Linear Algebra

1. Create 5 matrices with five different dimensions (1-D,2-D,...5-D)

In [54]:
```
import numpy as np
from numpy import linalg as la
```

In [3]:
```
arr=np.array([1,2,3])
print(arr)
```

```
[1 2 3]
```

In [57]:
```
arr1=np.array([[1,2,3],[4,5,6],[7,8,9]])
print(arr1)
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```

In [106]:
```
arr2=np.array([[[1,2,3,5],[7,8,9,7],
                [4,5,6,3],[7,8,9,1],
                [1,2,3,6],[7,8,9,1]]])
print(arr2)
```

```
[[[1 2 3 5]
  [7 8 9 7]
  [4 5 6 3]
  [7 8 9 1]
  [1 2 3 6]
  [7 8 9 1]]]
```

In [21]:
```
arr3=np.array([[[[1,2,3],[4,5,6]]],
               [[[4,5,6],[1,2,3]]]])
print(arr3)
```

```
[[[[1 2 3]
   [4 5 6]]]


 [[[4 5 6]
   [1 2 3]]]]
```

In [47]:
```
arr4=np.array([[[[[1,2,3]],[[4,5,6]]],[[[4,5,6]],[[1,2,3]]]],
               [[[[1,2,3]],[[4,5,6]]],[[[4,5,6]],[[1,2,3]]]],
               [[[[1,2,3]],[[4,5,6]]],[[[4,5,6]],[[1,2,3]]]],
               [[[[1,2,3]],[[4,5,6]]],[[[4,5,6]],[[1,2,3]]]]])
print(arr4)
```

```
[[[[[1 2 3]]

   [[4 5 6]]]


  [[[4 5 6]]

   [[1 2 3]]]]



 [[[[1 2 3]]

   [[4 5 6]]]


  [[[4 5 6]]

   [[1 2 3]]]]



 [[[[1 2 3]]

   [[4 5 6]]]


  [[[4 5 6]]

   [[1 2 3]]]]



 [[[[1 2 3]]

   [[4 5 6]]]


  [[[4 5 6]]

   [[1 2 3]]]]]
```

2. Find determinants of 5 matrices and display your output

In [75]:
```
print(la.det(arr1))
```

```
-9.51619735392994e-16
```

3. Find inverse of the above 5 matrices and display your output

In [58]:
```
print(la.inv(arr1))
```

```
[[ 3.15251974e+15 -6.30503948e+15  3.15251974e+15]
 [-6.30503948e+15  1.26100790e+16 -6.30503948e+15]
 [ 3.15251974e+15 -6.30503948e+15  3.15251974e+15]]
```

4. Find the rank, diagonal and trace of the 5 matrices

In [82]:
```
print(la.matrix_rank(arr))
```

```
1
```

In [83]:
```
print(la.matrix_rank(arr1))
```

```
2
```

In [84]:
```
print(la.matrix_rank(arr2))
```

```
[4]
```

In [85]:
```
print(la.matrix_rank(arr3))
```

```
[[2]
 [2]]
```

In [86]:
```
print(la.matrix_rank(arr4))
```

```
[[[1 1]
  [1 1]]

 [[1 1]
  [1 1]]

 [[1 1]
  [1 1]]

 [[1 1]
  [1 1]]]
```

In [87]:
```
print(np.diag(arr))
```

```
[[1 0 0]
 [0 2 0]
 [0 0 3]]
```

In [88]:
```
print(np.diag(arr1))
```

```
[1 5 9]
```

In [94]:
```
print(np.trace(arr1))
```

```
15
```

In [95]:
```
print(np.trace(arr2))
```

```
[1 2 3 5 6 7 1 2 3]
```

In [96]:
```
print(np.trace(arr3))
```

```
[[1 2 3]
 [4 5 6]]
```

In [97]:
```
print(np.trace(arr4))
```

```
[[[5 7 9]]

 [[5 7 9]]]
```

5. Find Eigen value and eigen vector for 5 matrices

In [99]:
```
x,y=la.eig(arr1)
print("root",x)
print("matrix",y)
```

```
root [ 1.61168440e+01 -1.11684397e+00 -3.38433605e-16]
matrix [[-0.23197069 -0.78583024  0.40824829]
 [-0.52532209 -0.08675134 -0.81649658]
 [-0.8186735   0.61232756  0.40824829]]
```

In [104]:
```
print(la.eigvals(arr1))
```

```
[ 1.61168440e+01 -1.11684397e+00 -3.38433605e-16]
```

DV

With the two datasets given (refer to drive) - Frame a problem statement, clean, preprocess and visulaize the data and interpret your conclusion"

1)Find the relation between radius_mean & area_mean

In [109]:
```
import pandas as pd
```

In [110]:
```
df=pd.read_csv(r"C:\Users\user\Downloads\8_BreastCancerPrediction.csv")
df
```

Out[110]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_
0	842302	M	17.99	10.38	122.80	1001.0	0.
1	842517	M	20.57	17.77	132.90	1326.0	0.
2	84300903	M	19.69	21.25	130.00	1203.0	0.
3	84348301	M	11.42	20.38	77.58	386.1	0.
4	84358402	M	20.29	14.34	135.10	1297.0	0.
...	...	...	...	...	...	...	
564	926424	M	21.56	22.39	142.00	1479.0	0.
565	926682	M	20.13	28.25	131.20	1261.0	0.
566	926954	M	16.60	28.08	108.30	858.1	0.
567	927241	M	20.60	29.33	140.10	1265.0	0.
568	92751	B	7.76	24.54	47.92	181.0	0.

569 rows × 33 columns

In [114]:
```
df.isnull()
```

Out[114]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mea
0	False	False	False	False	False	False	Fals
1	False	False	False	False	False	False	Fals
2	False	False	False	False	False	False	Fals
3	False	False	False	False	False	False	Fals
4	False	False	False	False	False	False	Fals
...	...	...	...	...	...	...	
564	False	False	False	False	False	False	Fals
565	False	False	False	False	False	False	Fals
566	False	False	False	False	False	False	Fals
567	False	False	False	False	False	False	Fals
568	False	False	False	False	False	False	Fals

569 rows × 33 columns

In [111]:
```
df1=df
df1.fillna(value=0)
```

Out[111]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_
0	842302	M	17.99	10.38	122.80	1001.0	0.
1	842517	M	20.57	17.77	132.90	1326.0	0.
2	84300903	M	19.69	21.25	130.00	1203.0	0.
3	84348301	M	11.42	20.38	77.58	386.1	0.
4	84358402	M	20.29	14.34	135.10	1297.0	0.
...	...	...	...	...	...	...	
564	926424	M	21.56	22.39	142.00	1479.0	0.
565	926682	M	20.13	28.25	131.20	1261.0	0.
566	926954	M	16.60	28.08	108.30	858.1	0.
567	927241	M	20.60	29.33	140.10	1265.0	0.
568	92751	B	7.76	24.54	47.92	181.0	0.

569 rows × 33 columns

In [113]:
```
df1.dropna(axis=1)
```

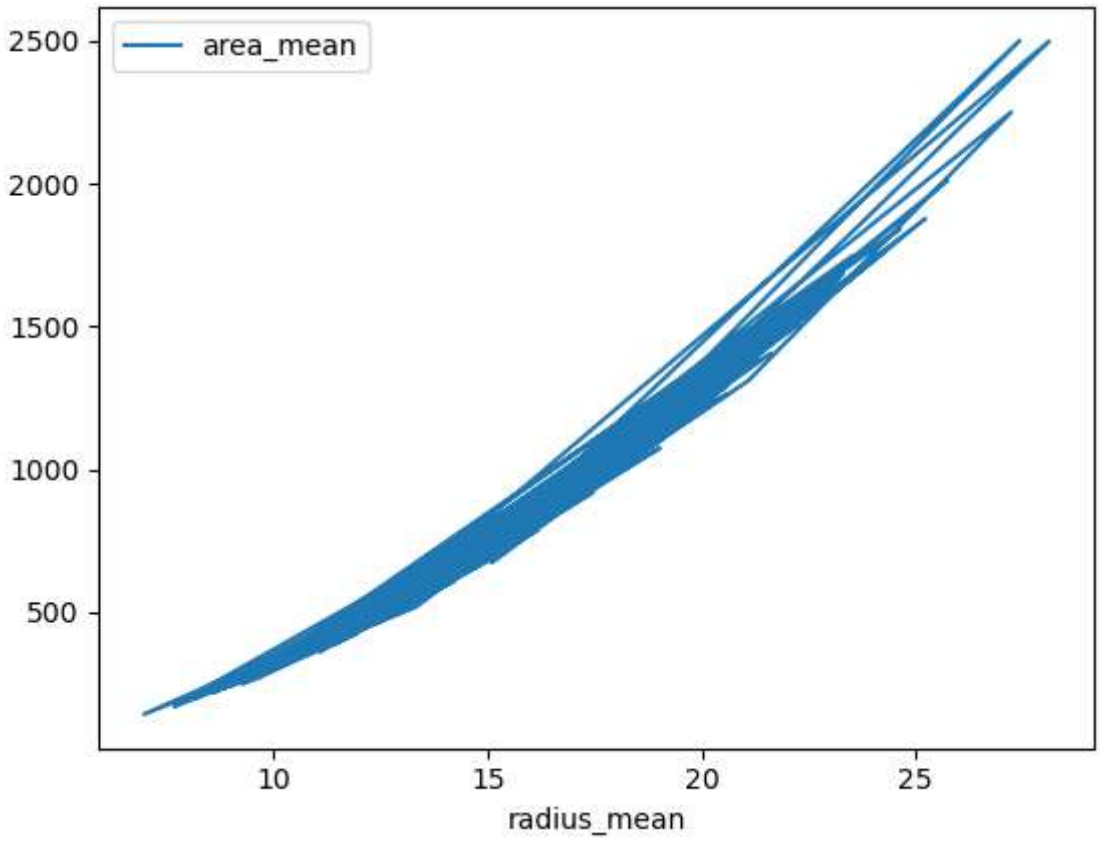
Out[113]:

	id	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_
0	842302	M	17.99	10.38	122.80	1001.0	0.
1	842517	M	20.57	17.77	132.90	1326.0	0.
2	84300903	M	19.69	21.25	130.00	1203.0	0.
3	84348301	M	11.42	20.38	77.58	386.1	0.
4	84358402	M	20.29	14.34	135.10	1297.0	0.
...	...	...	...	...	...	...	
564	926424	M	21.56	22.39	142.00	1479.0	0.
565	926682	M	20.13	28.25	131.20	1261.0	0.
566	926954	M	16.60	28.08	108.30	858.1	0.
567	927241	M	20.60	29.33	140.10	1265.0	0.
568	92751	B	7.76	24.54	47.92	181.0	0.

569 rows × 32 columns

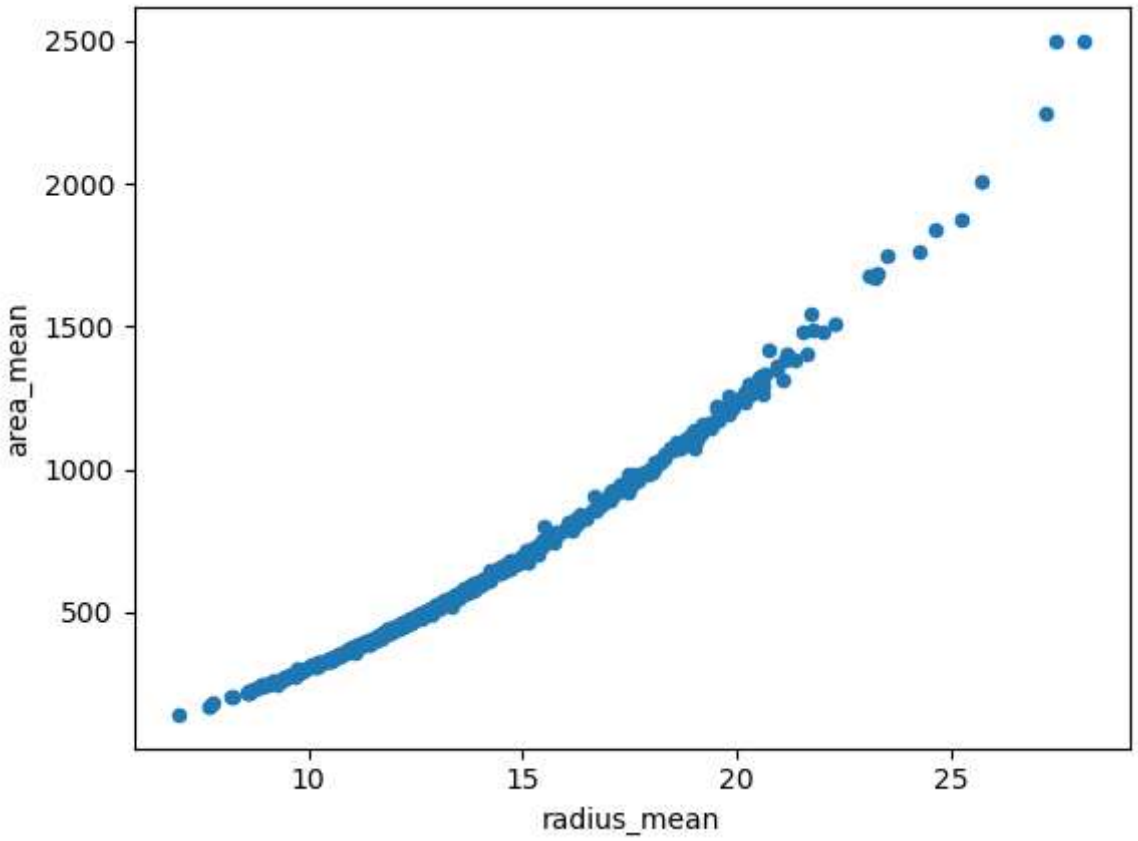
In [121]:
```
df1.plot.line('radius_mean','area_mean')
```

Out[121]:
```
<Axes: xlabel='radius_mean'>
```



In [116]:
```
df1.plot.scatter(x='radius_mean',y='area_mean')
```

Out[116]:
```
<Axes: xlabel='radius_mean', ylabel='area_mean'>
```



both radius_mean & area_mean consists of similar values

In [ ]:
```
2)vishualize Depthm & Btl_Cnt
```

In [122]:
```
dff=pd.read_csv(r"C:\Users\user\Downloads\bottle.csv")
dff
```

```
C:\Users\user\AppData\Local\Temp\ipykernel_7948\2901418093.py:1: DtypeWarnin
g: Columns (47,73) have mixed types. Specify dtype option on import or set lo
w_memory=False.
  dff=pd.read_csv(r"C:\Users\user\Downloads\bottle.csv")
```

Out[122]:

	Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2S
0	1	1	054.0 056.0	19-4903CR-HY-060-0930-05400560-0000A-3	0	10.500	33.4400	NaN	25.64900	Na
1	1	2	054.0 056.0	19-4903CR-HY-060-0930-05400560-0008A-3	8	10.460	33.4400	NaN	25.65600	Na
2	1	3	054.0 056.0	19-4903CR-HY-060-0930-05400560-0010A-7	10	10.460	33.4370	NaN	25.65400	Na
3	1	4	054.0 056.0	19-4903CR-HY-060-0930-05400560-0019A-3	19	10.450	33.4200	NaN	25.64300	Na
4	1	5	054.0 056.0	19-4903CR-HY-060-0930-05400560-0020A-7	20	10.450	33.4210	NaN	25.64300	Na
...	...	...	...	...	...	...	...	...	...	
864858	34404	864859	093.4 026.4	20-1611SR-MX-310-2239-09340264-0000A-7	0	18.744	33.4083	5.805	23.87055	108.7
864859	34404	864860	093.4 026.4	20-1611SR-MX-310-2239-09340264-0002A-3	2	18.744	33.4083	5.805	23.87072	108.7
864860	34404	864861	093.4 026.4	20-1611SR-MX-310-2239-09340264-0005A-3	5	18.692	33.4150	5.796	23.88911	108.4
864861	34404	864862	093.4 026.4	20-1611SR-MX-310-2239-09340264-0010A-3	10	18.161	33.4062	5.816	24.01426	107.7

	Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2S
864862	34404	864863	093.4 026.4	20-1611SR-MX-310-2239-09340264-0015A-3	15	17.533	33.3880	5.774	24.15297	105.6

864863 rows × 74 columns

In [123]:
```
dff1=dff[0:10000]
dff1
```

Out[123]:

Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	...	R_PHAEO	R_I
1	054.0 056.0	19-4903CR-HY-060-0930-05400560-0000A-3	0	10.50	33.440	NaN	25.649	NaN	...	NaN	
2	054.0 056.0	19-4903CR-HY-060-0930-05400560-0008A-3	8	10.46	33.440	NaN	25.656	NaN	...	NaN	
3	054.0 056.0	19-4903CR-HY-060-0930-05400560-0010A-7	10	10.46	33.437	NaN	25.654	NaN	...	NaN	
4	054.0 056.0	19-4903CR-HY-060-0930-05400560-0019A-3	19	10.45	33.420	NaN	25.643	NaN	...	NaN	
5	054.0 056.0	19-4903CR-HY-060-0930-05400560-0020A-7	20	10.45	33.421	NaN	25.643	NaN	...	NaN	
...	...	...	...	...	...	...	...	...	...	...	
9996	102.0 074.0	19-4906HO-HY-152-1806-10200740-0043A-3	43	15.71	33.640	5.46	24.769	96.5	...	NaN	
9997	102.0 074.0	19-4906HO-HY-152-1806-10200740-0050A-7	50	15.35	33.621	5.26	24.835	92.3	...	NaN	
9998	102.0 074.0	19-4906HO-HY-152-1806-10200740-0065A-3	65	14.64	33.510	4.74	24.904	82.0	...	NaN	
9999	102.0 074.0	19-4906HO-HY-152-1806-10200740-0075A-7	75	14.04	33.459	4.73	24.991	80.8	...	NaN	

Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	...	R_PHAEO	R_I
10000	102.0 074.0	19-4906HO-HY-152-1806-10200740-0100A-7	100	12.60	33.453	4.70	25.276	77.9	...	NaN	

'4 columns

In [128]:
```
dff.isnull()
```

Out[128]:

	Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	.
0	False	False	False	False	False	False	False	True	False	True	.
1	False	False	False	False	False	False	False	True	False	True	.
2	False	False	False	False	False	False	False	True	False	True	.
3	False	False	False	False	False	False	False	True	False	True	.
4	False	False	False	False	False	False	False	True	False	True	.
...	...	...	...	...	...	...	...	...	...	...	.
864858	False	False	False	False	False	False	False	False	False	False	.
864859	False	False	False	False	False	False	False	False	False	False	.
864860	False	False	False	False	False	False	False	False	False	False	.
864861	False	False	False	False	False	False	False	False	False	False	.
864862	False	False	False	False	False	False	False	False	False	False	.

864863 rows × 74 columns

In [126]:
```
dff2=dff1
dff2.fillna(value=1)
```

Out[126]:

	Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	...
0	1	1	054.0 056.0	19-4903CR-HY-060-0930-05400560-0000A-3	0	10.50	33.440	1.00	25.649	1.0	...
1	1	2	054.0 056.0	19-4903CR-HY-060-0930-05400560-0008A-3	8	10.46	33.440	1.00	25.656	1.0	...
2	1	3	054.0 056.0	19-4903CR-HY-060-0930-05400560-0010A-7	10	10.46	33.437	1.00	25.654	1.0	...
3	1	4	054.0 056.0	19-4903CR-HY-060-0930-05400560-0019A-3	19	10.45	33.420	1.00	25.643	1.0	...
4	1	5	054.0 056.0	19-4903CR-HY-060-0930-05400560-0020A-7	20	10.45	33.421	1.00	25.643	1.0	...
...	...	...	...	...	...	...	...	...	...	...	...
9995	331	9996	102.0 074.0	19-4906HO-HY-152-1806-10200740-0043A-3	43	15.71	33.640	5.46	24.769	96.5	...
9996	331	9997	102.0 074.0	19-4906HO-HY-152-1806-10200740-0050A-7	50	15.35	33.621	5.26	24.835	92.3	...
9997	331	9998	102.0 074.0	19-4906HO-HY-152-1806-10200740-0065A-3	65	14.64	33.510	4.74	24.904	82.0	...
9998	331	9999	102.0 074.0	19-4906HO-HY-152-1806-10200740-0075A-7	75	14.04	33.459	4.73	24.991	80.8	...

	Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	T_degC	Salnty	O2ml_L	STheta	O2Sat	...
9999	331	10000	102.0 074.0	19-4906HO-HY-152-1806-10200740-0100A-7	100	12.60	33.453	4.70	25.276	77.9	...

10000 rows × 74 columns

In [127]:
```
dff2.dropna(axis=1)
```

Out[127]:

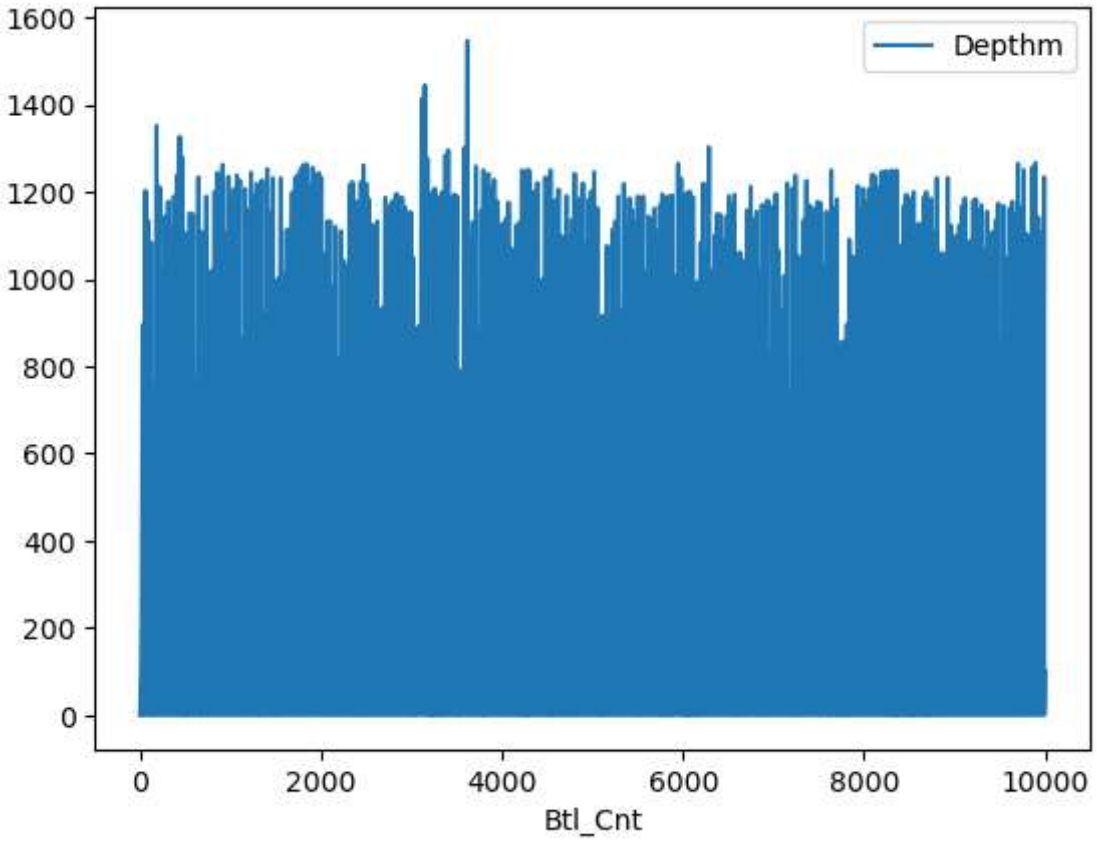
	Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	RecInd	P_qual	Chlqua	Phaqua	SiO3qu	N
0	1	1	054.0 056.0	19-4903CR-HY-060-0930-05400560-0000A-3	0	3	9.0	9.0	9.0	9.0	
1	1	2	054.0 056.0	19-4903CR-HY-060-0930-05400560-0008A-3	8	3	9.0	9.0	9.0	9.0	
2	1	3	054.0 056.0	19-4903CR-HY-060-0930-05400560-0010A-7	10	7	9.0	9.0	9.0	9.0	
3	1	4	054.0 056.0	19-4903CR-HY-060-0930-05400560-0019A-3	19	3	9.0	9.0	9.0	9.0	
4	1	5	054.0 056.0	19-4903CR-HY-060-0930-05400560-0020A-7	20	7	9.0	9.0	9.0	9.0	
...	...	...	...	...	...	...	...	...	...	...	...
9995	331	9996	102.0 074.0	19-4906HO-HY-152-1806-10200740-0043A-3	43	3	9.0	9.0	9.0	9.0	
9996	331	9997	102.0 074.0	19-4906HO-HY-152-1806-10200740-0050A-7	50	7	9.0	9.0	9.0	9.0	
9997	331	9998	102.0 074.0	19-4906HO-HY-152-1806-10200740-0065A-3	65	3	9.0	9.0	9.0	9.0	
9998	331	9999	102.0 074.0	19-4906HO-HY-152-1806-10200740-0075A-7	75	7	9.0	9.0	9.0	9.0	

	Cst_Cnt	Btl_Cnt	Sta_ID	Depth_ID	Depthm	RecInd	P_qual	Chlqua	Phaqua	SiO3qu	N
9999	331	10000	102.0 074.0	19-4906HO-HY-152-1806-10200740-0100A-7	100	7	9.0	9.0	9.0	9.0	

10000 rows × 19 columns

In [137]:
```
dff2.plot.line('Btl_Cnt','Depthm')
```

Out[137]:
```
<Axes: xlabel='Btl_Cnt'>
```



In [139]:
```
dff2.plot.scatter(x='Btl_Cnt',y='Depthm')
```

Out[139]:
```
<Axes: xlabel='Btl_Cnt', ylabel='Depthm'>
```