

```
In [1]: import pandas as pd
```

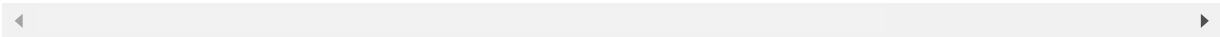
Dataset-1

```
In [2]: df=pd.read_csv(r"C:\Users\user\Downloads\1_fiat500_VehicleSelection_Dataset.csv")
df
```

Out[2]:

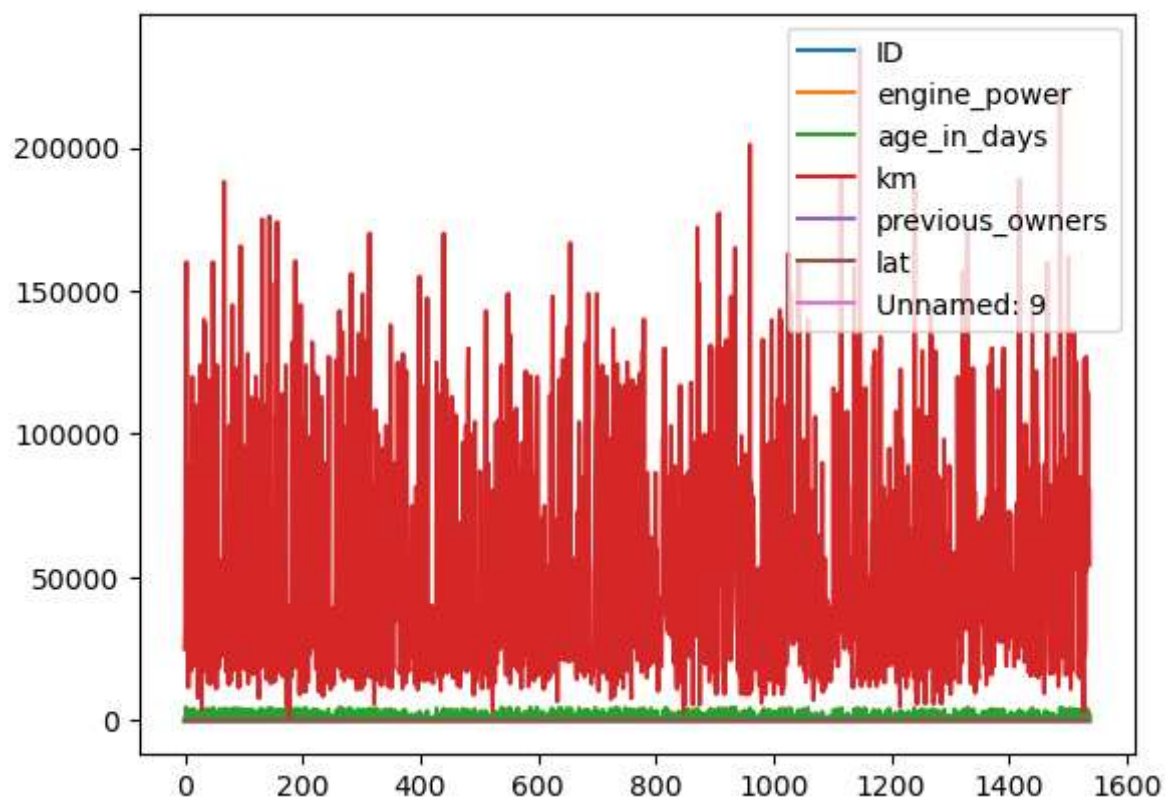
	ID	model	engine_power	age_in_days	km	previous_owners	lat	lon
0	1.0	lounge	51.0	882.0	25000.0	1.0	44.907242	8.6115598
1	2.0	pop	51.0	1186.0	32500.0	1.0	45.666359	12.241889
2	3.0	sport	74.0	4658.0	142228.0	1.0	45.503300	11.417
3	4.0	lounge	51.0	2739.0	160000.0	1.0	40.633171	17.634609
4	5.0	pop	73.0	3074.0	106880.0	1.0	41.903221	12.495650
...
1544	NaN	NaN	NaN	NaN	NaN	NaN	NaN	len
1545	NaN	NaN	NaN	NaN	NaN	NaN	NaN	con
1546	NaN	NaN	NaN	NaN	NaN	NaN	NaN	Null valu
1547	NaN	NaN	NaN	NaN	NaN	NaN	NaN	fi
1548	NaN	NaN	NaN	NaN	NaN	NaN	NaN	sear

1549 rows × 11 columns



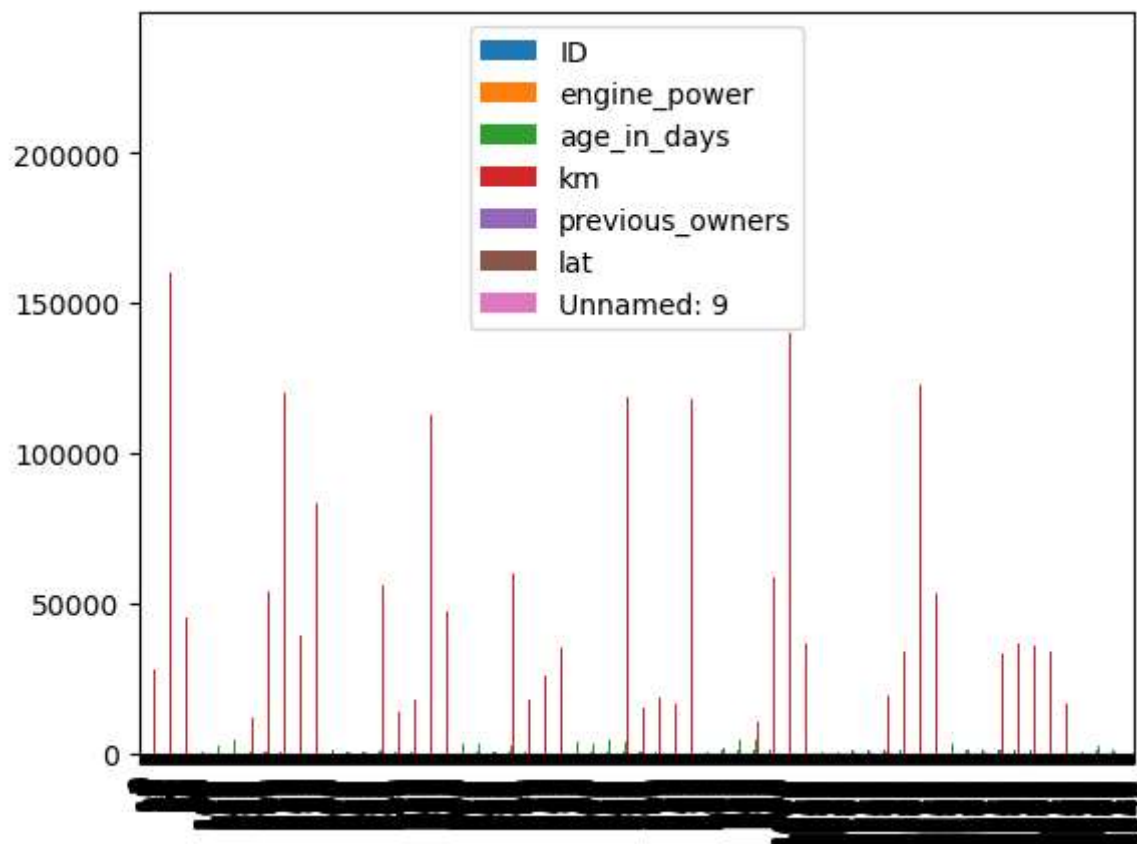
```
In [3]: df.plot.line()
```

```
Out[3]: <Axes: >
```



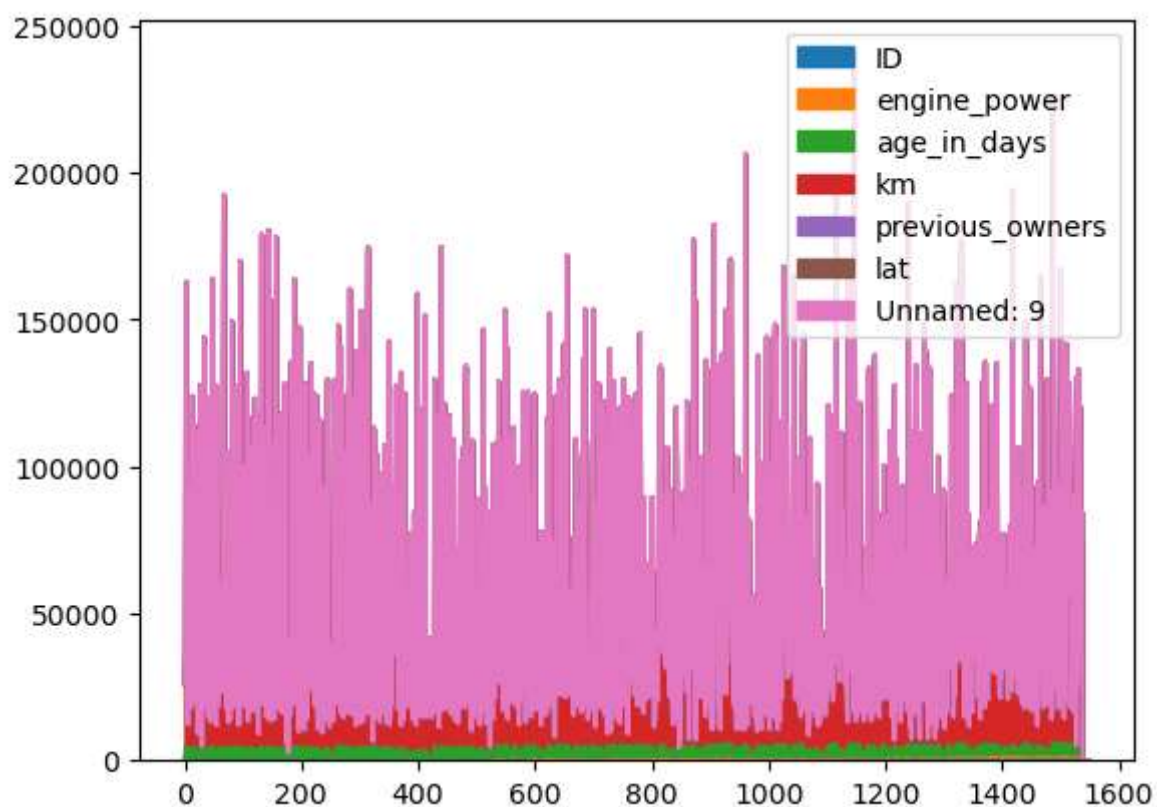
```
In [4]: df.plot.bar()
```

```
Out[4]: <Axes: >
```



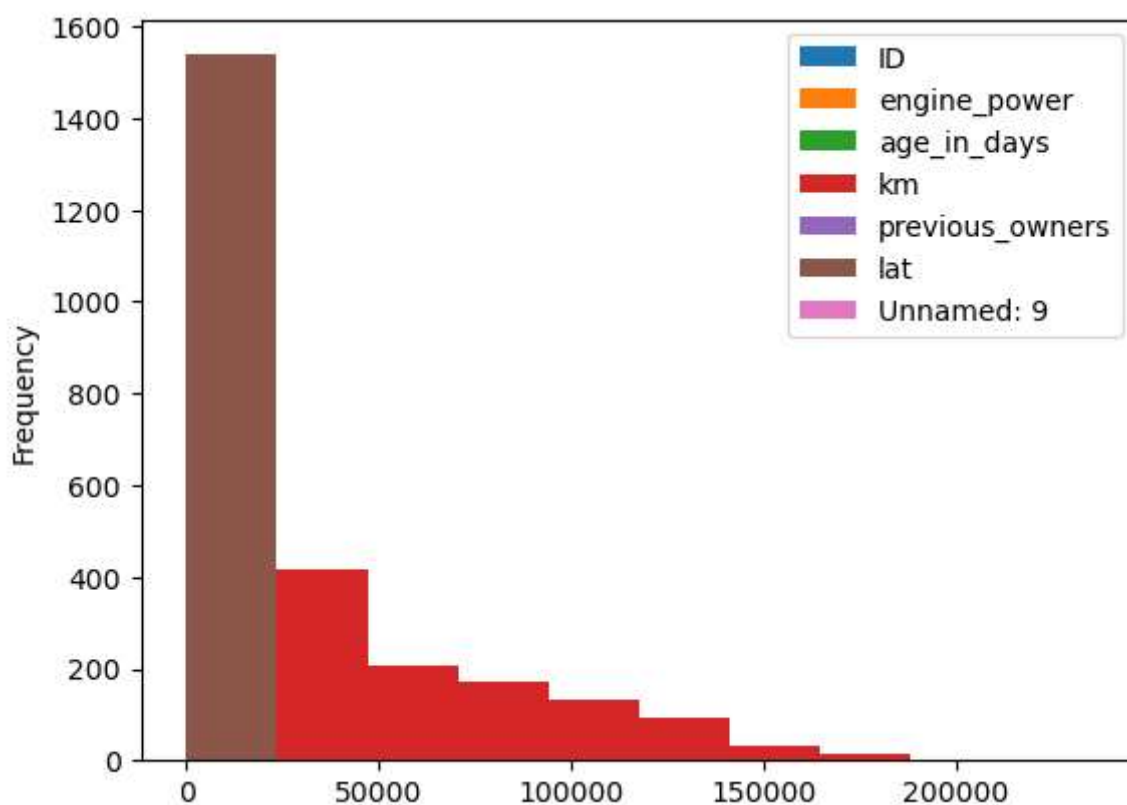
```
In [5]: df.plot.area()
```

```
Out[5]: <Axes: >
```



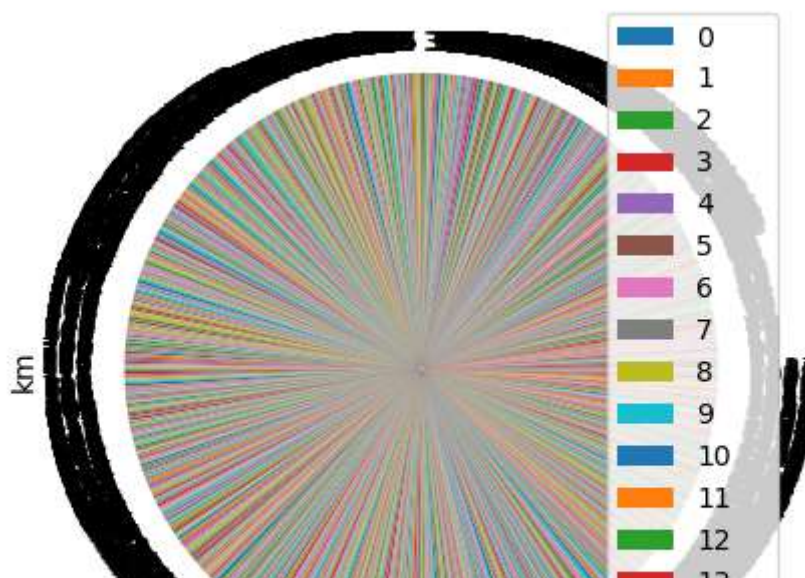
```
In [6]: df.plot.hist()
```

```
Out[6]: <Axes: ylabel='Frequency'>
```



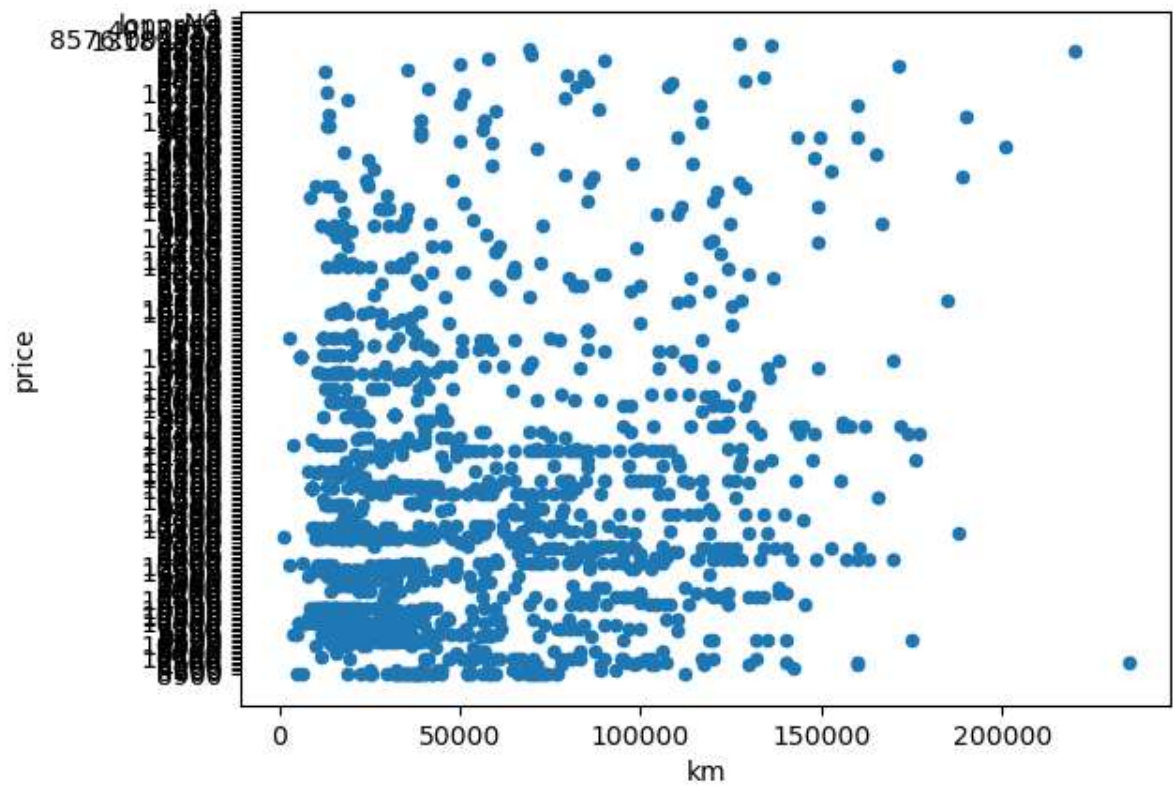
```
In [7]: df.plot.pie(y='km')
```

```
Out[7]: <Axes: ylabel='km'>
```



```
In [8]: df.plot.scatter(x='km',y='price')
```

```
Out[8]: <Axes: xlabel='km', ylabel='price'>
```



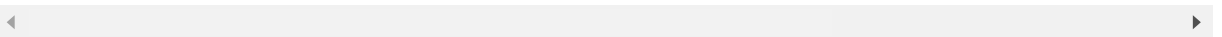
Dataset-2

```
In [9]: df1=pd.read_csv(r"C:\Users\user\Downloads\2_2015.csv")
df1
```

Out[9]:

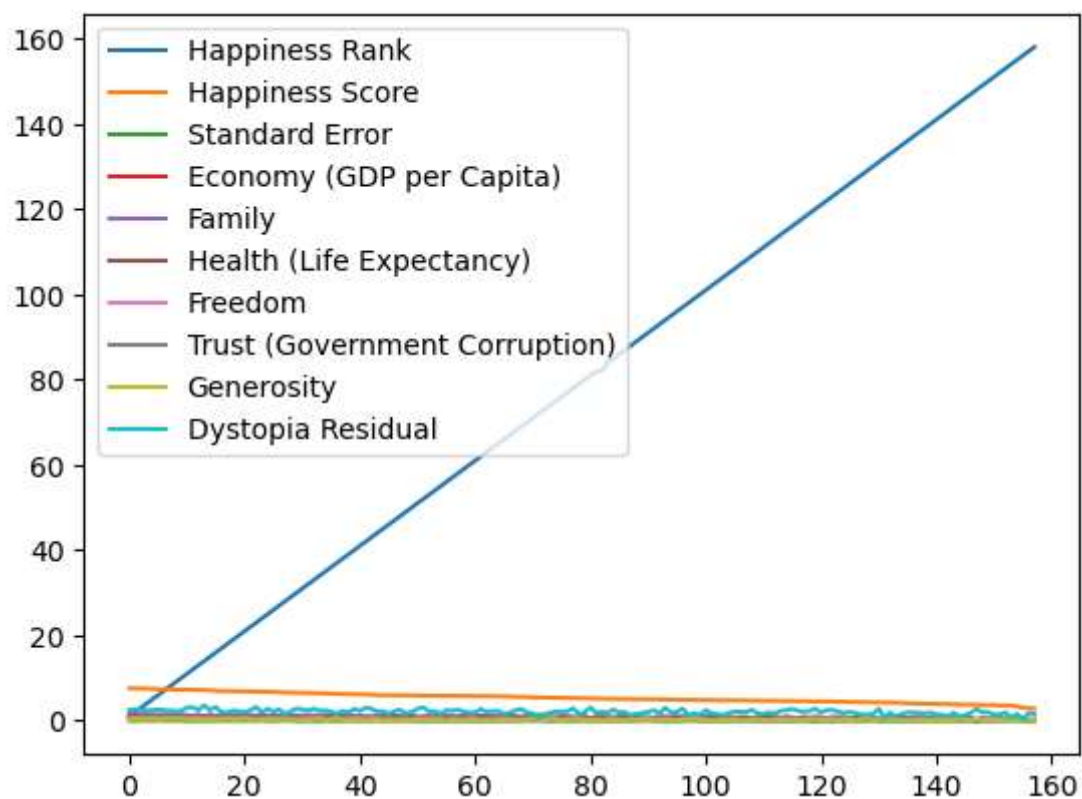
	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Fre
0	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143	0.1
1	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784	0.1
2	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464	0.1
3	Norway	Western Europe	4	7.522	0.03880	1.45900	1.33095	0.88521	0.1
4	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563	0.1
...
153	Rwanda	Sub-Saharan Africa	154	3.465	0.03464	0.22208	0.77370	0.42864	0.1
154	Benin	Sub-Saharan Africa	155	3.340	0.03656	0.28665	0.35386	0.31910	0.1
155	Syria	Middle East and Northern Africa	156	3.006	0.05015	0.66320	0.47489	0.72193	0.1
156	Burundi	Sub-Saharan Africa	157	2.905	0.08658	0.01530	0.41587	0.22396	0.1
157	Togo	Sub-Saharan Africa	158	2.839	0.06727	0.20868	0.13995	0.28443	0.1

158 rows × 12 columns



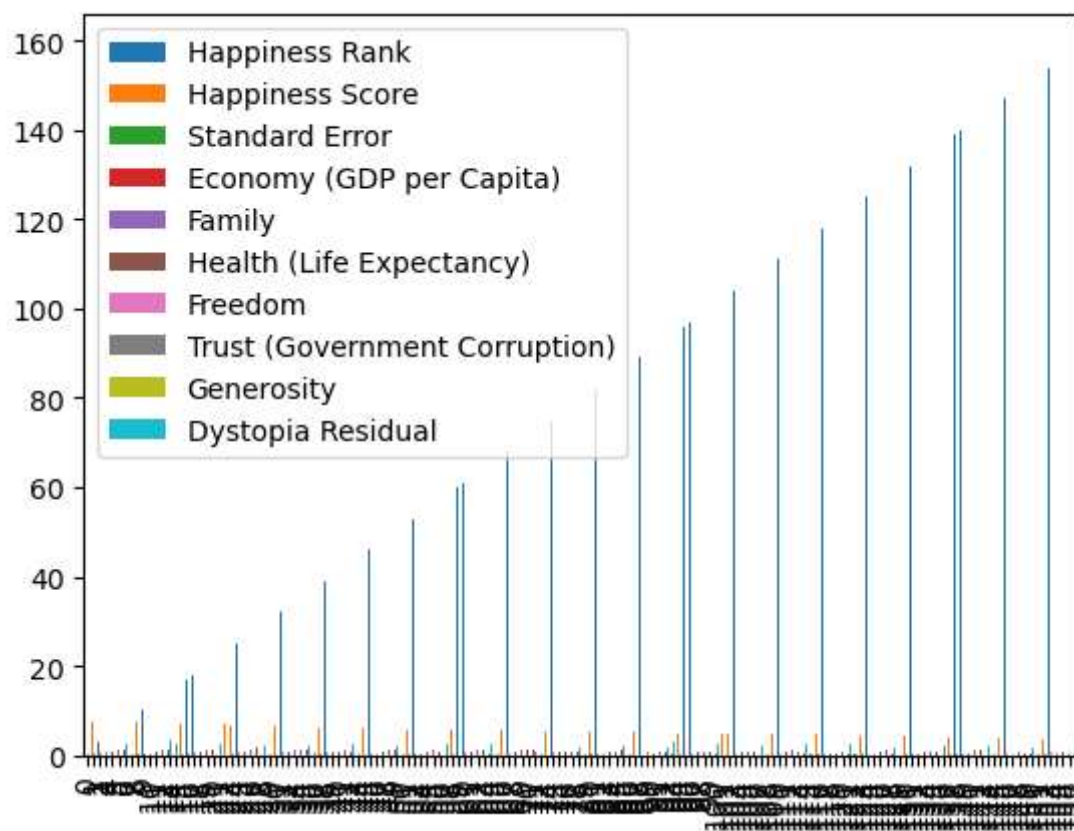
```
In [10]: df1.plot.line()
```

```
Out[10]: <Axes: >
```



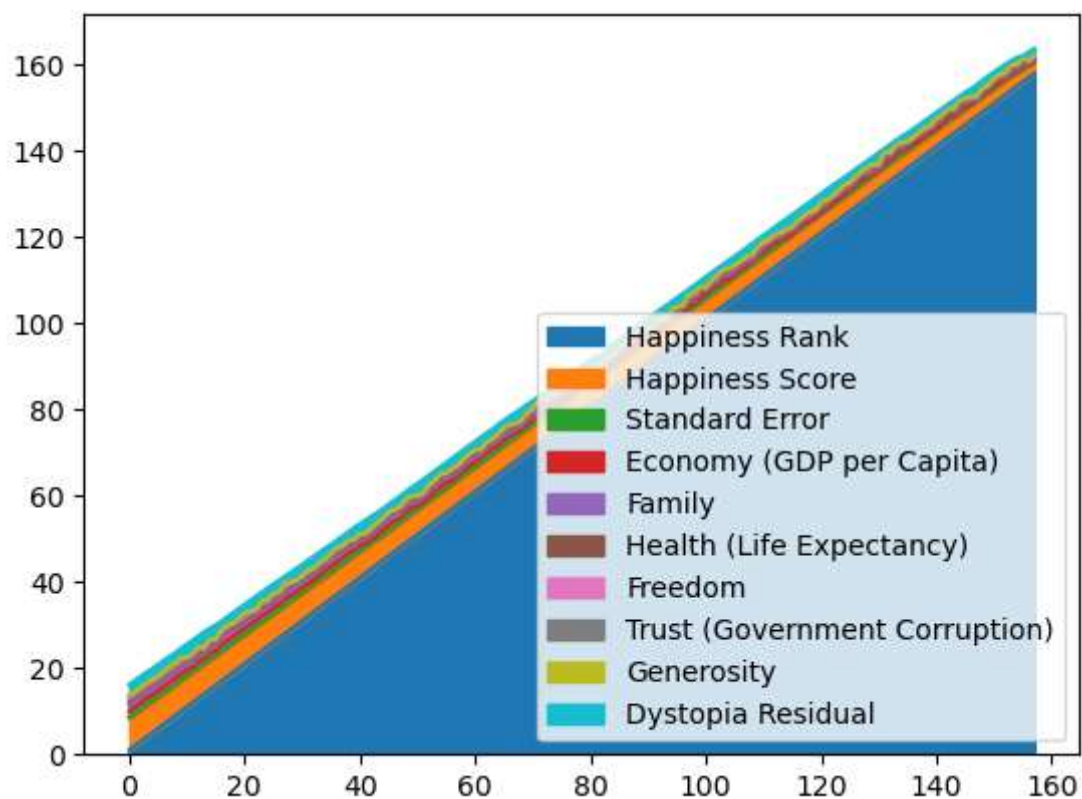

```
In [11]: df1.plot.bar()
```

```
Out[11]: <Axes: >
```



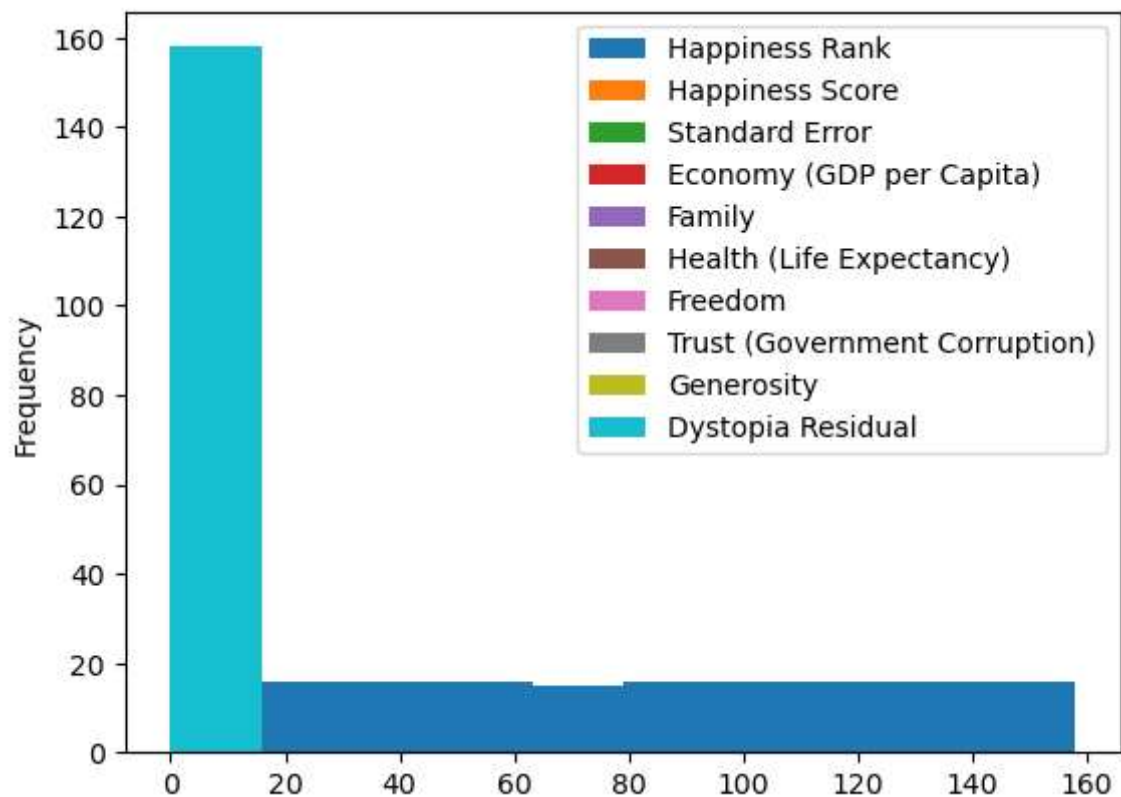
```
In [12]: df1.plot.area()
```

```
Out[12]: <Axes: >
```



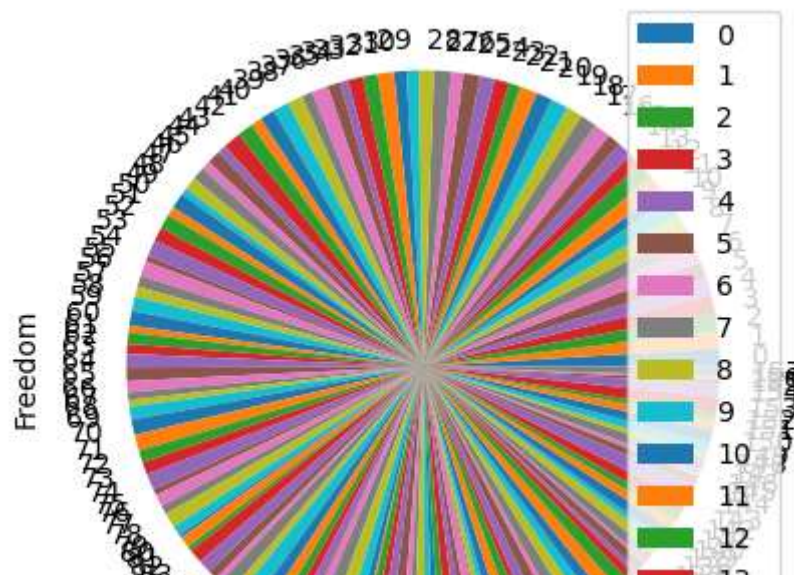
```
In [13]: df1.plot.hist()
```

```
Out[13]: <Axes: ylabel='Frequency'>
```



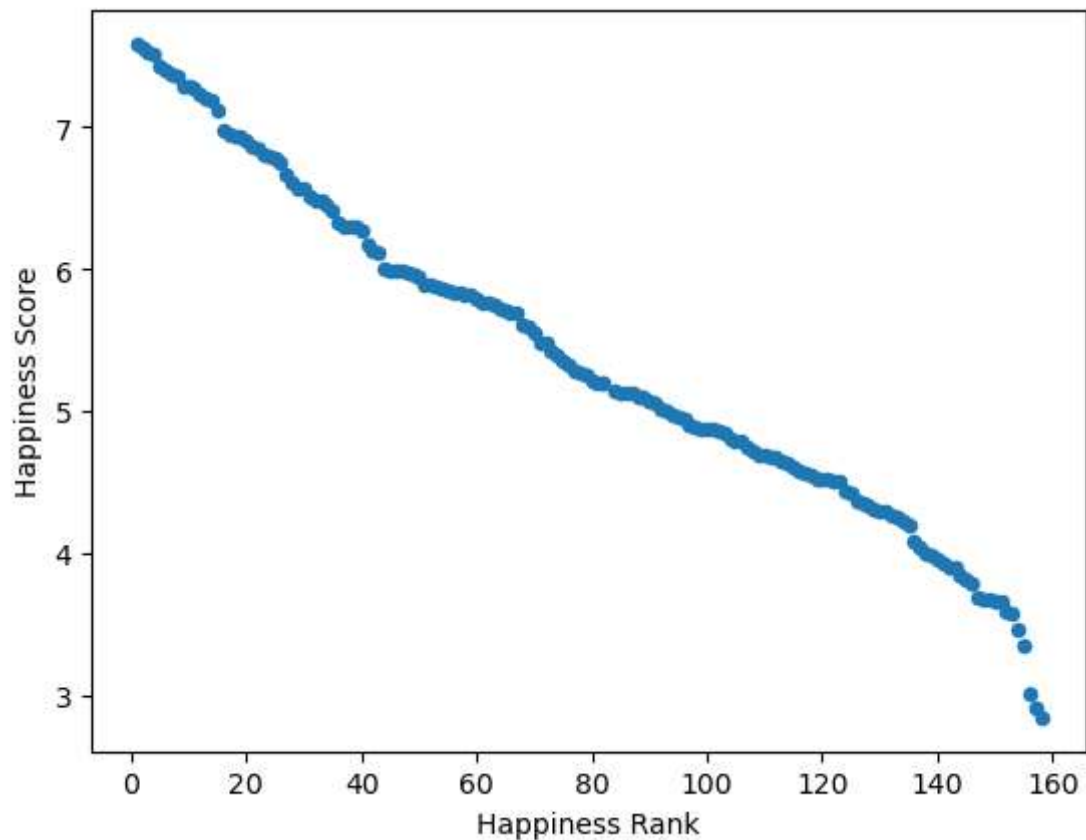
```
In [14]: df1.plot.pie(y='Freedom')
```

```
Out[14]: <Axes: ylabel='Freedom'>
```



```
In [15]: df1.plot.scatter(x='Happiness Rank',y='Happiness Score')
```

```
Out[15]: <Axes: xlabel='Happiness Rank', ylabel='Happiness Score'>
```



Dataset-3

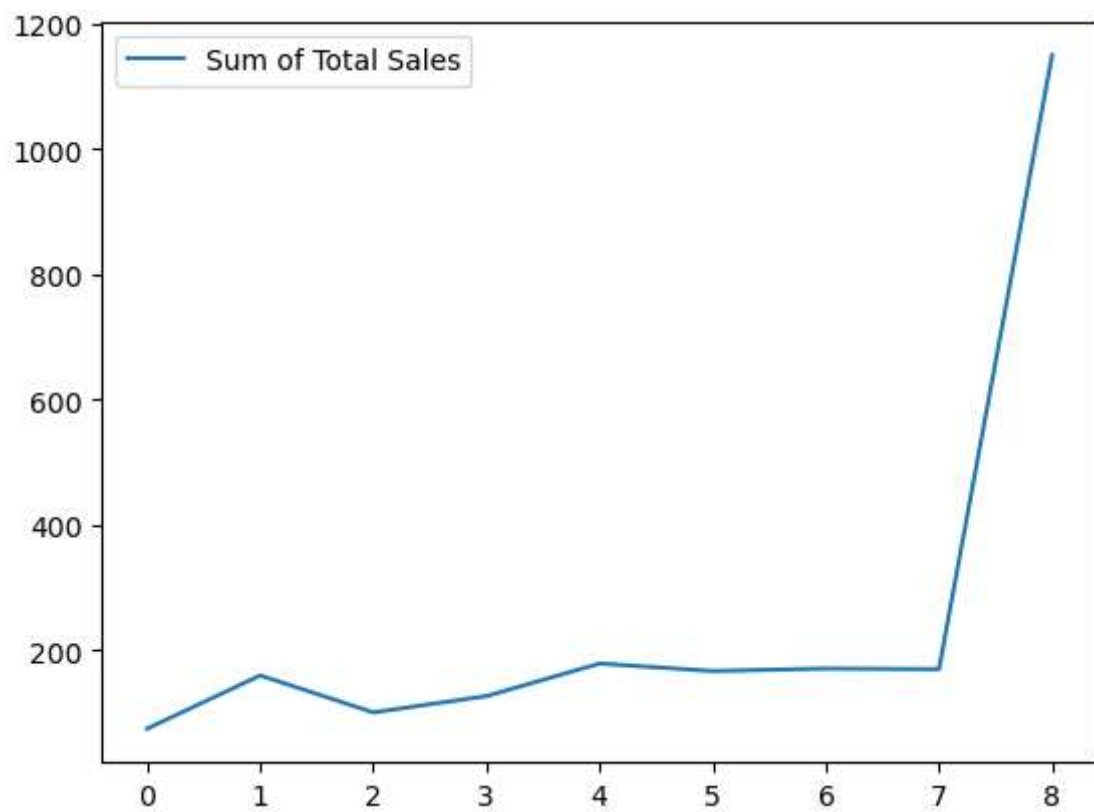
```
In [17]: df2=pd.read_csv(r"C:\Users\user\Downloads\3_Fitness-1.csv")
df2
```

```
Out[17]:
```

	Row Labels	Sum of Jan	Sum of Feb	Sum of Mar	Sum of Total Sales
0	A	5.62%	7.73%	6.16%	75
1	B	4.21%	17.27%	19.21%	160
2	C	9.83%	11.60%	5.17%	101
3	D	2.81%	21.91%	7.88%	127
4	E	25.28%	10.57%	11.82%	179
5	F	8.15%	16.24%	18.47%	167
6	G	18.54%	8.76%	17.49%	171
7	H	25.56%	5.93%	13.79%	170
8	Grand Total	100.00%	100.00%	100.00%	1150

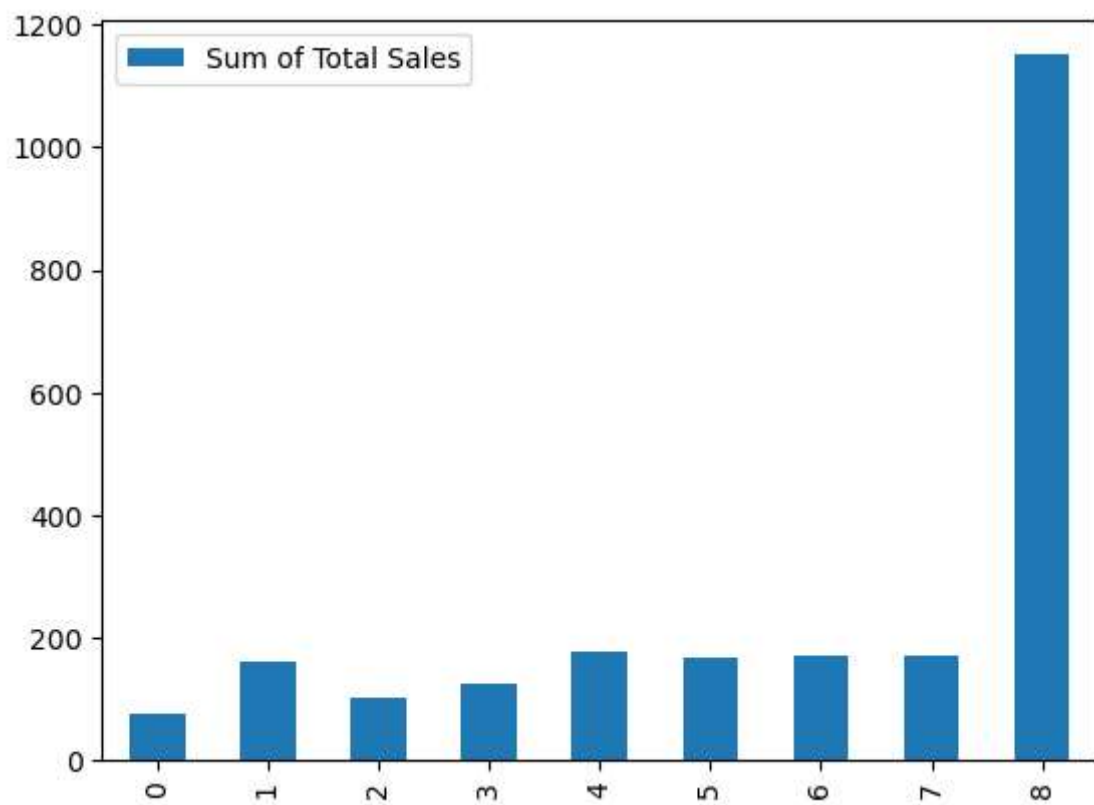
```
In [19]: df2.plot.line()
```

```
Out[19]: <Axes: >
```



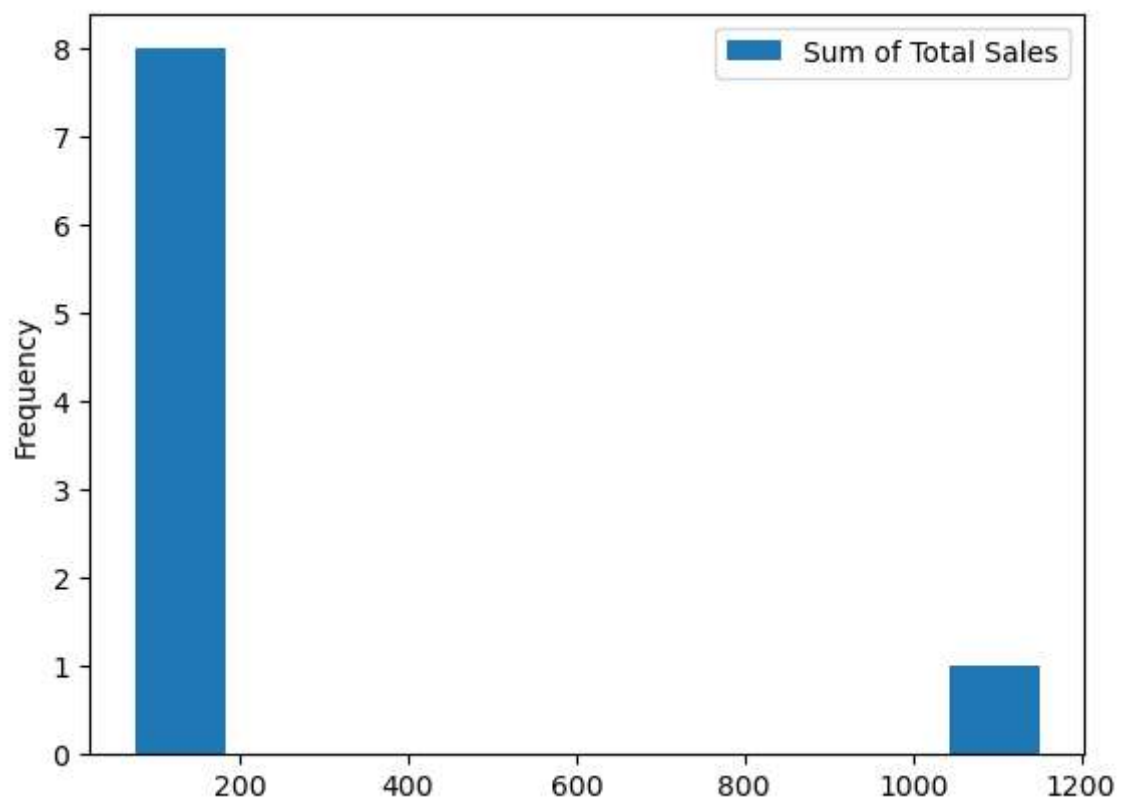
```
In [20]: df2.plot.bar()
```

```
Out[20]: <Axes: >
```



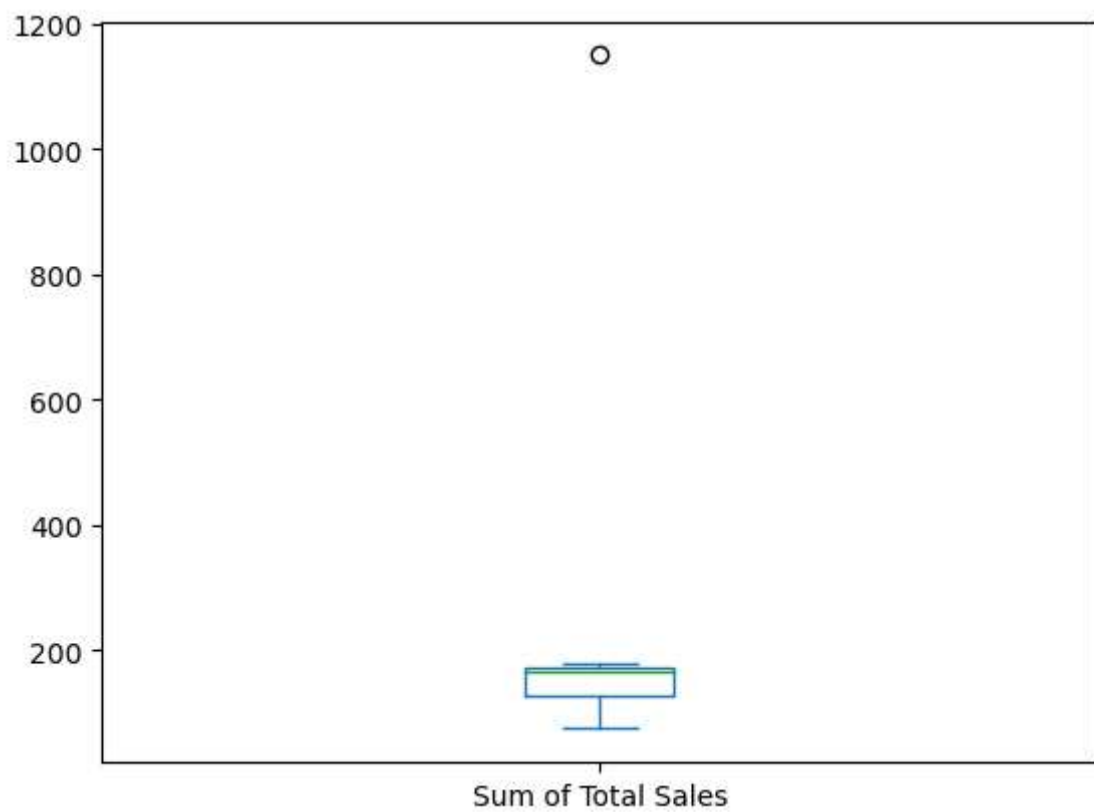
```
In [21]: df2.plot.hist()
```

```
Out[21]: <Axes: ylabel='Frequency'>
```



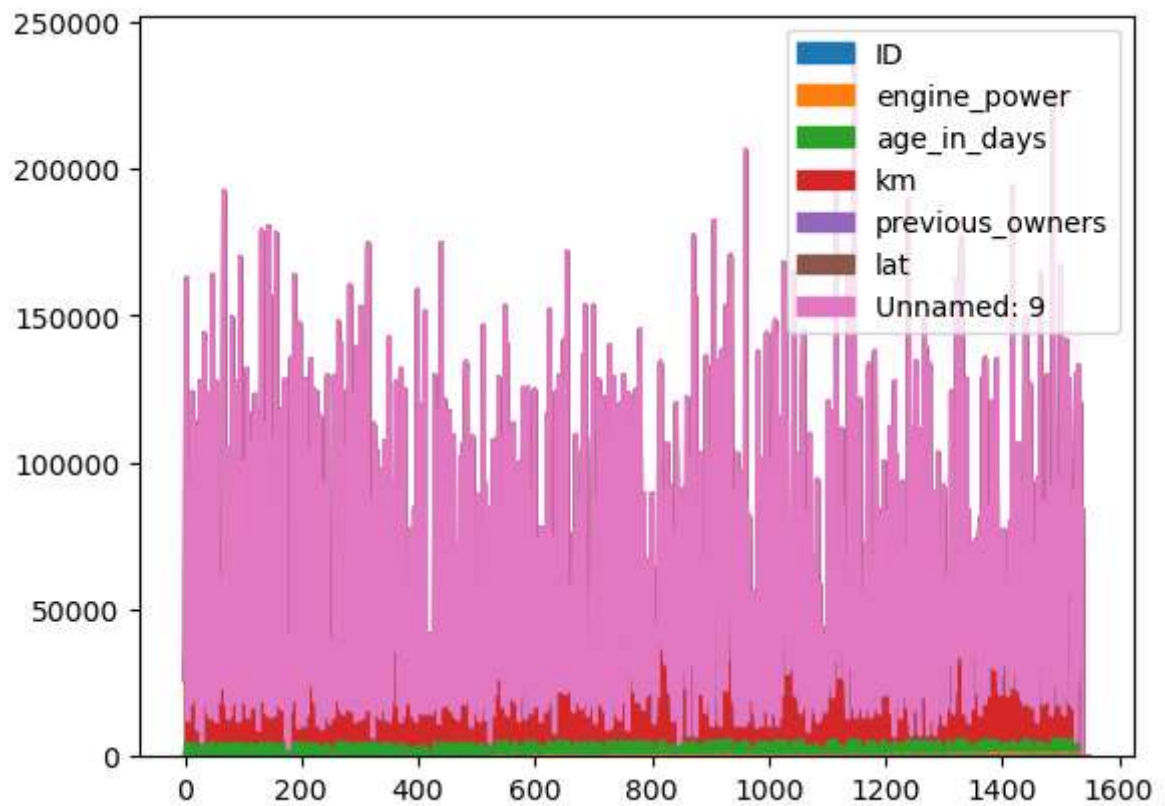
```
In [22]: df2.plot.box()
```

```
Out[22]: <Axes: >
```



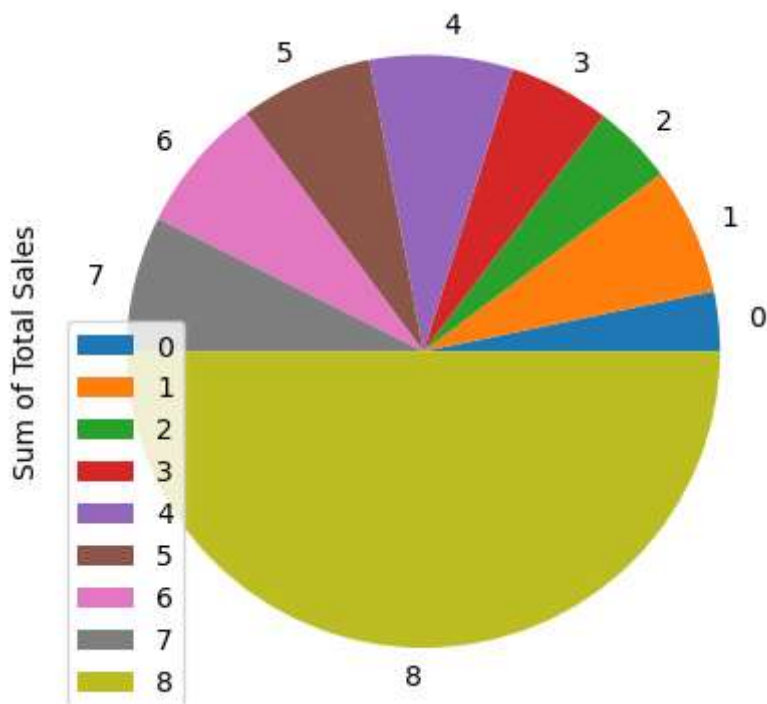

```
In [23]: df.plot.area()
```

```
Out[23]: <Axes: >
```



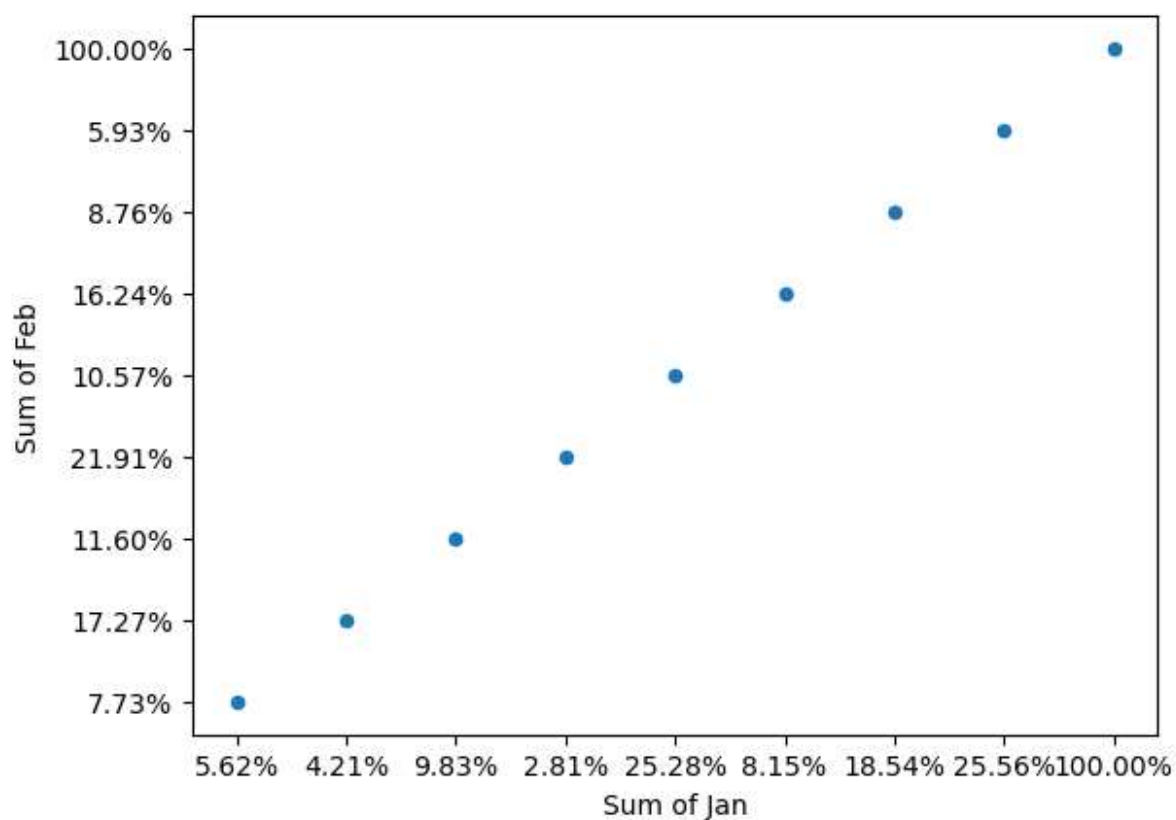
```
In [24]: df2.plot.pie(y='Sum of Total Sales')
```

```
Out[24]: <Axes: ylabel='Sum of Total Sales'>
```



```
In [25]: df2.plot.scatter(x='Sum of Jan',y='Sum of Feb')
```

```
Out[25]: <Axes: xlabel='Sum of Jan', ylabel='Sum of Feb'>
```



Dataset-4

```
In [26]: df3=pd.read_csv(r"C:\Users\user\Downloads\4_drug200.csv")
df3
```

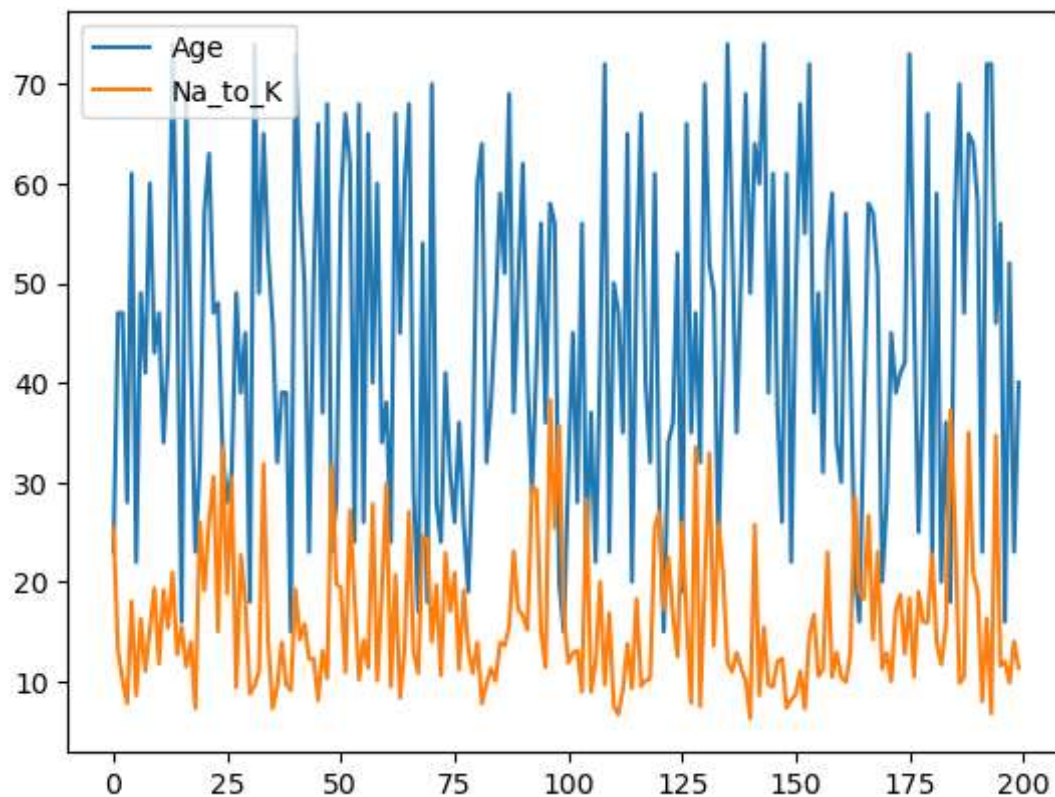
Out[26]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
...
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

200 rows × 6 columns

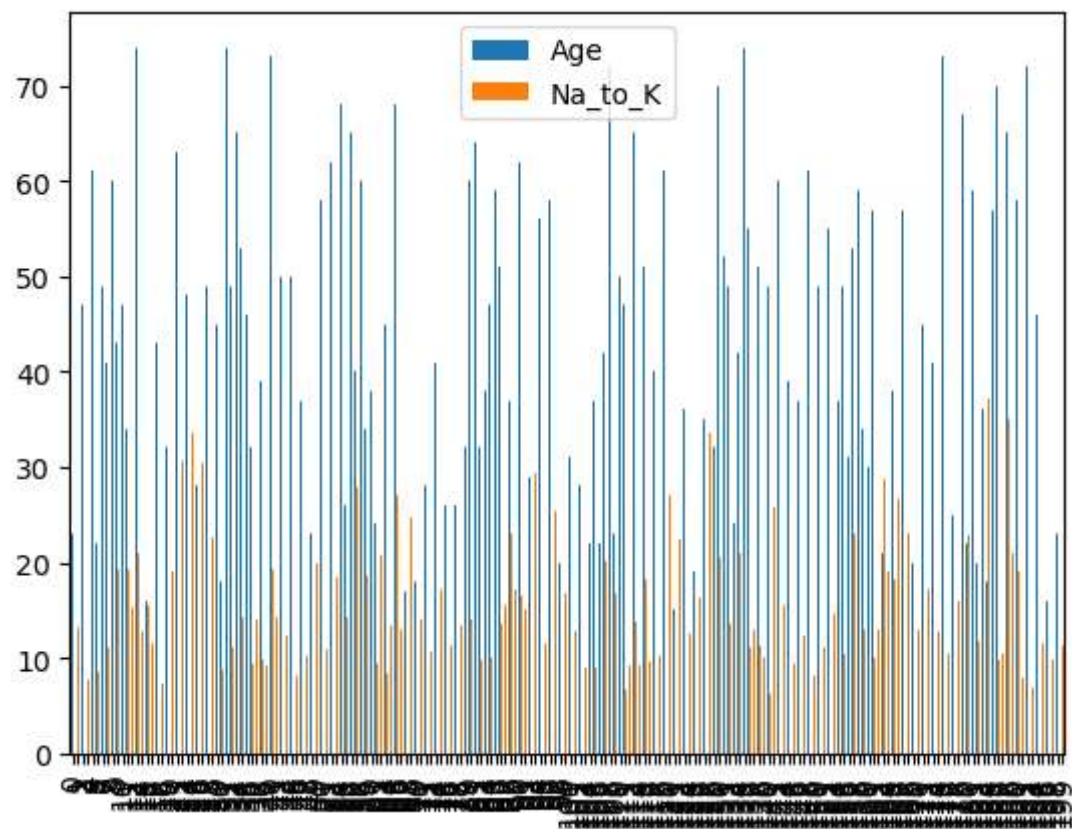
```
In [27]: df3.plot.line()
```

Out[27]: <Axes: >



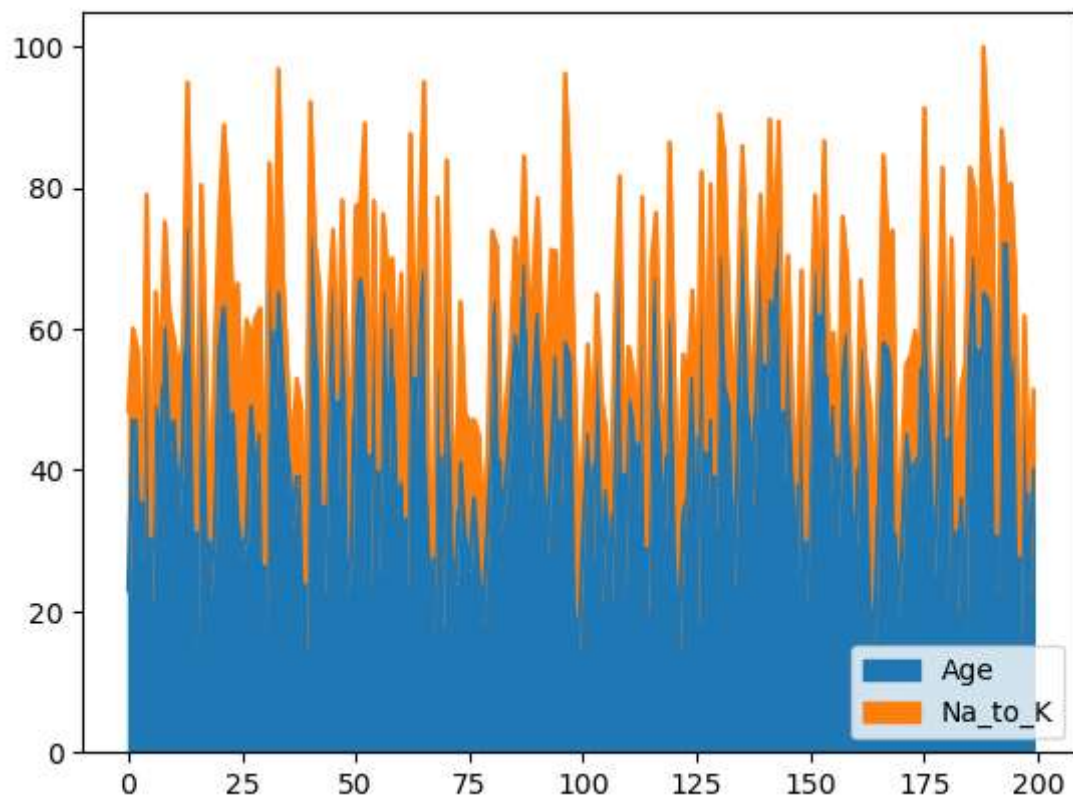
```
In [28]: df3.plot.bar()
```

```
Out[28]: <Axes: >
```



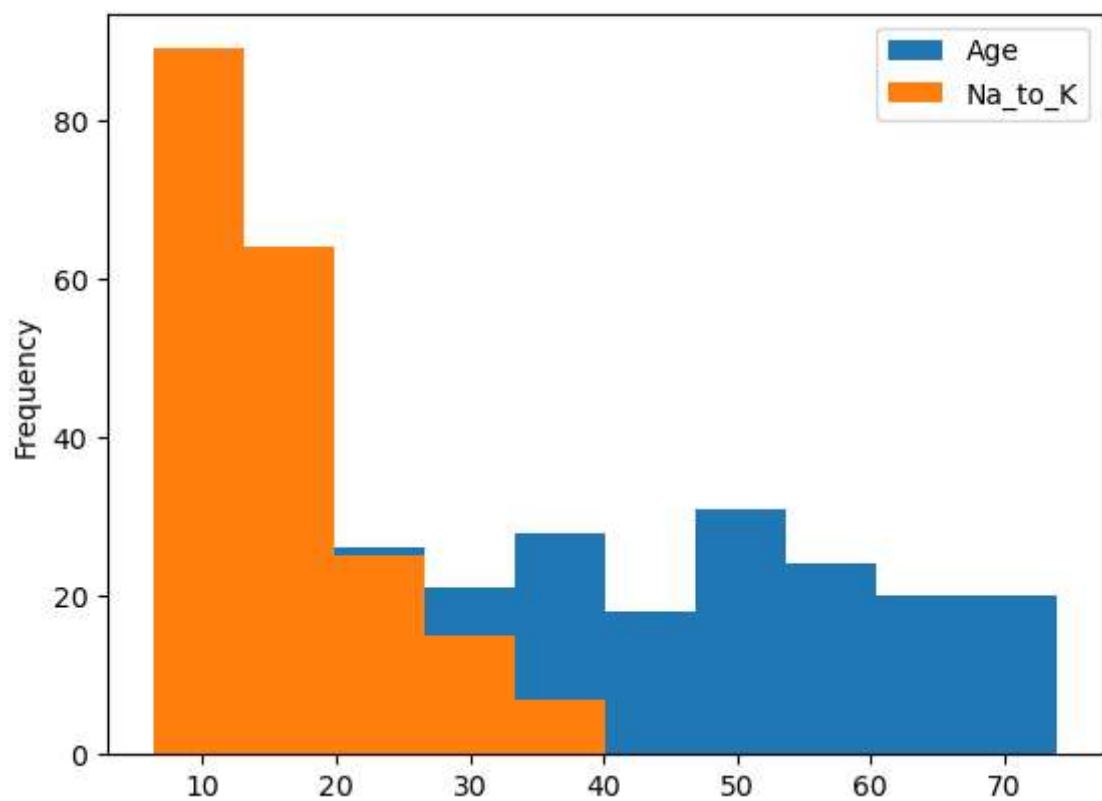
```
In [29]: df3.plot.area()
```

```
Out[29]: <Axes: >
```



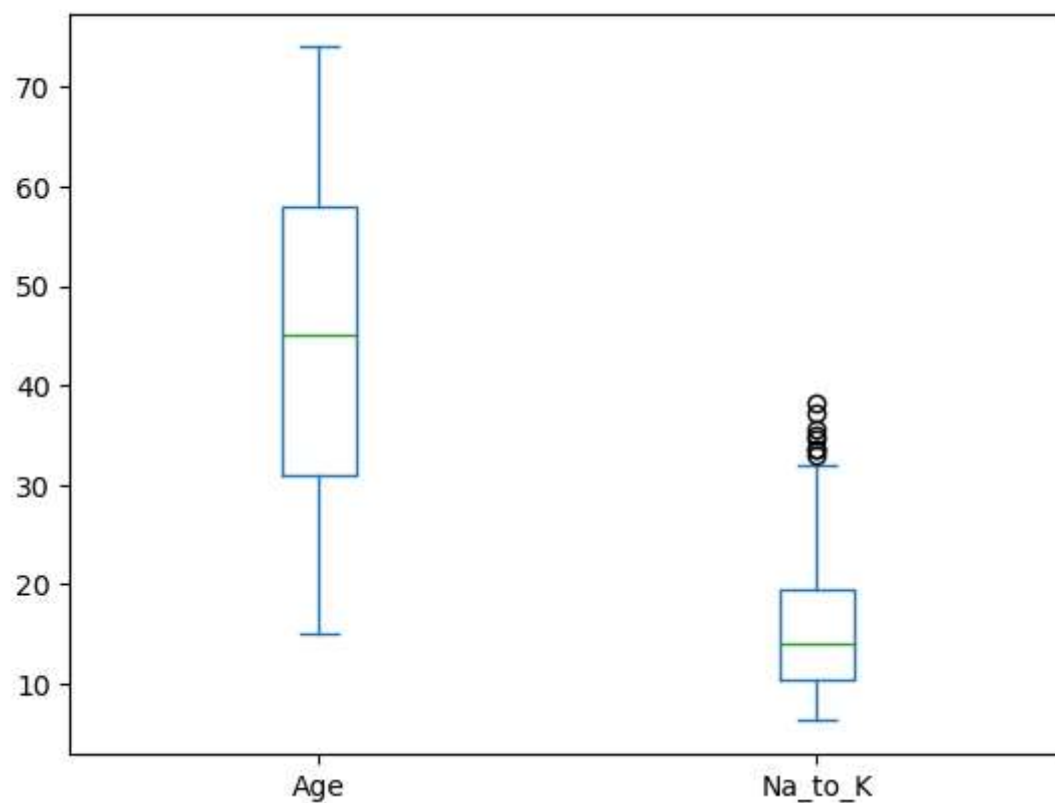
```
In [30]: df3.plot.hist()
```

```
Out[30]: <Axes: ylabel='Frequency'>
```



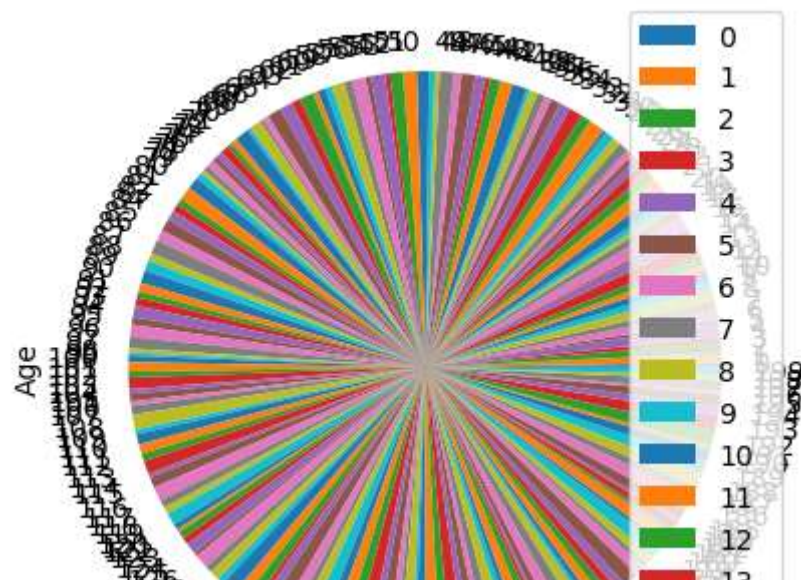
```
In [31]: df3.plot.box()
```

```
Out[31]: <Axes: >
```

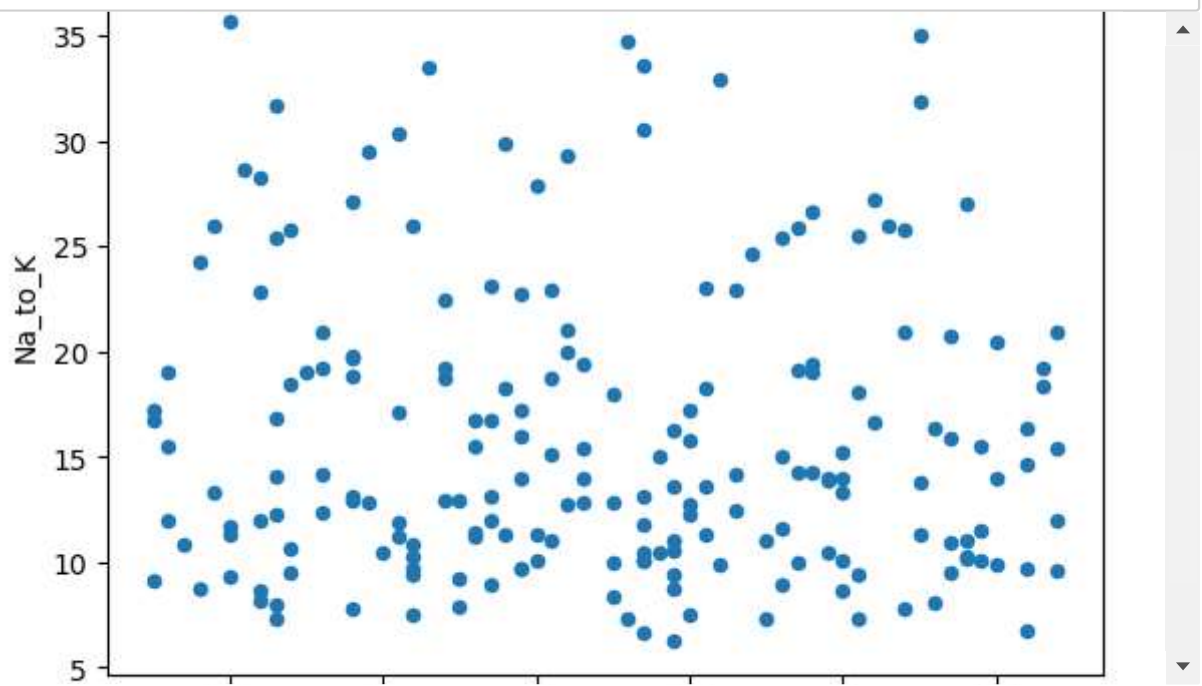


```
In [33]: df3.plot.pie(y='Age')
```

```
Out[33]: <Axes: ylabel='Age'>
```



```
In [35]: df3.plot.scatter(x='Age',y='Na_to_K')
```



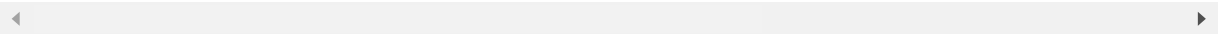
Dataset-5


```
In [59]: df4=pd.read_csv(r"C:\Users\user\Downloads\6_Salesworkload1.csv")
df4
```

Out[59]:

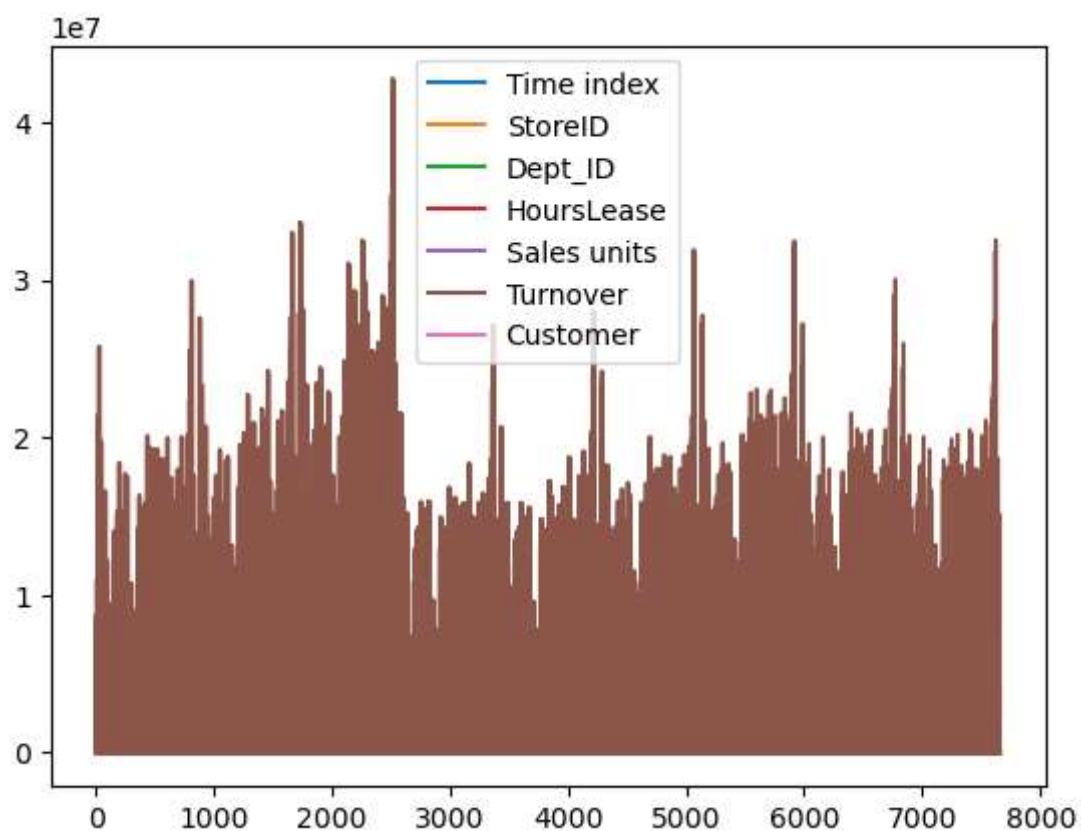
	MonthYear	Time index	Country	StoreID	City	Dept_ID	Dept. Name	HoursOwn	HoursLe
0	10.2016	1.0	United Kingdom	88253.0	London (I)	1.0	Dry	3184.764	
1	10.2016	1.0	United Kingdom	88253.0	London (I)	2.0	Frozen	1582.941	
2	10.2016	1.0	United Kingdom	88253.0	London (I)	3.0	other	47.205	
3	10.2016	1.0	United Kingdom	88253.0	London (I)	4.0	Fish	1623.852	
4	10.2016	1.0	United Kingdom	88253.0	London (I)	5.0	Fruits & Vegetables	1759.173	
...
7653	06.2017	9.0	Sweden	29650.0	Gothenburg	12.0	Checkout	6322.323	
7654	06.2017	9.0	Sweden	29650.0	Gothenburg	16.0	Customer Services	4270.479	
7655	06.2017	9.0	Sweden	29650.0	Gothenburg	11.0	Delivery	0	
7656	06.2017	9.0	Sweden	29650.0	Gothenburg	17.0	others	2224.929	
7657	06.2017	9.0	Sweden	29650.0	Gothenburg	18.0	all	39652.2	

7658 rows × 14 columns



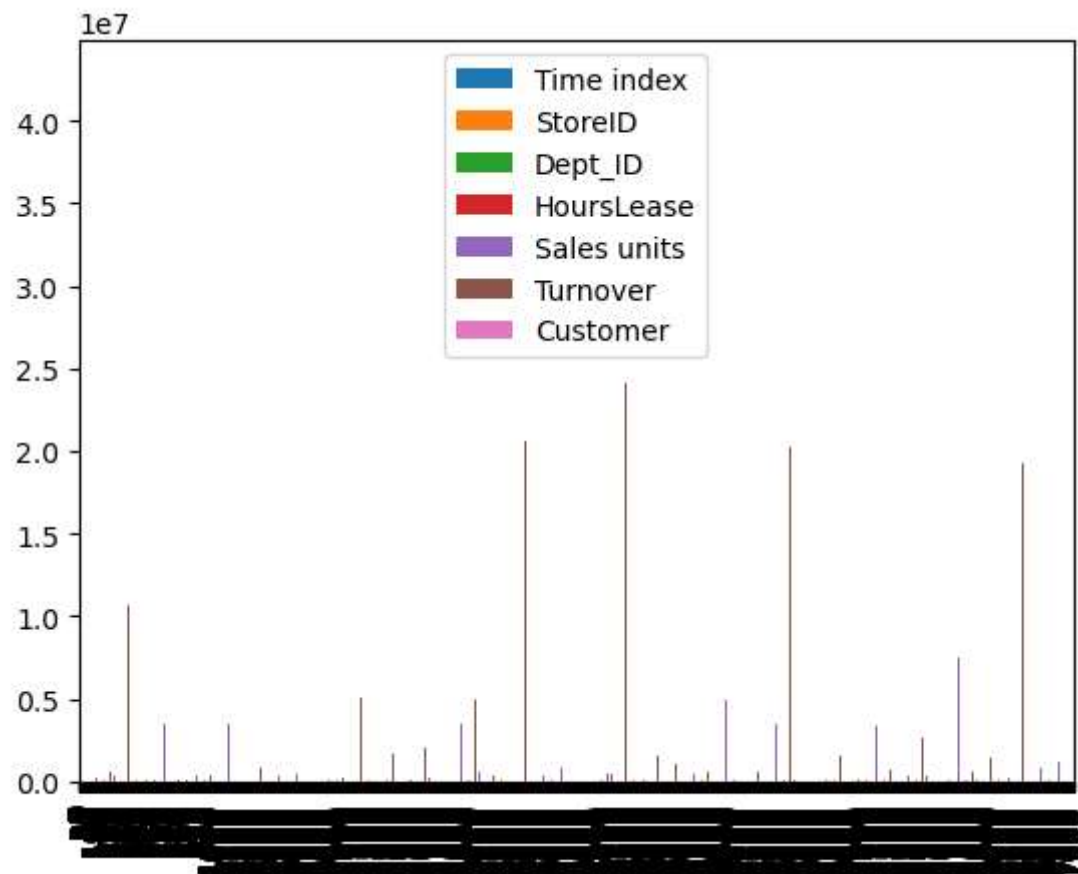
```
In [60]: df4.plot.line()
```

```
Out[60]: <Axes: >
```



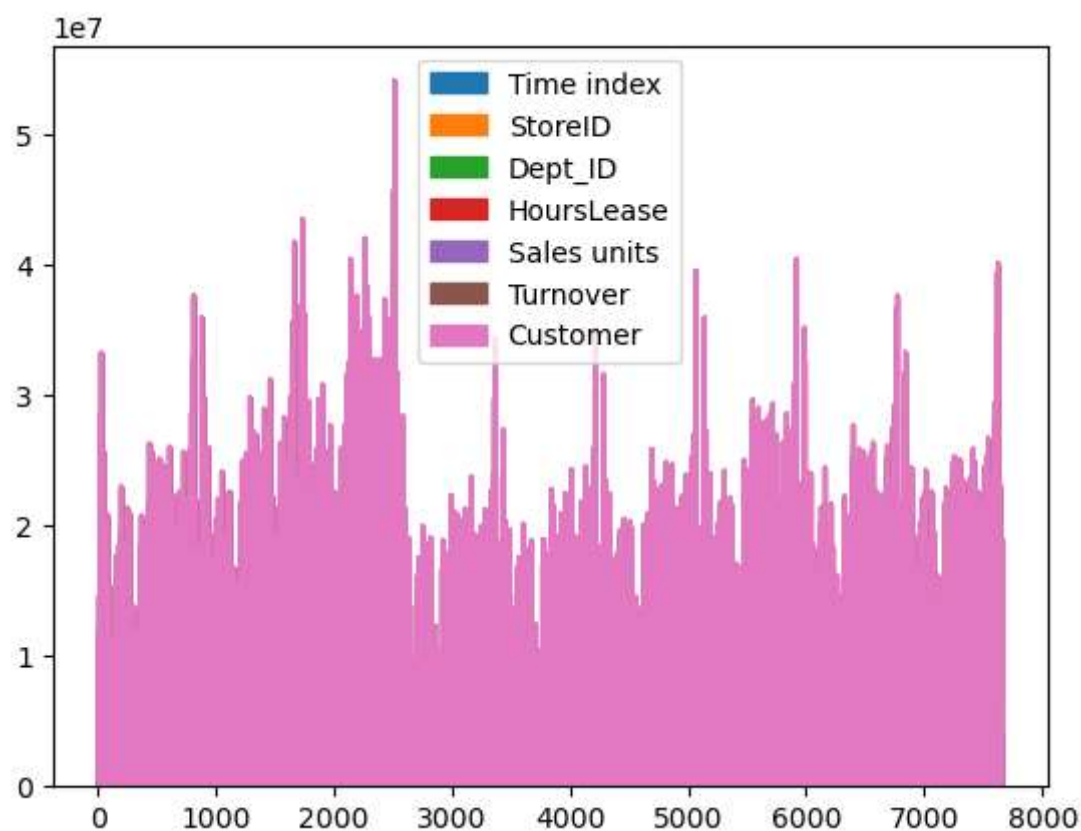
```
In [61]: df4.plot.bar()
```

```
Out[61]: <Axes: >
```



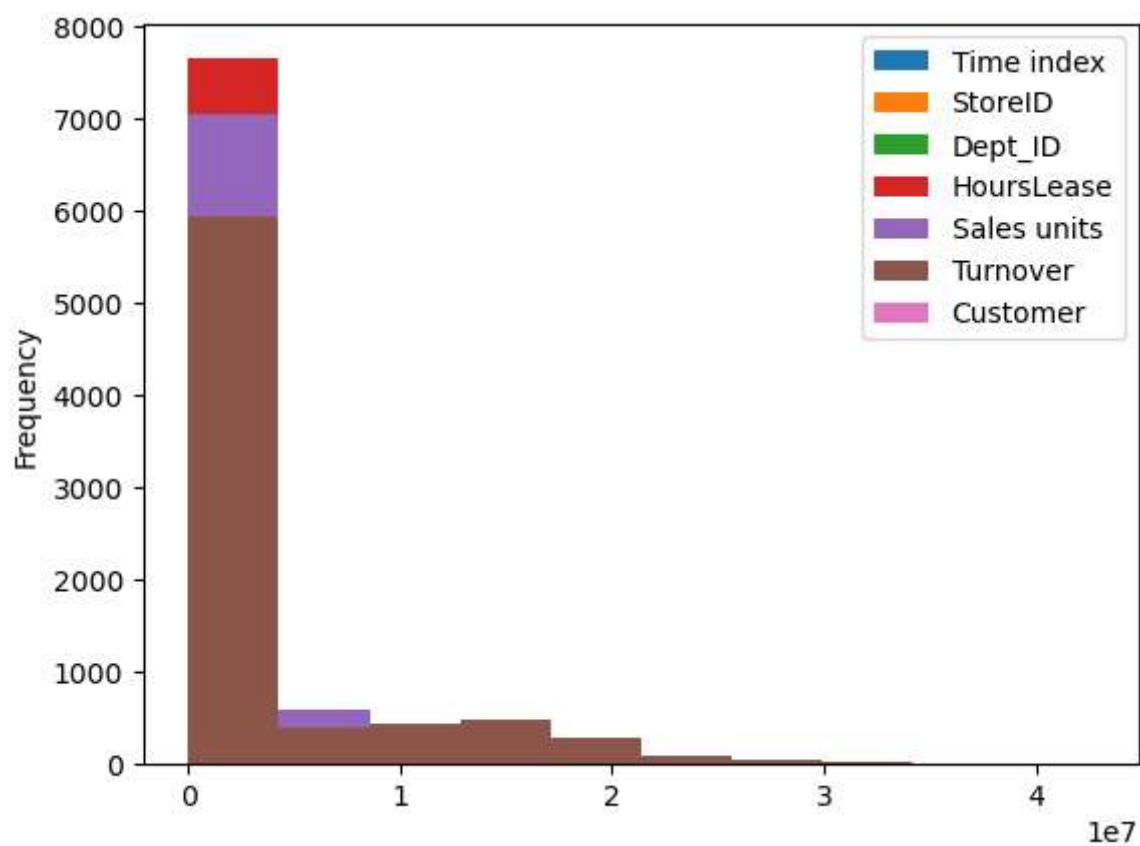
```
In [62]: df4.plot.area()
```

```
Out[62]: <Axes: >
```



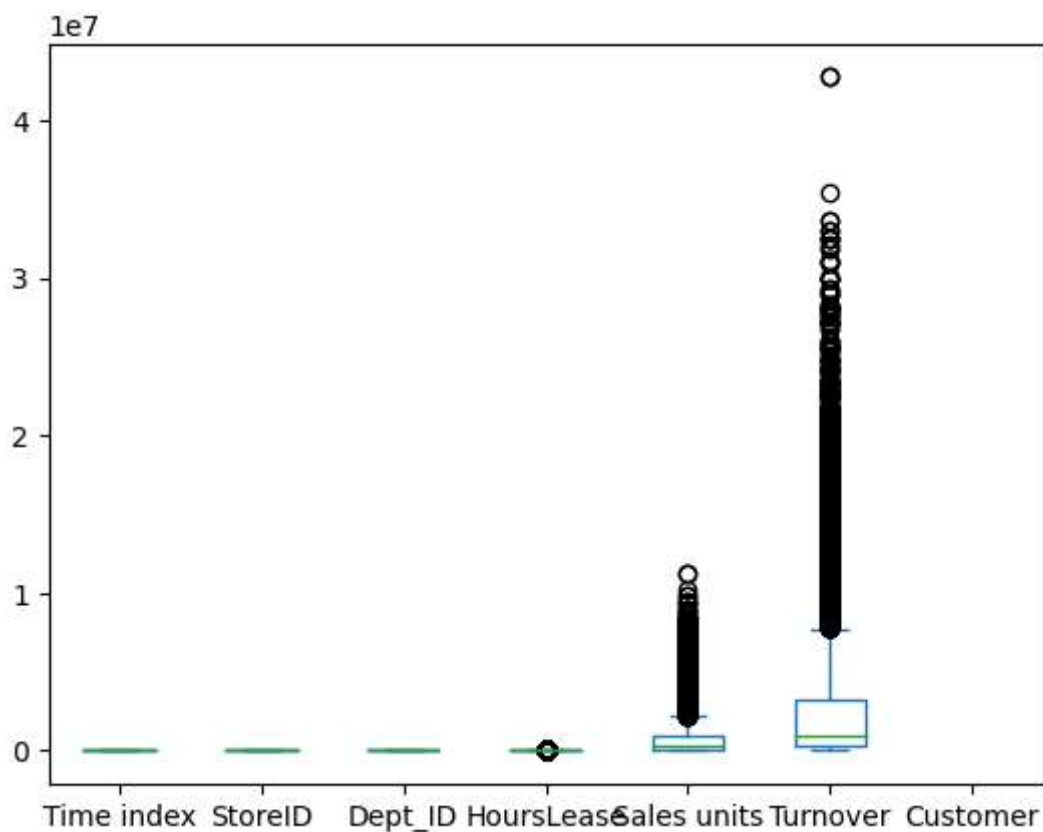
```
In [63]: df4.plot.hist()
```

```
Out[63]: <Axes: ylabel='Frequency'>
```



In [64]: df4.plot.box()

Out[64]: <Axes: >



In [68]: df4.plot.pie(y='Sales units')

```
In FigureCanvasAgg.get_renderer(self, cleared)
    409 reuse_renderer = (self._lastKey == key)
    410 if not reuse_renderer:
--> 411     self.renderer = RendererAgg(w, h, self.figure.dpi)
    412     self._lastKey = key
    413 elif cleared:
```

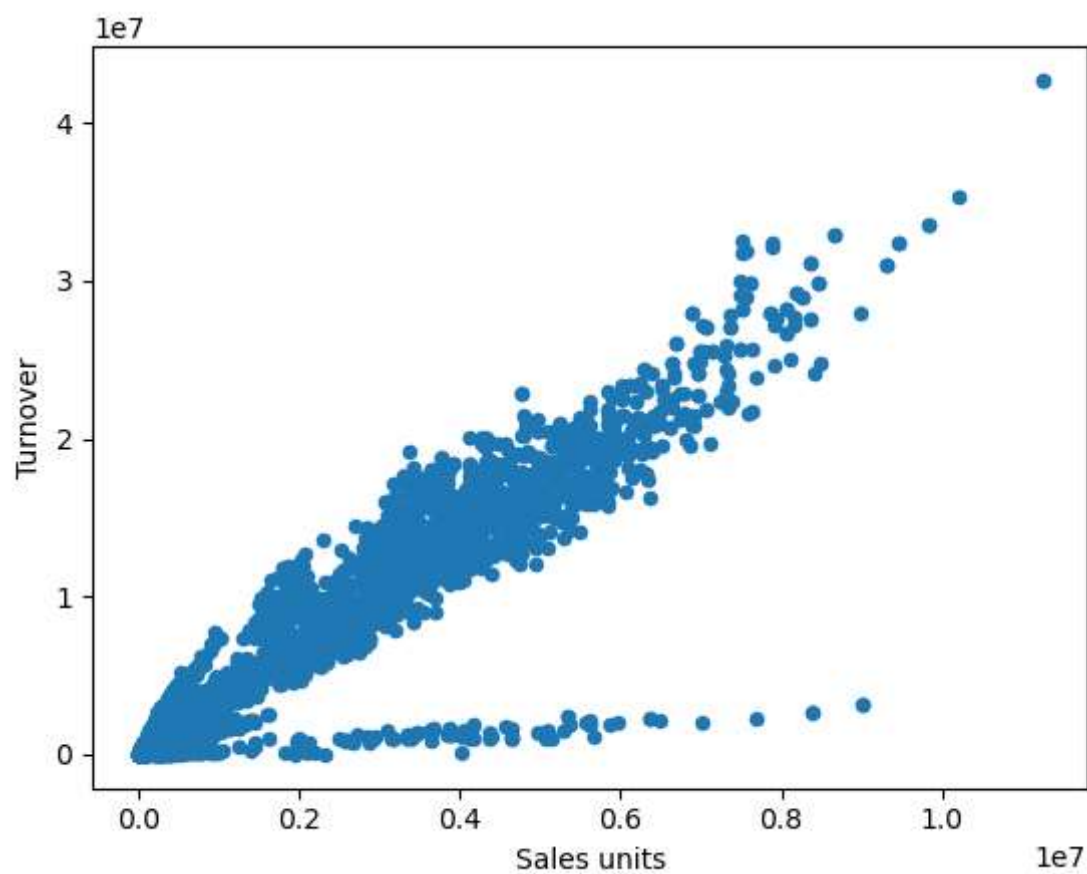
```
File ~\anaconda3\Lib\site-packages\matplotlib\backends\backend_agg.py:84, i
n RendererAgg.__init__(self, width, height, dpi)
    82 self.width = width
    83 self.height = height
--> 84 self._renderer = _RendererAgg(int(width), int(height), dpi)
    85 self._filter_renderers = []
    87 self._update_methods()
```

ValueError: Image size of 422x160423 pixels is too large. It must be less than 2^16 in each direction.

<Figure size 640x480 with 1 Axes>

```
In [67]: df4.plot.scatter(x='Sales units',y='Turnover')
```

```
Out[67]: <Axes: xlabel='Sales units', ylabel='Turnover'>
```



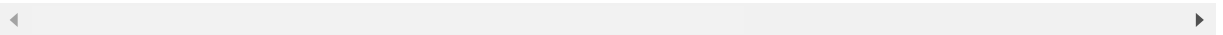
dataset-6

```
In [69]: df5=pd.read_csv(r"C:\Users\user\Downloads\7_uber.csv")
df5
```

Out[69]:

	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude
0	24238194	2015-05-07 19:52:06.0000003	7.5	2015-05-07 19:52:06 UTC	-73.999817	40.750000
1	27835199	2009-07-17 20:04:56.0000002	7.7	2009-07-17 20:04:56 UTC	-73.994355	40.750000
2	44984355	2009-08-24 21:45:00.00000061	12.9	2009-08-24 21:45:00 UTC	-74.005043	40.750000
3	25894730	2009-06-26 08:22:21.0000001	5.3	2009-06-26 08:22:21 UTC	-73.976124	40.750000
4	17610152	2014-08-28 17:47:00.000000188	16.0	2014-08-28 17:47:00 UTC	-73.925023	40.750000
...
199995	42598914	2012-10-28 10:49:00.00000053	3.0	2012-10-28 10:49:00 UTC	-73.987042	40.750000
199996	16382965	2014-03-14 01:09:00.0000008	7.5	2014-03-14 01:09:00 UTC	-73.984722	40.750000
199997	27804658	2009-06-29 00:42:00.00000078	30.9	2009-06-29 00:42:00 UTC	-73.986017	40.750000
199998	20259894	2015-05-20 14:56:25.0000004	14.5	2015-05-20 14:56:25 UTC	-73.997124	40.750000
199999	11951496	2010-05-15 04:08:00.00000076	14.1	2010-05-15 04:08:00 UTC	-73.984395	40.750000

200000 rows × 9 columns

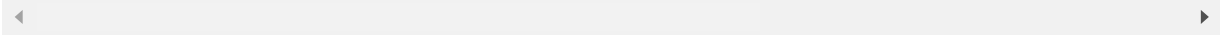



```
In [70]: df6=df5[0:10000]
df6
```

Out[70]:

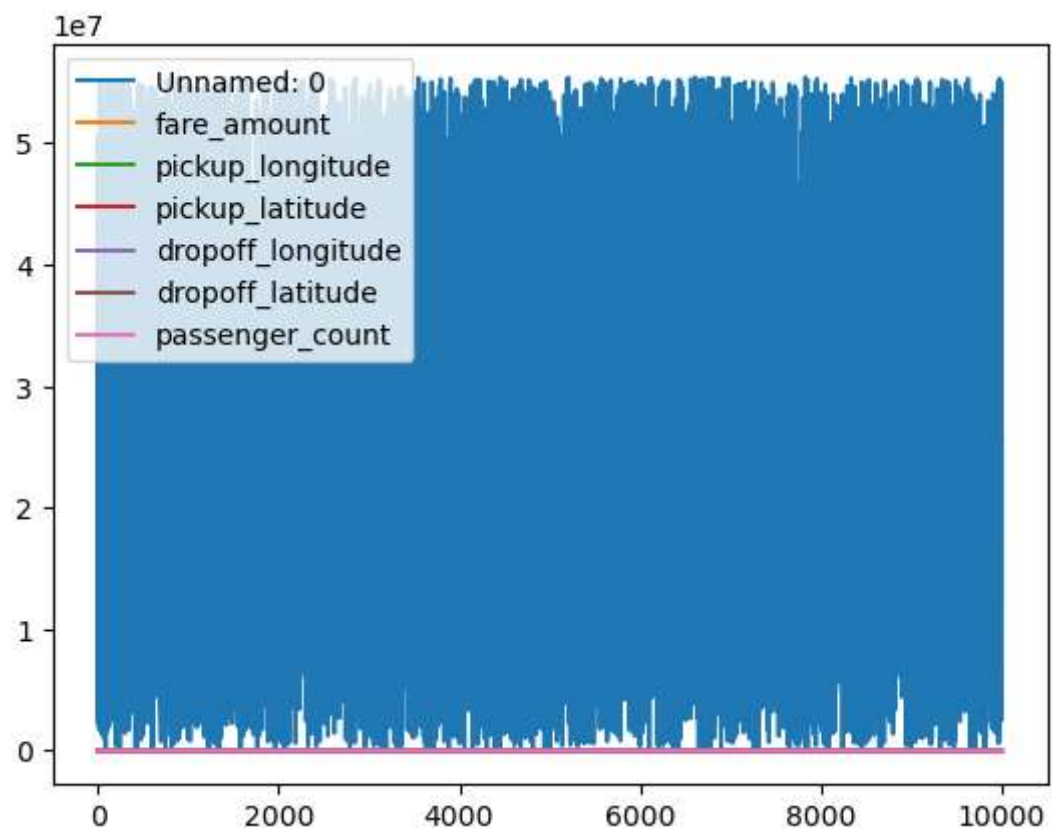
	Unnamed: 0	key	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude
0	24238194	2015-05-07 19:52:06.0000003	7.5	2015-05-07 19:52:06 UTC	-73.999817	40.731
1	27835199	2009-07-17 20:04:56.0000002	7.7	2009-07-17 20:04:56 UTC	-73.994355	40.721
2	44984355	2009-08-24 21:45:00.00000061	12.9	2009-08-24 21:45:00 UTC	-74.005043	40.741
3	25894730	2009-06-26 08:22:21.0000001	5.3	2009-06-26 08:22:21 UTC	-73.976124	40.791
4	17610152	2014-08-28 17:47:00.000000188	16.0	2014-08-28 17:47:00 UTC	-73.925023	40.741
...
9995	10890624	2011-03-13 08:32:00.00000014	3.7	2011-03-13 08:32:00 UTC	-73.987018	40.751
9996	45604084	2009-04-21 19:48:00.00000047	10.1	2009-04-21 19:48:00 UTC	-73.988603	40.691
9997	4825100	2010-01-25 10:56:00.00000089	7.3	2010-01-25 10:56:00 UTC	-73.974070	40.761
9998	6601568	2009-10-07 12:09:00.00000065	8.9	2009-10-07 12:09:00 UTC	-73.982192	40.761
9999	25777020	2014-04-18 00:16:48.0000004	4.5	2014-04-18 00:16:48 UTC	-73.978432	40.781

10000 rows × 9 columns



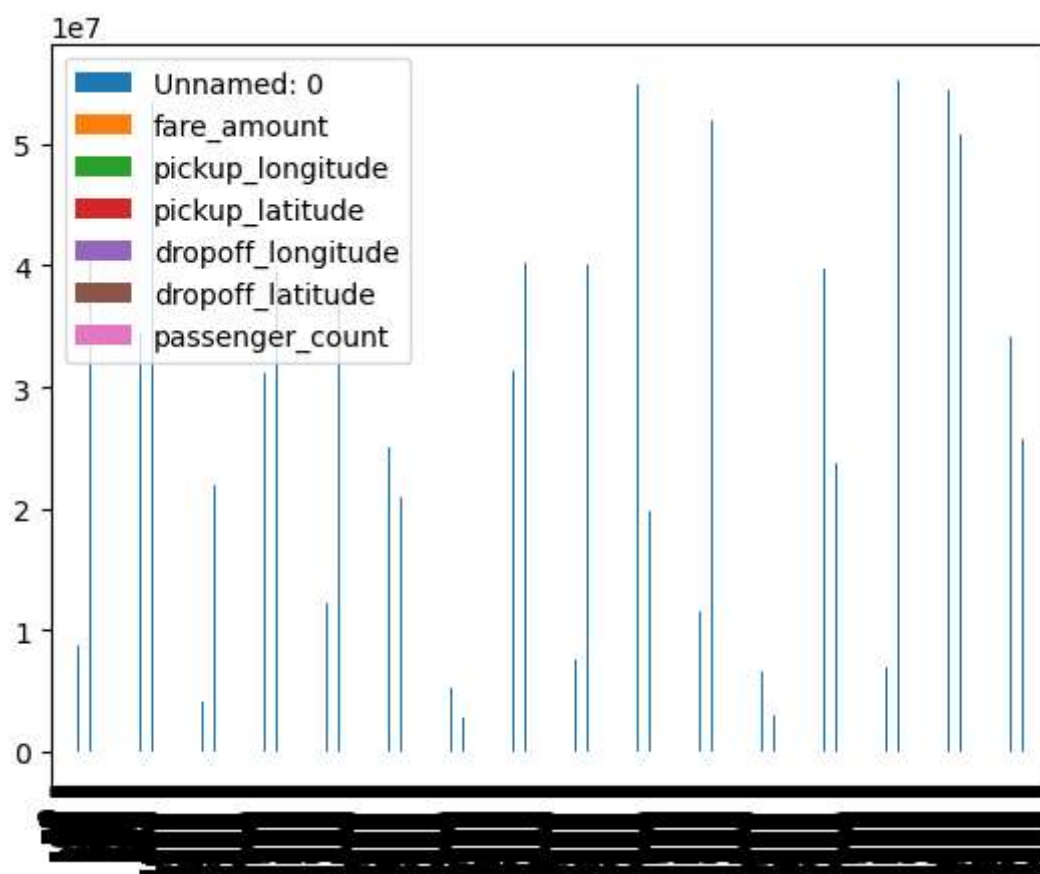
```
In [71]: df6.plot.line()
```

```
Out[71]: <Axes: >
```

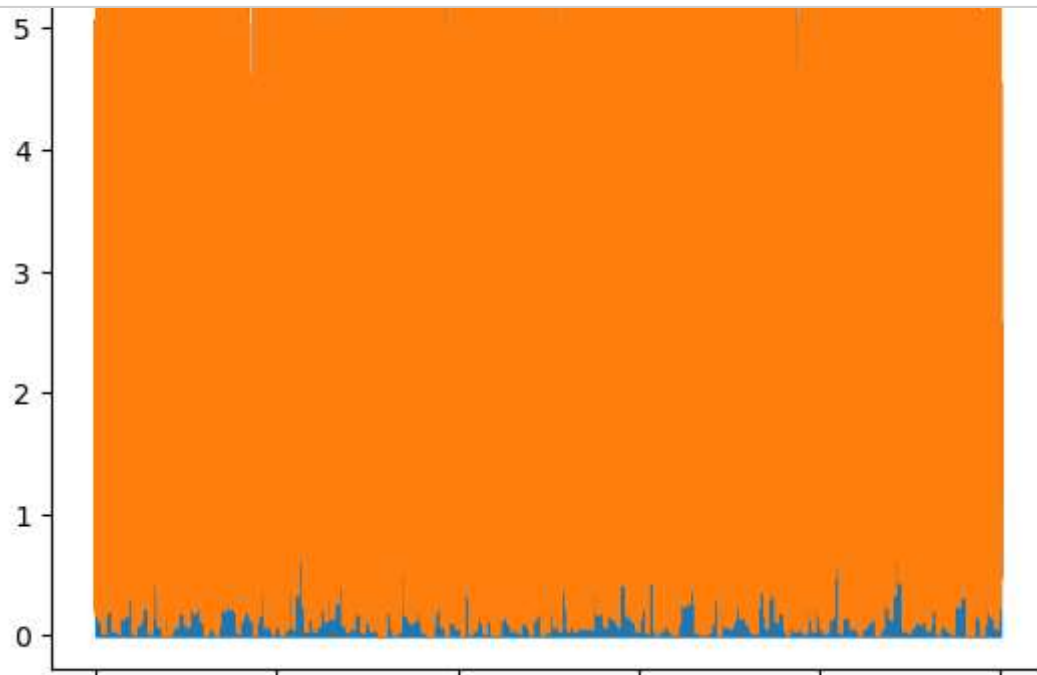


```
In [72]: df6.plot.bar()
```

```
Out[72]: <Axes: >
```

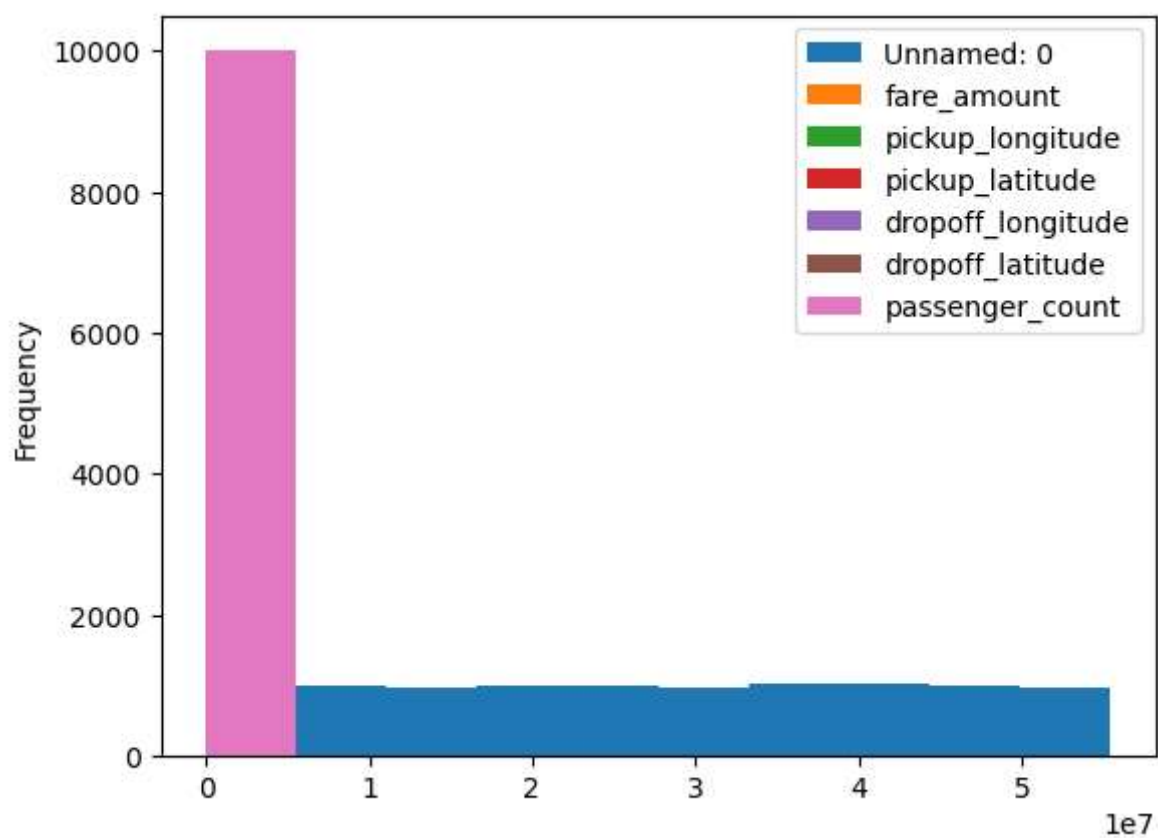


```
In [73]: df6.plot.area()
```



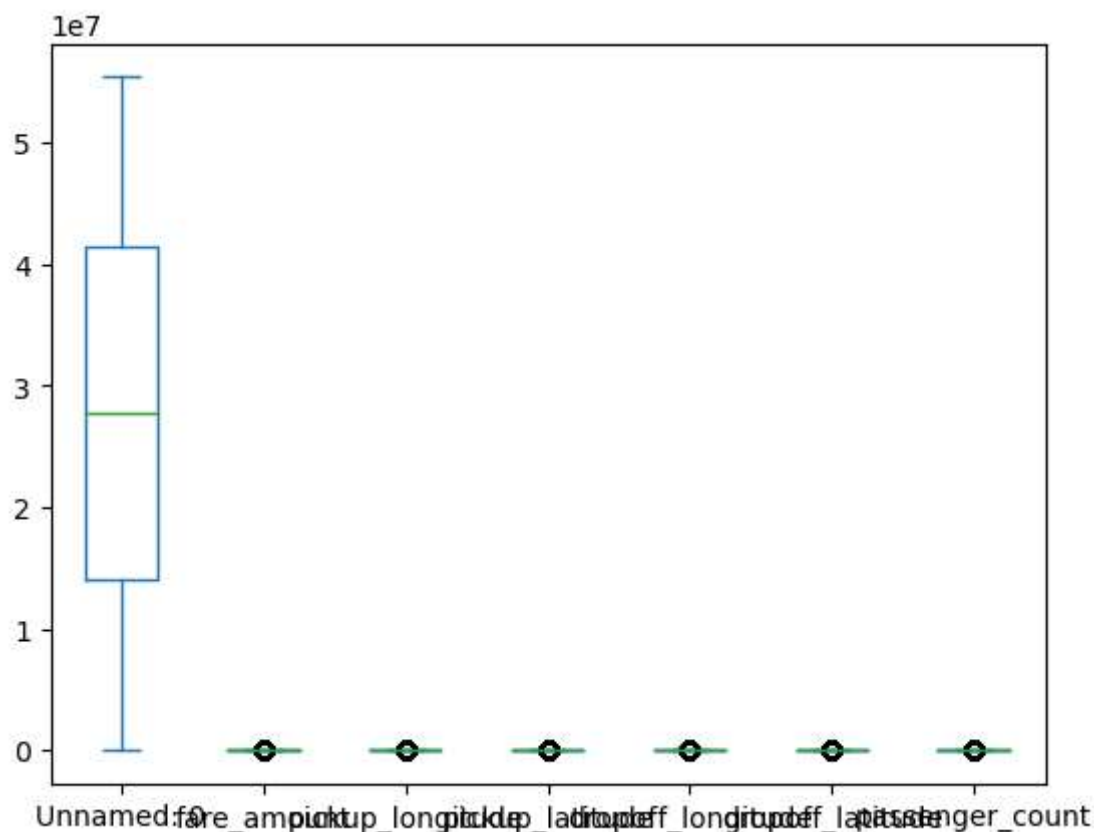
```
In [84]: df6.plot.hist()
```

```
Out[84]: <Axes: ylabel='Frequency'>
```



```
In [85]: df6.plot.box()
```

```
Out[85]: <Axes: >
```



```
In [83]: df6.plot.pie(y='passenger_count')
```

```
Out[83]: <Axes: ylabel='passenger_count'>
```

```
-----
ValueError                                Traceback (most recent call last)
File ~\anaconda3\Lib\site-packages\IPython\core\formatters.py:340, in BaseFormatter.__call__(self, obj)
    338     pass
    339 else:
--> 340     return printer(obj)
    341 # Finally look for special method names
    342 method = get_real_method(obj, self.print_method)

File ~\anaconda3\Lib\site-packages\IPython\core\pylabtools.py:152, in print_figure(fig, fmt, bbox_inches, base64, **kwargs)
    149     from matplotlib.backend_bases import FigureCanvasBase
    150     FigureCanvasBase(fig)
--> 152 fig.canvas.print_figure(bytes_io, **kw)
    153 data = bytes_io.getvalue()
    154 if fmt == 'svg':
```

```
In [82]: df6.plot.scatter(x='passenger_count',y='fare_amount')
```

```
-----  
KeyError                                Traceback (most recent call last)  
File ~\anaconda3\Lib\site-packages\pandas\core\indexes\base.py:3802, in IndexEngine.get_loc(self, key, method, tolerance)  
    3801 try:  
-> 3802     return self._engine.get_loc(casted_key)  
    3803 except KeyError as err:  
  
File ~\anaconda3\Lib\site-packages\pandas\_libs\index.py:138, in pandas._libs.index.IndexEngine.get_loc()  
  
File ~\anaconda3\Lib\site-packages\pandas\_libs\index.py:165, in pandas._libs.index.IndexEngine.get_loc()  
  
File pandas\_libs\hashtable_class_helper.pxi:5745, in pandas._libs.hashtable.PyObjectHashTable.get_item()  
  
File pandas\_libs\hashtable_class_helper.pxi:5753, in pandas._libs.hashtable.PyObjectHashTable.get_item()
```

```
In [ ]:
```