

In [1]: `!pip install numpy pandas matplotlib`

```
Requirement already satisfied: numpy in d:\anaconda\lib\site-packages (1.20.1)
Requirement already satisfied: pandas in d:\anaconda\lib\site-packages (1.2.4)
Requirement already satisfied: matplotlib in d:\anaconda\lib\site-packages (3.3.4)
Requirement already satisfied: pillow>=6.2.0 in d:\anaconda\lib\site-packages (from matp
lotlib) (8.2.0)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in d:\anaconda\l
ib\site-packages (from matplotlib) (2.4.7)
Requirement already satisfied: python-dateutil>=2.1 in d:\anaconda\lib\site-packages (fr
om matplotlib) (2.8.1)
Requirement already satisfied: cyclor>=0.10 in d:\anaconda\lib\site-packages (from matpl
otlib) (0.10.0)
Requirement already satisfied: kiwisolver>=1.0.1 in d:\anaconda\lib\site-packages (from
matplotlib) (1.3.1)
Requirement already satisfied: six in d:\anaconda\lib\site-packages (from cyclor>=0.10->
matplotlib) (1.15.0)
Requirement already satisfied: pytz>=2017.3 in d:\anaconda\lib\site-packages (from panda
s) (2021.1)
```

In [5]: `import numpy as np ; import pandas as pd`

In [7]: `arr = np.random.rand(10000, 10)`
`print(arr)`

```
[[0.87008356 0.2530678 0.36602123 ... 0.3042197 0.31991672 0.74376233]
 [0.32384194 0.73171479 0.29703702 ... 0.64590671 0.41837663 0.38168487]
 [0.4903878 0.26032612 0.54354787 ... 0.00865212 0.15029724 0.23832878]
 ...
 [0.96736171 0.8456456 0.82374085 ... 0.57169725 0.99679201 0.11813336]
 [0.1075936 0.81936957 0.34312011 ... 0.64848741 0.57621855 0.36537816]
 [0.99279153 0.8598557 0.84557286 ... 0.9381265 0.62945322 0.86181099]]
```

Data Frame

In [8]: `df = pd.DataFrame(arr)`
`df`

Out[8]:

	0	1	2	3	4	5	6	7	8	9
0	0.870084	0.253068	0.366021	0.139006	0.228586	0.036814	0.744692	0.304220	0.319917	0.743762
1	0.323842	0.731715	0.297037	0.730139	0.040181	0.836412	0.873667	0.645907	0.418377	0.381685
2	0.490388	0.260326	0.543548	0.592135	0.794590	0.400072	0.743808	0.008652	0.150297	0.238329
3	0.034133	0.711402	0.194532	0.639019	0.984384	0.252967	0.844040	0.479505	0.034654	0.315026
4	0.622396	0.691759	0.448588	0.023834	0.870275	0.094391	0.161510	0.080294	0.679568	0.347895
...
9995	0.745416	0.425762	0.575949	0.424995	0.352951	0.957898	0.220255	0.128817	0.979272	0.442010
9996	0.101936	0.406111	0.203203	0.929655	0.159630	0.323379	0.653097	0.403361	0.390412	0.748682
9997	0.967362	0.845646	0.823741	0.113634	0.223949	0.945657	0.844360	0.571697	0.996792	0.118133

	0	1	2	3	4	5	6	7	8	9
9998	0.107594	0.819370	0.343120	0.076203	0.578291	0.878705	0.053677	0.648487	0.576219	0.365378
9999	0.992792	0.859856	0.845573	0.115691	0.100936	0.389209	0.977167	0.938127	0.629453	0.861811

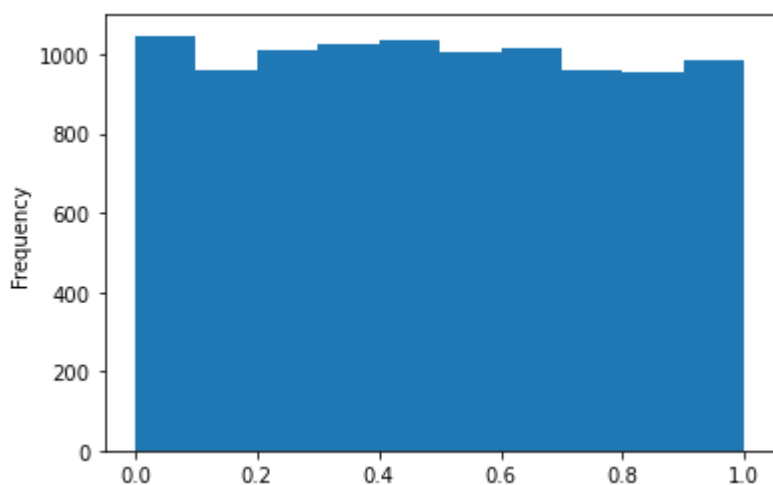
10000 rows × 10 columns



Histogram

In [37]: `df[9].plot.hist()`

Out[37]: `<AxesSubplot:ylabel='Frequency'>`



converting Data Frame to csv

In [48]: `df.to_csv('assignment1.csv', index=False)`

In [49]: `pd.read_csv('assignment1.csv')`

Out[49]:

	0	1	2	3	4	5	6	7	8	9
0	0.870084	0.253068	0.366021	0.139006	0.228586	0.036814	0.744692	0.304220	0.319917	0.743762
1	0.323842	0.731715	0.297037	0.730139	0.040181	0.836412	0.873667	0.645907	0.418377	0.381685
2	0.490388	0.260326	0.543548	0.592135	0.794590	0.400072	0.743808	0.008652	0.150297	0.238329
3	0.034133	0.711402	0.194532	0.639019	0.984384	0.252967	0.844040	0.479505	0.034654	0.315026
4	0.622396	0.691759	0.448588	0.023834	0.870275	0.094391	0.161510	0.080294	0.679568	0.347895
...
9995	0.745416	0.425762	0.575949	0.424995	0.352951	0.957898	0.220255	0.128817	0.979272	0.442010
9996	0.101936	0.406111	0.203203	0.929655	0.159630	0.323379	0.653097	0.403361	0.390412	0.748682

	0	1	2	3	4	5	6	7	8	9
9997	0.967362	0.845646	0.823741	0.113634	0.223949	0.945657	0.844360	0.571697	0.996792	0.118133
9998	0.107594	0.819370	0.343120	0.076203	0.578291	0.878705	0.053677	0.648487	0.576219	0.365378
9999	0.992792	0.859856	0.845573	0.115691	0.100936	0.389209	0.977167	0.938127	0.629453	0.861811

10000 rows × 10 columns



In []: