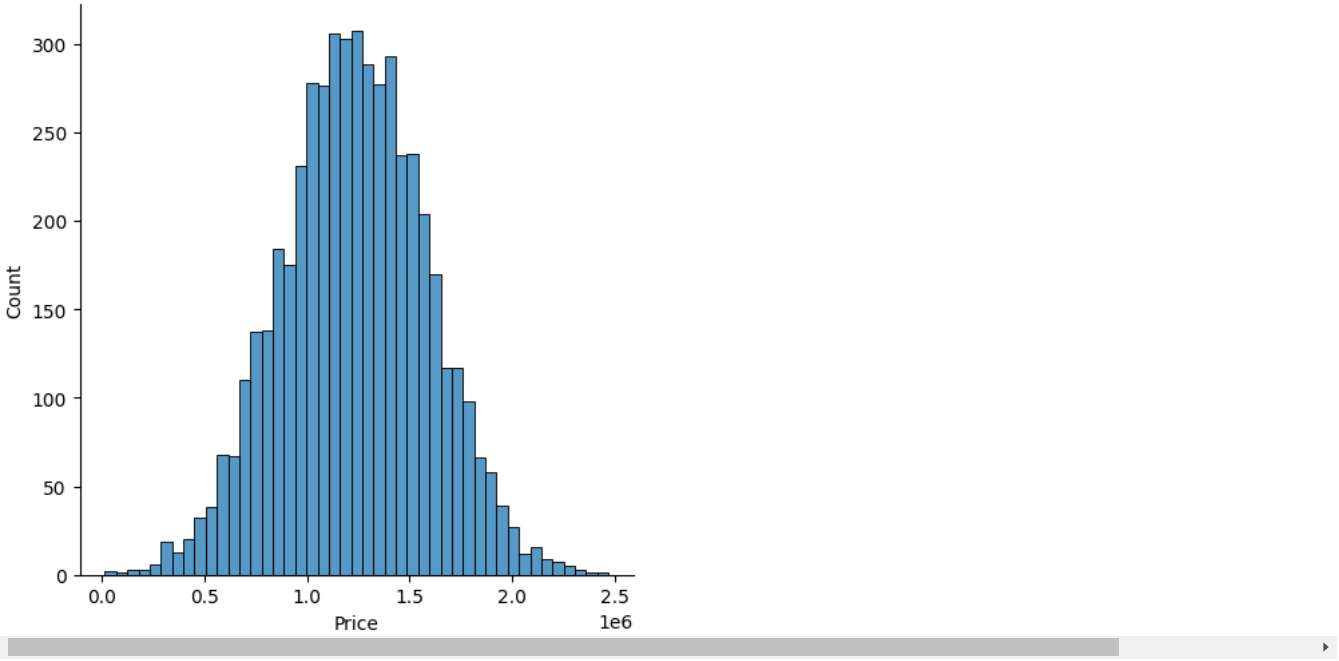
```
1 sns.pairplot(df)
```

```
C:\Users\Gokul Jana\AppData\Local\Programs\Python\Python311\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure la
  self._figure.tight_layout(*args, **kwargs)
<seaborn.axisgrid.PairGrid at 0x19c3dd77810>
```



```
1 sns.displot(df['Price'])
```

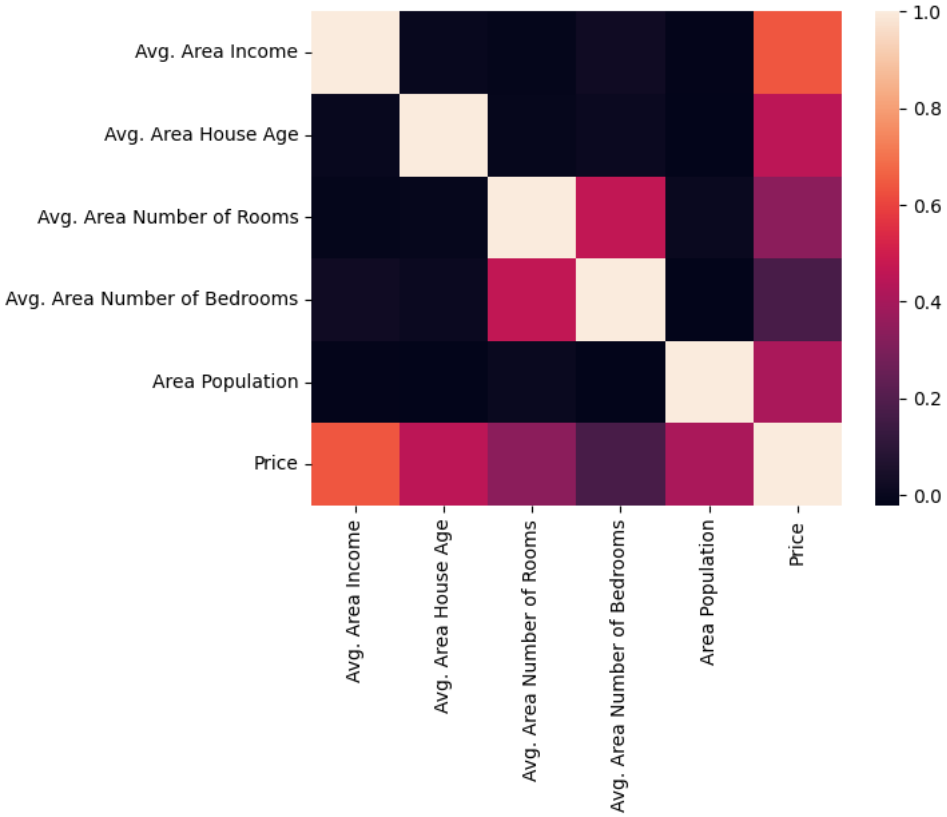
```
C:\Users\Gokul Jana\AppData\Local\Programs\Python\Python311\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layo
  self._figure.tight_layout(*args, **kwargs)
<seaborn.axisgrid.FacetGrid at 0x19c407e1e10>
```




```
1 df1=df[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms',
2        'Avg. Area Number of Bedrooms', 'Area Population', 'Price']]
```

```
1 sns.heatmap(df1.corr())
```

```
<Axes: >
```



Train the model

```
1 x=df1[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of Rooms',
2        'Avg. Area Number of Bedrooms', 'Area Population']]
3 y=df1['Price']
```

```
1 from sklearn.model_selection import train_test_split
2 x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
```

```
1 from sklearn.linear_model import LinearRegression
2 lr=LinearRegression()
3 lr.fit(x,y)
4 print(lr.intercept_)
```

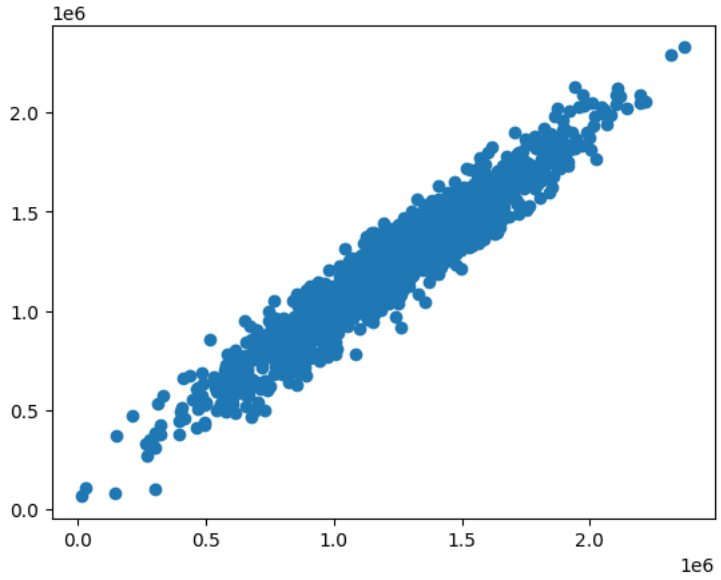
```
-2637299.033328585
```

```
1 coeff=pd.DataFrame(lr.coef_,x.columns,columns=["Co-efficient"])
2 coeff
```

	Co-efficient
Avg. Area Income	21.578049
Avg. Area House Age	165637.026941
Avg. Area Number of Rooms	120659.948816
Avg. Area Number of Bedrooms	1651.139054
Area Population	15.200744

```
1 pred=lr.predict(x_test)
2 plt.scatter(y_test,pred)
```

```
<matplotlib.collections.PathCollection at 0x19c457d5fd0>
```




```
1 lr.score(x_test,y_test)
```

```
0.9201330768038205
```

Bottle

```
1 df2=pd.read_csv("E:/Datasets/9_bottle.csv")
2 df2
```

2	1	3	054.0 056.0	4903CR-HY-060-0930-05400560-0010A-7	10	10.460	33.4370	NaN	25.65400	NaN	...	NaN	10	NaN	NaN
3	1	4	054.0 056.0	19-4903CR-HY-060-0930-05400560-0019A-3	19	10.450	33.4200	NaN	25.64300	NaN	...	NaN	19	NaN	NaN
4	1	5	054.0 056.0	19-4903CR-HY-060-0930-05400560-0020A-7	20	10.450	33.4210	NaN	25.64300	NaN	...	NaN	20	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
864858	34404	864859	093.4 026.4	20-1611SR-MX-310-2239-09340264-0000A-7	0	18.744	33.4083	5.805	23.87055	108.74	...	0.18	0	NaN	NaN
864859	34404	864860	093.4 026.4	20-1611SR-MX-310-2239-09340264-0002A-3	2	18.744	33.4083	5.805	23.87072	108.74	...	0.18	2	4.0	NaN
864860	34404	864861	093.4 026.4	20-1611SR-MX-310-2239-09340264-0005A-3	5	18.692	33.4150	5.796	23.88911	108.46	...	0.18	5	3.0	NaN
864861	34404	864862	093.4 026.4	20-1611SR-MX-310-2239-09340264-0010A-3	10	18.161	33.4062	5.816	24.01426	107.74	...	0.31	10	2.0	NaN
864862	34404	864863	093.4 026.4	20-1611SR-MX-310-2239-09340264-0015A-3	15	17.533	33.3880	5.774	24.15297	105.66	...	0.61	15	1.0	NaN

864863 rows × 74 columns

```
1 df2.isna().sum()
```

```
Cst_Cnt                    0
Btl_Cnt                    0
Sta_ID                     0
Depth_ID                   0
Depthm                     0
                         ...
TA1                   862779
TA2                   864629
pH2                   864853
pH1                   864779
DIC Quality Comment   864808
Length: 74, dtype: int64
```

```
1 df2.columns
```

```
Index(['Cst_Cnt', 'Btl_Cnt', 'Sta_ID', 'Depth_ID', 'Depthm', 'T_degC',
       'Salnty', 'O2ml_L', 'STheta', 'O2Sat', 'Oxy_µmol/Kg', 'BtlNum',
       'RecInd', 'T_prec', 'T_qual', 'S_prec', 'S_qual', 'P_qual', 'O_qual',
       'SThtaq', 'O2Satq', 'ChlorA', 'Chlqua', 'Phaeop', 'Phaqua', 'PO4uM',
       'PO4q', 'SiO3uM', 'SiO3qu', 'NO2uM', 'NO2q', 'NO3uM', 'NO3q', 'NH3uM',
       'NH3q', 'C14As1', 'C14A1p', 'C14A1q', 'C14As2', 'C14A2p', 'C14A2q',
       'DarkAs', 'DarkAp', 'DarkAq', 'MeanAs', 'MeanAp', 'MeanAq', 'IncTim',
       'LightP', 'R_Depth', 'R_TEMP', 'R_POTEMP', 'R_SALINITY', 'R_SIGMA',
       'R_SVA', 'R_DYNHT', 'R_O2', 'R_O2Sat', 'R_SIO3', 'R_PO4', 'R_NO3',
```

```
       'R_NO2', 'R_NH4', 'R_CHLA', 'R_PHAEO', 'R_PRES', 'R_SAMP', 'DIC1',
       'DIC2', 'TA1', 'TA2', 'pH2', 'pH1', 'DIC Quality Comment'],
      dtype='object')
```

```
df3=df2.drop(['DIC2', 'TA1', 'TA2', 'pH2', 'pH1', 'DIC Quality Comment'],axis=1)
```

```
df4=df3.drop(['BtlNum', 'T_qual', 'S_qual', 'O_qual', 'SThtaq', 'NH3uM', 'C14As1',
       'C14A1p', 'C14As2', 'C14A2p', 'DarkAs', 'DarkAp', 'MeanAs', 'MeanAp',
       'IncTim', 'LightP', 'R_NH4', 'R_SAMP', 'DIC1'],axis=1)
```

```
df4.isna().sum()
```

```
Cst_Cnt            0
Btl_Cnt            0
Sta_ID             0
Depth_ID           0
Depthm             0
T_degC         10963
Salnty         47354
O2ml_L        168662
STheta         52689
O2Sat         203589
Oxy_µmol/Kg   203595
RecInd             0
T_prec         10963
S_prec         47354
P_qual        191108
O2Satq        647066
ChlorA        639591
Chlqua        225697
Phaeop        639592
Phaqua        225693
PO4uM         451546
PO4q          413077
SiO3uM        510772
SiO3qu        353997
NO2uM         527287
NO2q          335389
NO3uM         527460
NO3q          334930
NH3q           56564
C14A1q         16258
C14A2q         16240
DarkAq         24423
MeanAq         24424
R_Depth            0
R_TEMP         10963
R_POTEMP       46047
R_SALINITY     47354
R_SIGMA        52856
R_SVA          52771
R_DYNHT        46657
R_O2          168662
R_O2Sat       198415
R_SIO3        510764
R_PO4         451538
R_NO3         527452
R_NO2         527279
R_CHLA        639587
R_PHAEO       639588
R_PRES             0
dtype: int64
```

```
df4.describe()
```

	Cst_Cnt	Btl_Cnt	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	Oxy_µm
count	864863.000000	864863.000000	864863.000000	853900.000000	817509.000000	696201.000000	812174.000000	661274.000000	661268.0
mean	17138.790958	432432.000000	226.831951	10.799677	33.840350	3.392468	25.819394	57.103779	148.8
std	10240.949817	249664.587269	316.050259	4.243825	0.461843	2.073256	1.167787	37.094137	90.1
min	1.000000	1.000000	0.000000	1.440000	28.431000	-0.010000	20.934000	-0.100000	-0.4
25%	8269.000000	216216.500000	46.000000	7.680000	33.488000	1.360000	24.965000	21.100000	60.9
50%	16848.000000	432432.000000	125.000000	10.060000	33.863000	3.440000	25.996000	54.400000	151.0
75%	26557.000000	648647.500000	300.000000	13.880000	34.196900	5.500000	26.646000	97.600000	240.3
max	34404.000000	864863.000000	5351.000000	31.140000	37.034000	11.130000	250.784000	214.100000	485.7

8 rows × 47 columns

```
df5=df4.iloc[0:5000,:]
df5
```

	Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	...	R_DYNHT	R_O2	R_O2Sat	R_SIO3	R_PO
0	1	1	054.0 056.0	19-4903CR-HY-060-0930-05400560-0000A-3	0	10.50	33.440	NaN	25.649	NaN	...	0.00	NaN	NaN	NaN	Na
1	1	2	054.0 056.0	19-4903CR-HY-060-0930-05400560-0008A-3	8	10.46	33.440	NaN	25.656	NaN	...	0.01	NaN	NaN	NaN	Na
2	1	3	054.0 056.0	19-4903CR-HY-060-0930-05400560-0010A-7	10	10.46	33.437	NaN	25.654	NaN	...	0.02	NaN	NaN	NaN	Na
3	1	4	054.0 056.0	19-4903CR-HY-060-0930-05400560-0019A-3	19	10.45	33.420	NaN	25.643	NaN	...	0.04	NaN	NaN	NaN	Na
4	1	5	054.0 056.0	19-4903CR-HY-060-0930-05400560-0020A-7	20	10.45	33.421	NaN	25.643	NaN	...	0.04	NaN	NaN	NaN	Na
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	.
4995	165	4996	092.0 098.0	19-4904NS-HY-102-1342-09200980-0099A-3	99	11.41	33.440	5.42	25.490	87.6	...	0.28	5.42	87.6	NaN	Na
4996	165	4997	092.0 098.0	19-4904NS-HY-102-1342-09200980-0100A-7	100	11.36	33.444	5.39	25.502	87.0	...	0.28	5.39	87.0	NaN	Na
4997	165	4998	092.0 098.0	19-4904NS-HY-102-1342-09200980	125	10.16	33.555	4.59	25.800	72.2	...	0.34	4.59	72.2	NaN	Na

```
per=df5.isna().sum()/len(df5)*100
per1=pd.DataFrame(per,df5.columns,)
per1
```

	0
Cst_Cnt	0.00
Btl_Cnt	0.00
Sta_ID	0.00
Depth_ID	0.00
Depthm	0.00
T_degC	0.40
Salnty	3.04
O2ml_L	43.80
STheta	3.34
O2Sat	45.74
Oxy_µmol/Kg	45.74
Reclnd	0.00
T_prec	0.40
S_prec	3.04
P_qual	0.00
O2Satq	52.88
ChlorA	100.00
Chlqua	0.00
Phaeop	100.00
Phaqua	0.00
PO4uM	79.08
PO4q	20.92
SiO3uM	100.00
SiO3qu	0.00
NO2uM	100.00
NO2q	0.00
NO3uM	100.00
NO3q	0.00
NH3q	0.00
C14A1q	0.00
C14A2q	0.00

```
1 pr=per1[per1[0]>75].index
2 pr
```

```
Index(['ChlorA', 'Phaeop', 'PO4uM', 'SiO3uM', 'NO2uM', 'NO3uM', 'R_SIO3',
       'R_PO4', 'R_NO3', 'R_NO2', 'R_CHLA', 'R_PHAEO'],
      dtype='object')
```

```
df6=df5.drop(['ChlorA', 'Phaeop', 'PO4uM', 'SiO3uM', 'NO2uM', 'NO3uM', 'R_SIO3',
       'R_PO4', 'R_NO3', 'R_NO2', 'R_CHLA', 'R_PHAEO'],axis=1)
df6.isna().sum()/len(df5)*100
```

```
Cst_Cnt         0.00
Btl_Cnt         0.00
Sta_ID          0.00
Depth_ID        0.00
Depthm          0.00
T_degC          0.40
Salnty          3.04
O2ml_L         43.80
STheta          3.34
O2Sat          45.74
Oxy_µmol/Kg    45.74
RecInd          0.00
T_prec          0.40
S_prec          3.04
P_qual          0.00
O2Satq         52.88
Chlqua          0.00
Phaqua          0.00
PO4q           20.92
SiO3qu          0.00
NO2q            0.00
NO3q            0.00
NH3q            0.00
C14A1q          0.00
C14A2q          0.00
DarkAq          0.00
MeanAq          0.00
R_Depth         0.00
R_TEMP          0.40
R_POTEMP        4.50
R_SALINITY      3.04
R_SIGMA         5.62
R_SVA           5.62
R_DYNHT         4.28
R_O2           43.80
R_O2Sat        46.20
R_PRES          0.00
dtype: float64
```

```
df6.fillna(df6["T_degC"].mean())
df6.fillna(df6["Salnty"].mean())
df6.fillna(df6["O2ml_L"].mean())
df6.fillna(df6["STheta"].mean())
df6.fillna(df6["O2Sat"].median())
df6.fillna(df6["Oxy_µmol/Kg"].mean())
df6.fillna(df6["T_prec"].mean())
df6.fillna(df6["S_prec"].mean())
df6.fillna(df6["O2Satq"].mean())
df6.fillna(df6["PO4q"].mean())
df6.fillna(df6["R_TEMP"].mean())
df6.fillna(df6["R_POTEMP"].mean())
df6.fillna(df6["R_SALINITY"].mean())
df6.fillna(df6["R_SIGMA"].mean())
df6.fillna(df6["R_SVA"].mean())
df6.fillna(df6["R_DYNHT"].mean())
df6.fillna(df6["R_O2"].mean())
df6=df6.fillna(df6["R_O2Sat"].mean())
```

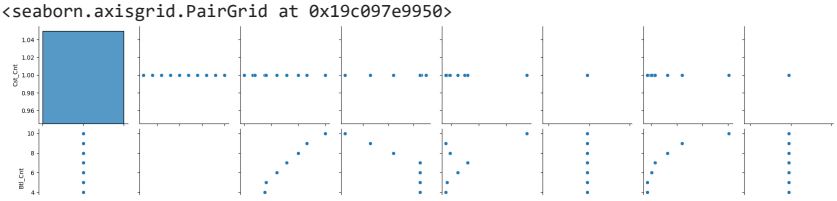
```
df6.isna().sum()
```

```
Cst_Cnt        0
Btl_Cnt        0
Sta_ID         0
Depth_ID       0
Depthm         0
T_degC         0
Salnty         0
O2ml_L         0
STheta         0
O2Sat          0
Oxy_µmol/Kg    0
RecInd         0
T_prec         0
S_prec         0
P_qual         0
O2Satq         0
Chlqua         0
```

```
Phaqua         0
PO4q           0
SiO3qu         0
NO2q           0
NO3q           0
NH3q           0
C14A1q         0
C14A2q         0
DarkAq         0
MeanAq         0
R_Depth        0
R_TEMP         0
R_POTEMP       0
R_SALINITY     0
R_SIGMA        0
R_SVA          0
R_DYNHT        0
R_O2           0
R_O2Sat        0
R_PRES         0
dtype: int64
```

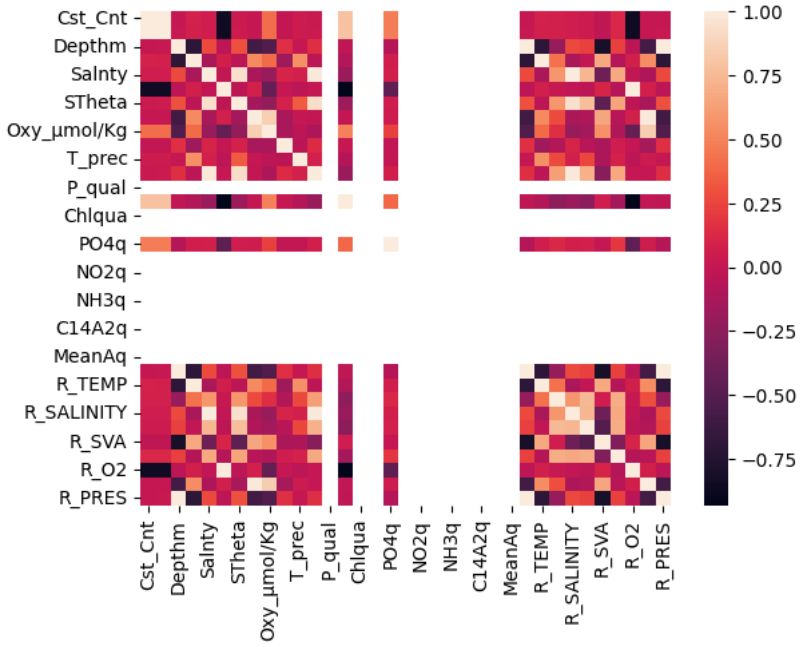
```
df7=df6.iloc[:10,:10]
sns.pairplot(df7)
```

```
C:\Users\Gokul Jana\AppData\Local\Programs\Python\Python311\Lib\site-packages\seab
  self._figure.tight_layout(*args, **kwargs)
<seaborn.axisgrid.PairGrid at 0x19c097e9950>
```



```
df6=df6.drop(["Sta_ID","Depth_ID"],axis=1)
sns.heatmap(df6.corr())
```

```
<Axes: >
```




```
x=df6.drop(["R_O2"],axis=1)
y=df6["R_O2"]
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3)
model1=LinearRegression()
model1.fit(x_train,y_train)
model1.intercept_
```

```
4.476419235288631e-13
```

```
coeff=pd.DataFrame(model1.coef_,x.columns,columns=["Coefficient"])
coeff
```