

# 1.Importing libraries

In [1]:

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

# 2. importing dataset

In [2]:

```
data=pd.read_csv(r"C:\Users\user\Downloads\VE.CSV.csv")
data
```

Out[2]:

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom
0	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143	0.66557
1	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784	0.62877
2	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464	0.64938
3	Norway	Western Europe	4	7.522	0.03880	1.45900	1.33095	0.88521	0.66973
4	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563	0.63297
...	...	...	...	...	...	...	...	...	...
153	Rwanda	Sub-Saharan Africa	154	3.465	0.03464	0.22208	0.77370	0.42864	0.59201
154	Benin	Sub-Saharan Africa	155	3.340	0.03656	0.28665	0.35386	0.31910	0.48450
155	Syria	Middle East and Northern Africa	156	3.006	0.05015	0.66320	0.47489	0.72193	0.15684
156	Burundi	Sub-Saharan Africa	157	2.905	0.08658	0.01530	0.41587	0.22396	0.11850
157	Togo	Sub-Saharan Africa	158	2.839	0.06727	0.20868	0.13995	0.28443	0.36453

158 rows × 12 columns

## 3.head()

In [3]:

```
data.head(7)
```

Out[3]:

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	(GDP per Capita)
0	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143	0.66557	
1	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784	0.62877	
2	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464	0.64938	
3	Norway	Western Europe	4	7.522	0.03880	1.45900	1.33095	0.88521	0.66973	
4	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563	0.63297	
5	Finland	Western Europe	6	7.406	0.03140	1.29025	1.31826	0.88911	0.64169	
6	Netherlands	Western Europe	7	7.378	0.02799	1.32944	1.28017	0.89284	0.61576	

## 4.tail()

In [4]:

```
data.tail(8)
```

Out[4]:

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom
150	Ivory Coast	Sub-Saharan Africa	151	3.655	0.05141	0.46534	0.77115	0.15185	0.46866
151	Burkina Faso	Sub-Saharan Africa	152	3.587	0.04324	0.25812	0.85188	0.27125	0.39493
152	Afghanistan	Southern Asia	153	3.575	0.03084	0.31982	0.30285	0.30335	0.23414
153	Rwanda	Sub-Saharan Africa	154	3.465	0.03464	0.22208	0.77370	0.42864	0.59201
154	Benin	Sub-Saharan Africa	155	3.340	0.03656	0.28665	0.35386	0.31910	0.48450

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom
155	Syria	Middle East and Northern Africa	156	3.006	0.05015	0.66320	0.47489	0.72193	0.15684
156	Burundi	Sub-Saharan Africa	157	2.905	0.08658	0.01530	0.41587	0.22396	0.11850
157	Togo	Sub-Saharan Africa	158	2.839	0.06727	0.20868	0.13995	0.28443	0.36453

## 5.describe()

In [5]:

```
data.describe()
```

Out[5]:

	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	True (Governmer Corruption)
count	158.000000	158.000000	158.000000	158.000000	158.000000	158.000000	158.000000	158.000000
mean	79.493671	5.375734	0.047885	0.846137	0.991046	0.630259	0.428615	0.14342
std	45.754363	1.145010	0.017146	0.403121	0.272369	0.247078	0.150693	0.12003
min	1.000000	2.839000	0.018480	0.000000	0.000000	0.000000	0.000000	0.00000
25%	40.250000	4.526000	0.037268	0.545808	0.856823	0.439185	0.328330	0.06167
50%	79.500000	5.232500	0.043940	0.910245	1.029510	0.696705	0.435515	0.10722
75%	118.750000	6.243750	0.052300	1.158448	1.214405	0.811013	0.549092	0.18025
max	158.000000	7.587000	0.136930	1.690420	1.402230	1.025250	0.669730	0.55191

◀ ▶

## 6.shape

In [6]:

```
np.shape(data)
```

Out[6]: (158, 12)

## 7.size

In [7]:

```
np.size(data)
```

Out[7]: 1896

## 8.isna()

In [8]:

data.isna()

Out[8]:

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom	(Governance) Cor
0	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False
...	...	...	...	...	...	...	...	...	...	...
153	False	False	False	False	False	False	False	False	False	False
154	False	False	False	False	False	False	False	False	False	False
155	False	False	False	False	False	False	False	False	False	False
156	False	False	False	False	False	False	False	False	False	False
157	False	False	False	False	False	False	False	False	False	False

158 rows × 12 columns



## 9.dropna()

In [9]:

data.dropna()

Out[9]:

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom
0	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143	0.66557
1	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784	0.62877
2	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464	0.64938
3	Norway	Western Europe	4	7.522	0.03880	1.45900	1.33095	0.88521	0.66973
4	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563	0.63297

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom
...	...	...	...	...	...	...	...	...	...
153	Rwanda	Sub-Saharan Africa	154	3.465	0.03464	0.22208	0.77370	0.42864	0.59201
154	Benin	Sub-Saharan Africa	155	3.340	0.03656	0.28665	0.35386	0.31910	0.48450
155	Syria	Middle East and Northern Africa	156	3.006	0.05015	0.66320	0.47489	0.72193	0.15684
156	Burundi	Sub-Saharan Africa	157	2.905	0.08658	0.01530	0.41587	0.22396	0.11850
157	Togo	Sub-Saharan Africa	158	2.839	0.06727	0.20868	0.13995	0.28443	0.36453

158 rows × 12 columns

## 10. selecting specific columns

In [10]:

```
da=data[["Happiness Rank","Family"]]
da
```

Out[10]:

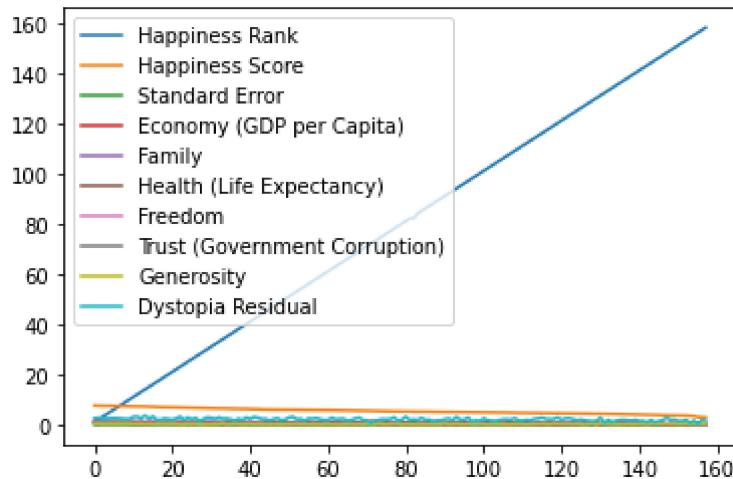
	Happiness Rank	Family
0	1	1.34951
1	2	1.40223
2	3	1.36058
3	4	1.33095
4	5	1.32261
...	...	...
153	154	0.77370
154	155	0.35386
155	156	0.47489
156	157	0.41587
157	158	0.13995

158 rows × 2 columns

## 11.line plot

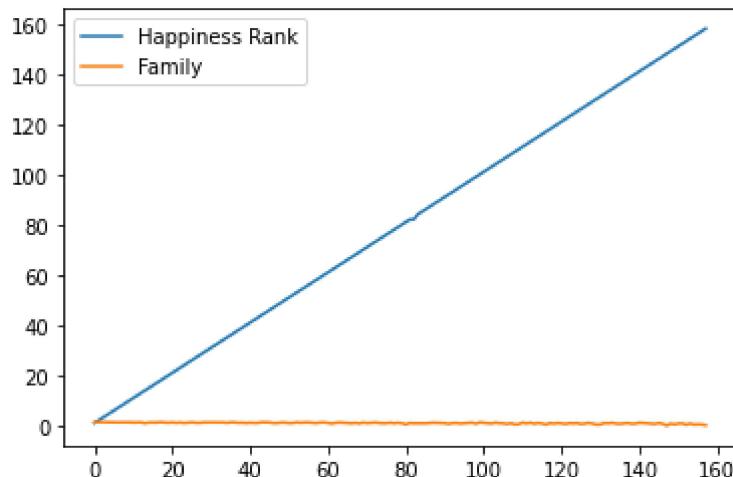
```
In [11]: data.plot.line()
```

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



```
In [12]: da.plot.line()
```

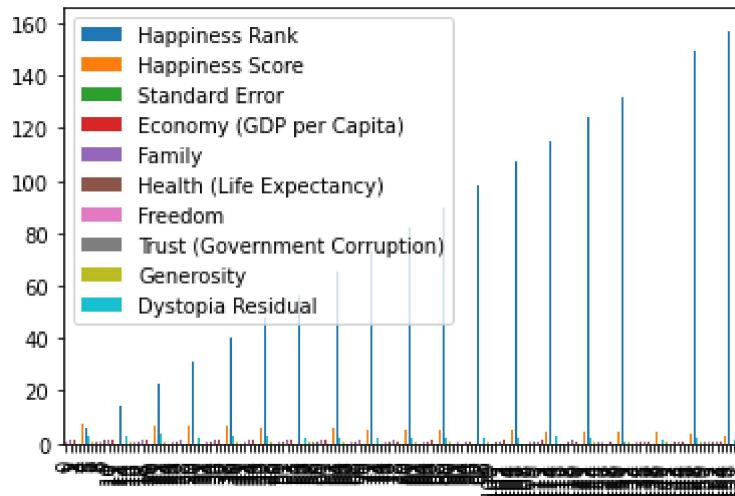
```
Out[12]: <AxesSubplot:>
```



## 12.bar plot

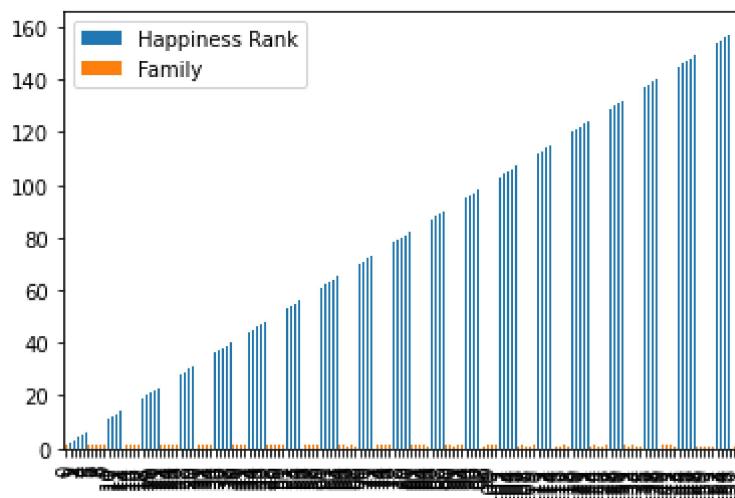
```
In [13]: data.plot.bar()
```

```
Out[13]: <AxesSubplot:>
```



In [14]: `da.plot.bar()`

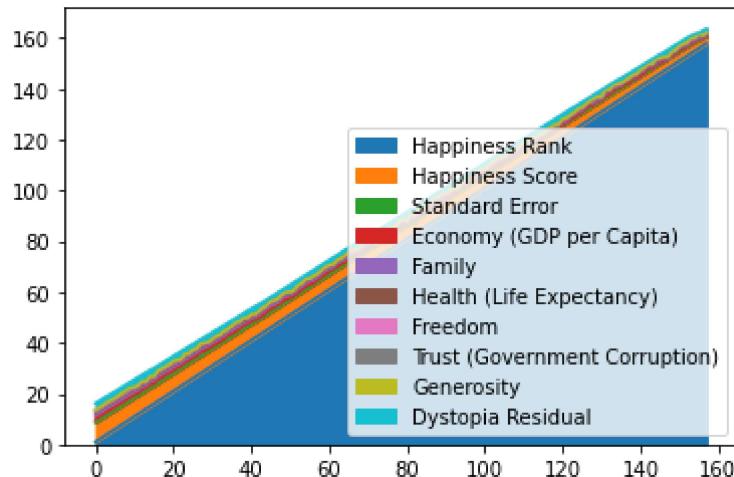
Out[14]: <AxesSubplot:>



## 13.area plot

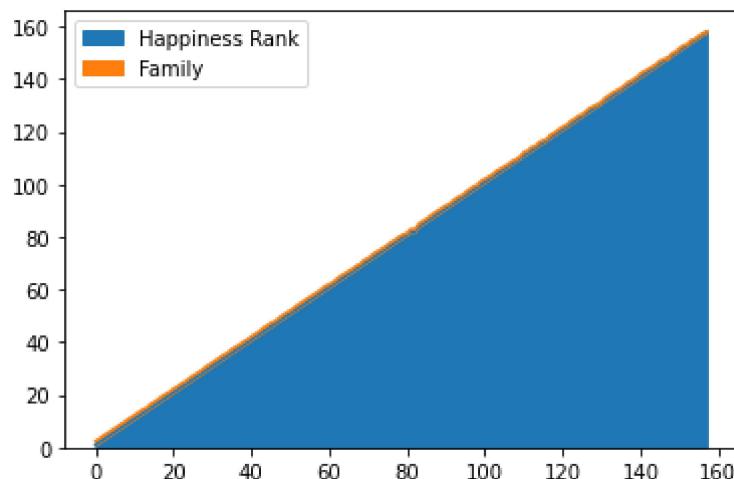
In [15]: `data.plot.area()`

Out[15]: <AxesSubplot:>



In [16]: `da.plot.area()`

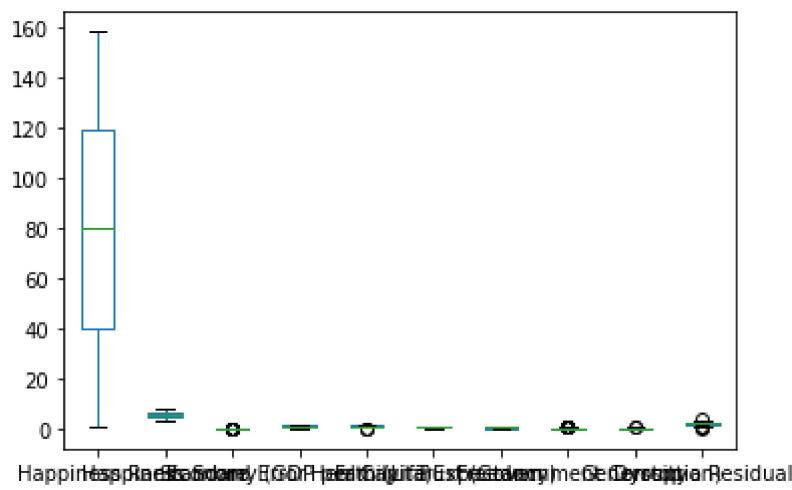
Out[16]: <AxesSubplot:>



## 14. BOX plot

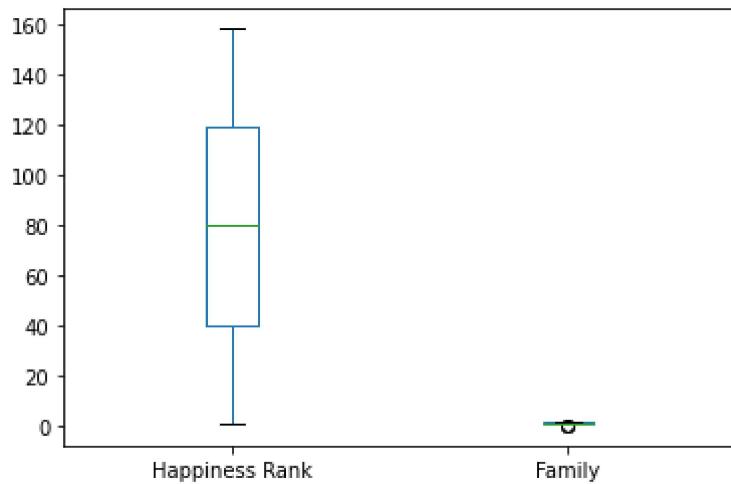
In [17]: `data.plot.box()`

Out[17]: <AxesSubplot:>



```
In [18]: da.plot.box()
```

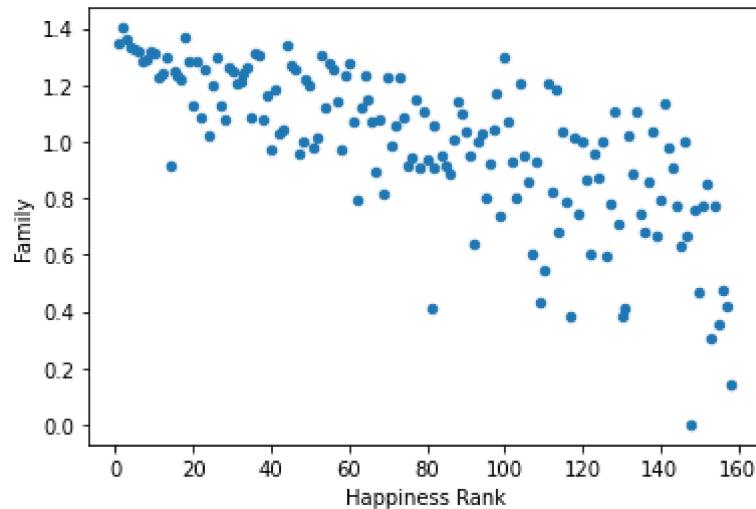
```
Out[18]: <AxesSubplot:
```



## 15.scatter plot

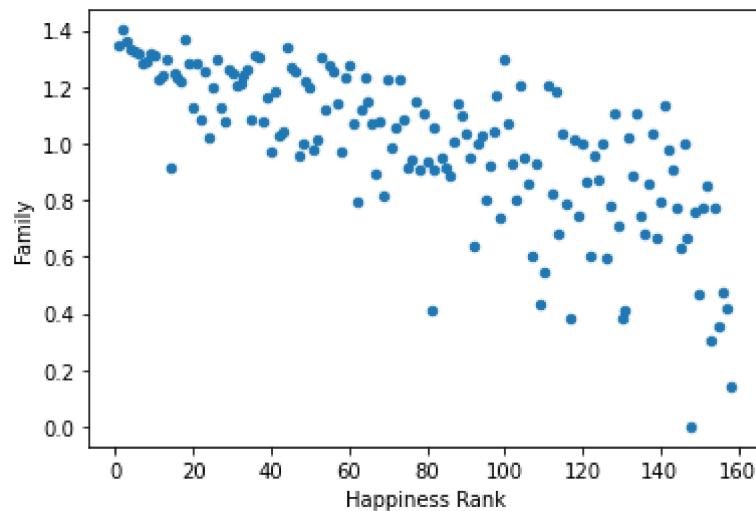
```
In [20]: data.plot.scatter(x="Happiness Rank", y="Family")
```

```
Out[20]: <AxesSubplot:xlabel='Happiness Rank', ylabel='Family'>
```



```
In [21]: da.plot.scatter(x="Happiness Rank",y="Family")
```

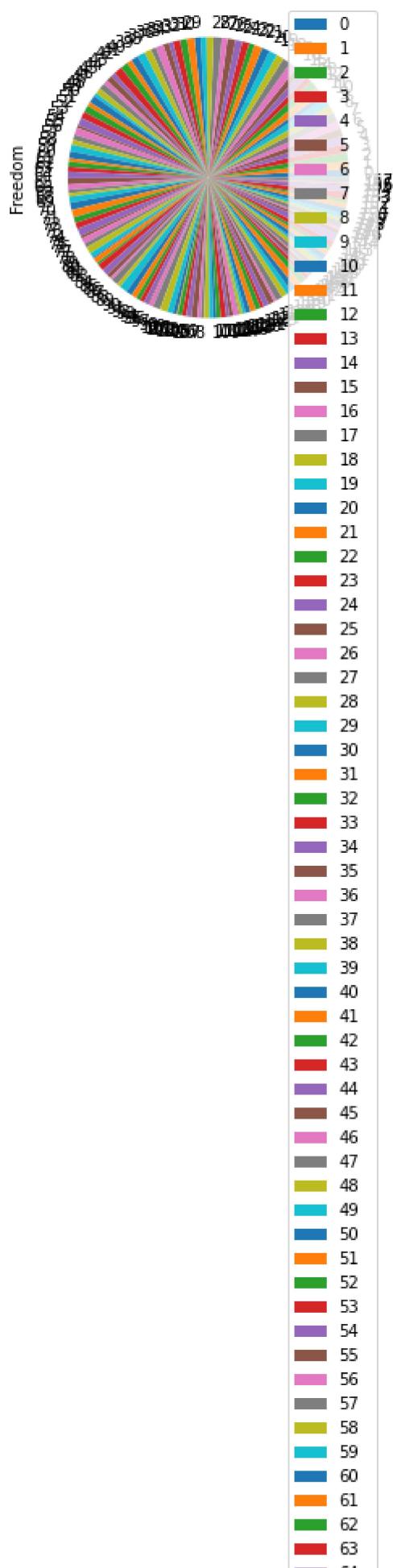
```
Out[21]: <AxesSubplot:xlabel='Happiness Rank', ylabel='Family'>
```

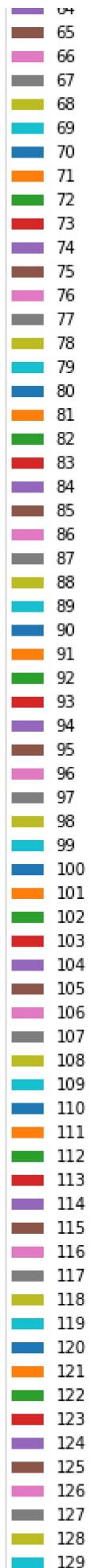


## 16.pie chart

```
In [22]: data.plot.pie(x="Standard Error",y="Freedom")
```

```
Out[22]: <AxesSubplot:ylabel='Freedom'>
```

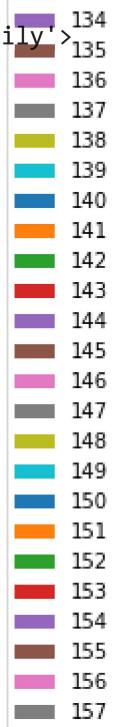


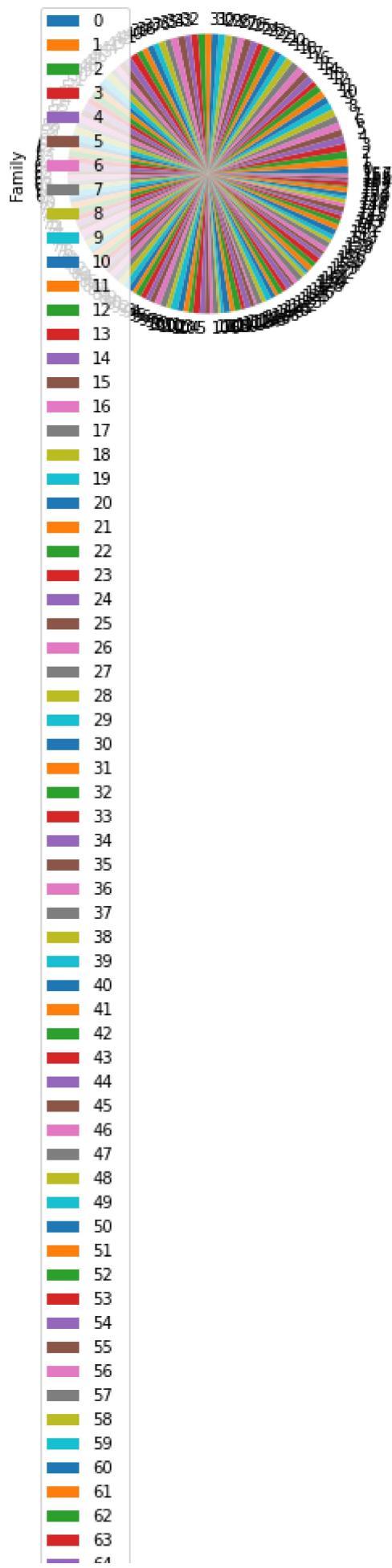


130

```
In [23]: da.plot.pie(x="Happiness Rank",y="Family")
```

```
Out[23]: <AxesSubplot:ylabel='Family'>
```



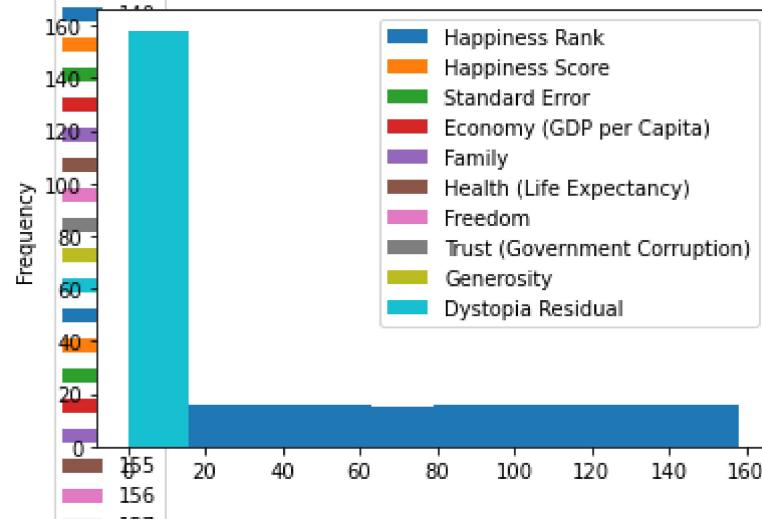


64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129

# 17.histogram

```
In [24]: data.plot.hist()
```

```
Out[24]: <AxesSubplot:ylabel='Frequency'>
```



```
In [ ]:
```