

Problem Statement

In [1]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:

```
d=pd.read_csv(r"C:\Users\user\Downloads\bottle.csv")[0:50]
d
```

a\Anaconda3\lib\site-packages\IPython\core\interactiveshell
ypeWarning: Columns (47,73) have mixed types.Specify dtype
ort or set low_memory=False.
= await self.run_ast_nodes(code_ast.body, cell_name,

tl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	...	R_PHAEO	R_PRES	R_SAM
1	054.0	19-4903CR-HY-060-0930-05400560-0000A-3	0	10.50	33.440	NaN	25.649	NaN	...	NaN	0	Na
	056.0											
	054.0	19-4903CR-HY-060-0930-05400560-0000A-3										

In [3]:

```
d.head(10)
```

Out[3]:

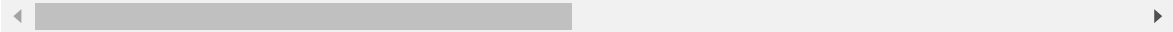
Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	...
0	1	1	19-4903CR-HY-060-0930-05400560-0000A-3	0	10.50	33.440	NaN	25.649	NaN	...
1	1	2	19-4903CR-HY-060-0930-05400560-0008A-3	8	10.46	33.440	NaN	25.656	NaN	...
2	1	3	19-4903CR-HY-060-0930-05400560-0010A-7	10	10.46	33.437	NaN	25.654	NaN	...
3	1	4	19-4903CR-HY-060-0930-05400560-0019A-3	19	10.45	33.420	NaN	25.643	NaN	...
4	1	5	19-4903CR-HY-060-0930-05400560-0020A-7	20	10.45	33.421	NaN	25.643	NaN	...
5	1	6	19-4903CR-HY-060-0930-05400560-0030A-7	30	10.45	33.431	NaN	25.651	NaN	...
6	1	7	19-4903CR-HY-060-0930-05400560-0039A-3	39	10.45	33.440	NaN	25.658	NaN	...
7	1	8	19-4903CR-HY-060-0930-05400560-0050A-7	50	10.24	33.424	NaN	25.682	NaN	...
8	1	9	19-4903CR-HY-060-0930-05400560-0058A-3	58	10.06	33.420	NaN	25.710	NaN	...

```
      Cst_Cnt  Btl_Cnt  Sta_ID  Depth_ID  Depthm  T_degC  Salnty  O2ml_L  STheta  O2Sat  ...
In [4]: 1      10      054.0      19-4903CR-HY-060-0930-056.0      75      9.86  33.494      NaN  25.801  NaN  ...
d.describe()
```

Out[4]:
10 rows × 74 columns

	Cst_Cnt	Btl_Cnt	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	Oxy
count	50.000000	50.000000	50.000000	50.000000	49.000000	0.0	49.000000	0.0	
mean	1.420000	25.500000	201.800000	8.234000	33.676980	NaN	26.186367	NaN	
std	0.498569	14.57738	216.51639	1.837252	0.370375	NaN	0.539026	NaN	
min	1.000000	1.000000	0.000000	4.440000	32.939000	NaN	25.335000	NaN	
25%	1.000000	13.250000	47.750000	6.812500	33.437000	NaN	25.682000	NaN	
50%	1.000000	25.500000	125.000000	8.620000	33.689000	NaN	26.162000	NaN	
75%	2.000000	37.750000	284.250000	9.810000	33.957000	NaN	26.680000	NaN	
max	2.000000	50.000000	896.000000	10.500000	34.310000	NaN	27.198000	NaN	

8 rows × 70 columns



In [5]:

```
d.info()
```

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

```
RangeIndex: 50 entries, 0 to 49
```

```
Data columns (total 74 columns):
```

#	Column	Non-Null Count	Dtype
0	Cst_Cnt	50 non-null	int64
1	Btl_Cnt	50 non-null	int64
2	Sta_ID	50 non-null	object
3	Depth_ID	50 non-null	object
4	Depthm	50 non-null	int64
5	T_degC	50 non-null	float64
6	Salnty	49 non-null	float64
7	O2ml_L	0 non-null	float64
8	STheta	49 non-null	float64
9	O2Sat	0 non-null	float64
10	Oxy_μmol/Kg	0 non-null	float64
11	BtlNum	0 non-null	float64
12	RecInd	50 non-null	int64
13	T_prec	50 non-null	float64
14	T_qual	0 non-null	float64
15	S_prec	49 non-null	float64
16	S_qual	1 non-null	float64
17	P_qual	50 non-null	float64
18	O_qual	50 non-null	float64
19	SThtaq	1 non-null	float64
20	O2Satq	50 non-null	float64
21	ChlorA	0 non-null	float64
22	Chlqua	50 non-null	float64
23	Phaeop	0 non-null	float64
24	Phaqua	50 non-null	float64
25	PO4uM	0 non-null	float64
26	PO4q	50 non-null	float64
27	SiO3uM	0 non-null	float64
28	SiO3qu	50 non-null	float64
29	NO2uM	0 non-null	float64
30	NO2q	50 non-null	float64
31	NO3uM	0 non-null	float64
32	NO3q	50 non-null	float64
33	NH3uM	0 non-null	float64
34	NH3q	50 non-null	float64
35	C14As1	0 non-null	float64
36	C14A1p	0 non-null	float64
37	C14A1q	50 non-null	float64
38	C14As2	0 non-null	float64
39	C14A2p	0 non-null	float64
40	C14A2q	50 non-null	float64
41	DarkAs	0 non-null	float64
42	DarkAp	0 non-null	float64
43	DarkAq	50 non-null	float64
44	MeanAs	0 non-null	float64
45	MeanAp	0 non-null	float64
46	MeanAq	50 non-null	float64
47	IncTim	0 non-null	object
48	LightP	0 non-null	float64
49	R_Depth	50 non-null	float64
50	R_TEMP	50 non-null	float64
51	R_POTEMP	50 non-null	float64
52	R_SALINITY	49 non-null	float64
53	R_SIGMA	49 non-null	float64
54	R_SVA	49 non-null	float64
55	R_DYNHT	50 non-null	float64

```

56 R_O2          0 non-null    float64
57 R_O2Sat      0 non-null    float64
58 R_SIO3       0 non-null    float64
59 R_PO4        0 non-null    float64
60 R_NO3        0 non-null    float64
61 R_NO2        0 non-null    float64
62 R_NH4        0 non-null    float64
63 R_CHLA       0 non-null    float64
64 R_PHAEO      0 non-null    float64
65 R_PRES       50 non-null    int64
66 R_SAMP       0 non-null    float64
67 DIC1        0 non-null    float64
68 DIC2        0 non-null    float64
69 TA1         0 non-null    float64
70 TA2         0 non-null    float64
71 pH2         0 non-null    float64
72 pH1         0 non-null    float64
73 DIC Quality Comment 0 non-null    object

```

dtypes: float64(65), int64(5), object(4)

memory usage: 29.0+ KB

In [6]:

```
d.columns
```

Out[6]:

```

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_qua
l',
      'SThtaQ', 'O2Satq', 'ChlorA', 'Chlqua', 'Phaeop', 'Phaqua', 'P04u
M',
      'P04q', 'SiO3uM', 'SiO3qu', 'NO2uM', 'NO2q', 'NO3uM', 'NO3q', 'NH3u
M',
      'NH3q', 'C14As1', 'C14A1p', 'C14A1q', 'C14As2', 'C14A2p', 'C14A2q',
      'DarkAs', 'DarkAp', 'DarkAq', 'MeanAs', 'MeanAp', 'MeanAq', 'IncTi
m',
      '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')

```

In []:

In [7]:

```
da=d[['Cst_Cnt', 'Btl_Cnt', 'Sta_ID', 'Depth_ID', 'Depthm', 'T_degC']]
da
```

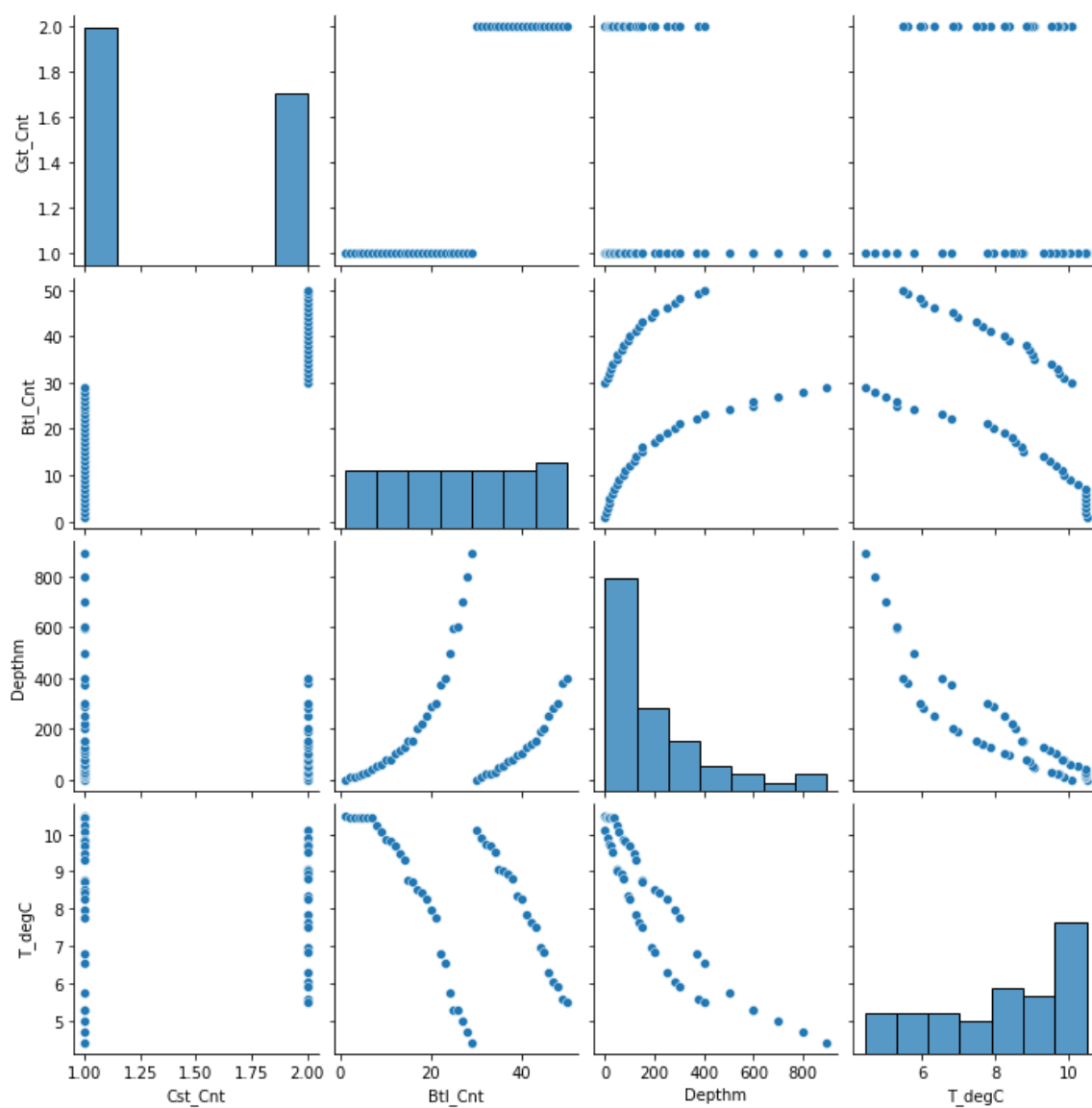
			056.0		0400A-7		
23	1	24	054.0 056.0	19-4903CR-HY-060-0930-05400560-	0500A-7	500	5.78
24	1	25	054.0 056.0	19-4903CR-HY-060-0930-05400560-	0598A-3	598	5.30
25	1	26	054.0 056.0	19-4903CR-HY-060-0930-05400560-	0600A-7	600	5.29
26	1	27	054.0 056.0	19-4903CR-HY-060-0930-05400560-	0700A-7	700	5.01
27	1	28	054.0 056.0	19-4903CR-HY-060-0930-05400560-	0800A-7	800	4.72
28	1	29	054.0 056.0	19-4903CR-HY-060-0930-05400560-	0896A-3	896	4.44
29	2	30	052.0 075.0	19-4903CR-HY-060-2112-05200750-	0000B-3	0	10.10
30	2	31	052.0 075.0	19-4903CR-HY-060-2112-05200750-	0010A-3	10	9.89

In [8]:

```
sns.pairplot(da)
```

Out[8]:

<seaborn.axisgrid.PairGrid at 0x2262e079640>



In [9]:

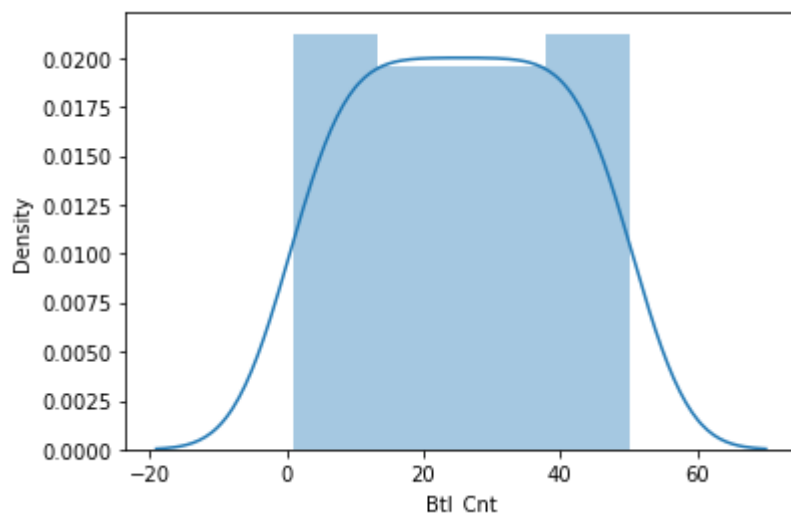
```
sns.distplot(d['Btl_Cnt'])
```

C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:2557: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

```
warnings.warn(msg, FutureWarning)
```

Out[9]:

```
<AxesSubplot:xlabel='Btl_Cnt', ylabel='Density'>
```

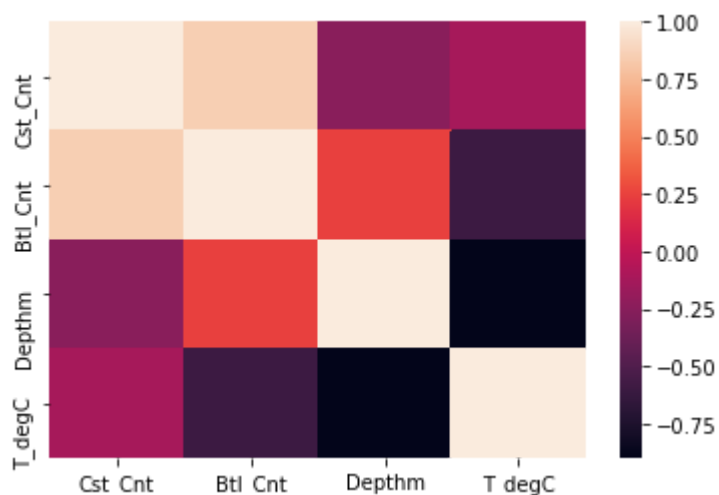


In [13]:

```
sns.heatmap(da.corr())
```

Out[13]:

```
<AxesSubplot:>
```



In [22]:

```
x=da[['Cst_Cnt', 'Btl_Cnt']]
y=da['Depthm']
```

In [23]:

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

In [24]:

```
from sklearn.linear_model import LinearRegression

lr=LinearRegression()
lr.fit(x_train,y_train)
```

Out[24]:

LinearRegression()

In [25]:

```
print(lr.intercept_)
```

613.9867751763823

In [26]:

```
coeff=pd.DataFrame(lr.coef_,x.columns,columns=['Co-effecient'])
coeff
```

Out[26]:

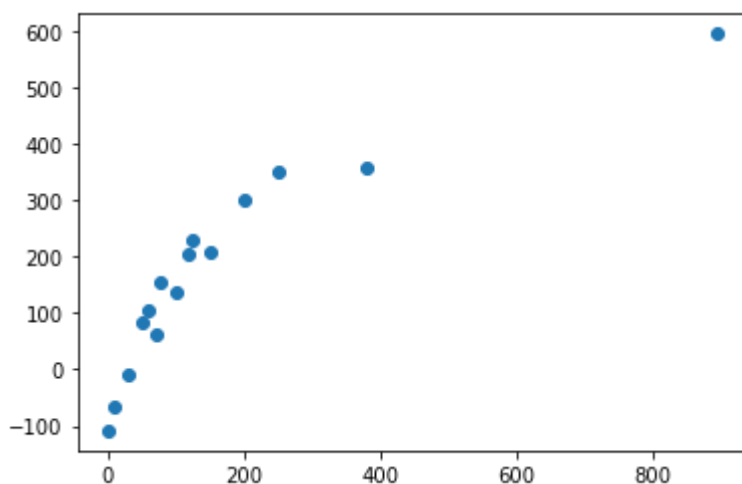
	Co-effecient
Cst_Cnt	-727.565651
Btl_Cnt	24.420375

In [27]:

```
prediction=lr.predict(x_test)
plt.scatter(y_test,prediction)
```

Out[27]:

<matplotlib.collections.PathCollection at 0x2263116d100>



In [28]:

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

0.7700204951831962

In []: