

```
In [2]: import pandas as pd
data=pd.read_csv('/Users/Rana/Desktop/SpeciesSearchResults.csv')
regions=pd.read_csv('/Users/Rana/Desktop/species_regions.csv')
```

```
In [3]: data.describe()
```

```
Out[3]:
```

	Unnamed: 0	Unnamed: 0.1
count	9596.000000	9596.000000
mean	4797.500000	4797.500000
std	2770.270925	2770.270925
min	0.000000	0.000000
25%	2398.750000	2398.750000
50%	4797.500000	4797.500000
75%	7196.250000	7196.250000
max	9595.000000	9595.000000

```
In [ ]:
```

```
In [4]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9596 entries, 0 to 9595
Data columns (total 17 columns):
#   Column                                     Non-Null Count  Dtype
---  ---
0   Unnamed: 0                                9596 non-null   int64
1   Unnamed: 0.1                              9596 non-null   int64
2   Scientific Name                           9596 non-null   object
3   Common Name                               9596 non-null   object
4   Inverted Common Name                      9596 non-null   object
5   Species Code                              9596 non-null   object
6   Population Code                           9596 non-null   object
7   CurrentDistribution                        8356 non-null   object
8   Family                                    9593 non-null   object
9   First Listed                              2593 non-null   object
10  Species Group                             9596 non-null   object
11  Lead Region                               9526 non-null   object
12  Federal Listing Status                    9596 non-null   object
13  Regions of Occurrence                     7608 non-null   object
14  U.S. or Foreign Listed                    9525 non-null   object
15  Vertebrate/Invertebrate/Plant           9596 non-null   object
16  Where Listed                              9569 non-null   object
dtypes: int64(2), object(15)
memory usage: 1.2+ MB
```

```
In [5]: data.head(10)
```

Out [5]:

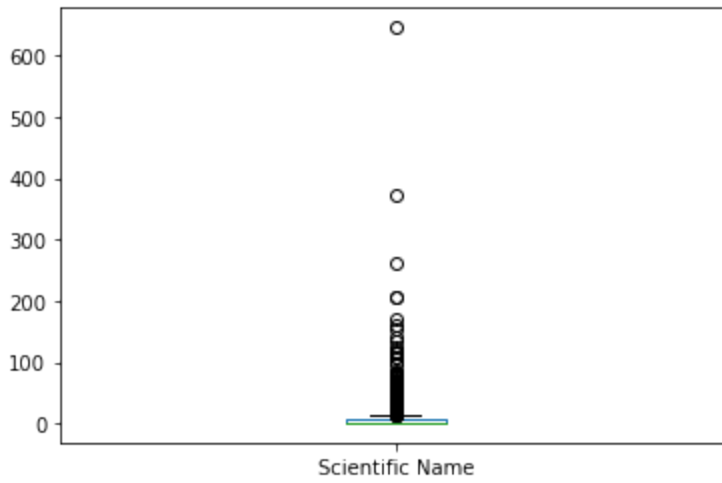
	Unnamed: 0	Unnamed: 0.1	Scientific Name	Common Name	Inverted Common Name	Species Code	Population Code	CurrentDistri
0	0	0	Abies fraseri	Fraser fir	Fir, Fraser	R00B	P01	
1	1	1	Abies guatemalensis	Guatemalan Fir (=pinabete)	Fir, Guatemalan (=pinabete)	R003	P01	Mexico, Hc Guate S
2	2	2	Ablautus schlingeri	Oso Flaco robber fly	Robber fly, Oso Flaco	I003	I01	
3	3	3	Abronia alpina	Ramshaw Meadows sand- verbena	Sand- verbena, Ramshaw Meadows	Q009	P01	
4	4	4	Abronia ammophila	Yellowstone Sand Verbena	Verbena Yellowstone Sand	Q3LO	P01	
5	5	5	Abronia ammophila var.	No common name	No common name	Q3LN	P01	
6	6	6	Abronia ammophila	[Unnamed] sand- verbena	Sand- verbena, [unnamed]	Q3AM	P01	
7	7	7	Abronia bigelovii	[Unnamed] sand- verbena	Sand- verbena, [unnamed]	Q00A	P01	
8	8	8	Abronia macrocarpa	Large- fruited sand- verbena	Sand- verbena, large- fruited	Q24E	P01	TX; U.S
9	9	9	Abronia turbinata	[Unnamed] sand- verbena	Sand- verbena, [unnamed]	Q24F	P01	

```
In [6]: data['First Listed']=pd.to_datetime(data['First Listed'])
```

```
In [7]: #data['First Listed'].unique()
```

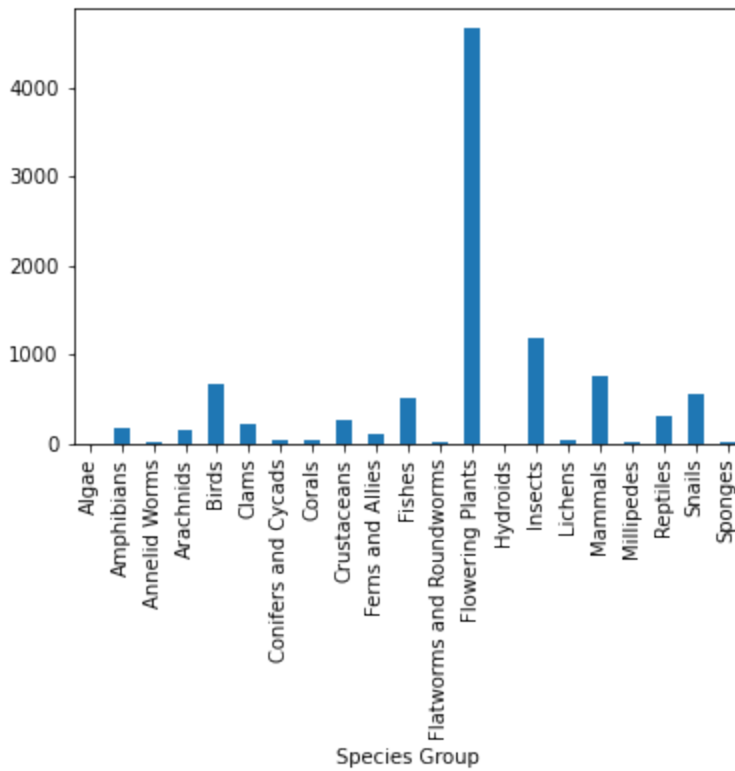
```
In [8]: data['Scientific Name'].groupby(data['Family']).count().plot(kind='box')
```

```
Out [8]: <AxesSubplot:>
```



```
In [9]: data['Scientific Name'].groupby(data['Species Group']).count().plot(kind='bar')
```

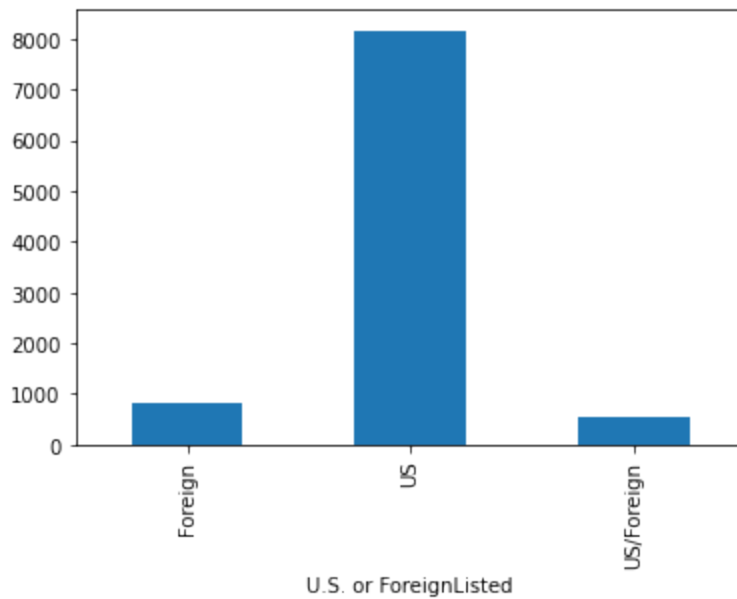
Out[9]: <AxesSubplot:xlabel='Species Group'>



In []:

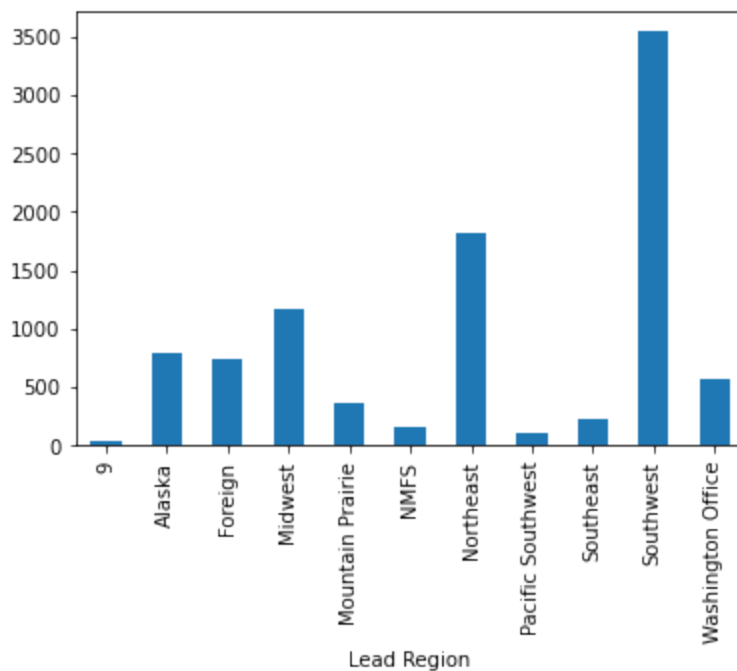
```
In [10]: data['Scientific Name'].groupby(data['U.S. or ForeignListed']).count().plot(kind='bar')
```

Out[10]: <AxesSubplot:xlabel='U.S. or ForeignListed'>



```
In [11]: data['Scientific Name'].groupby(data['Lead Region']).count().plot(kind='bar')
```

```
Out[11]: <AxesSubplot:xlabel='Lead Region'>
```



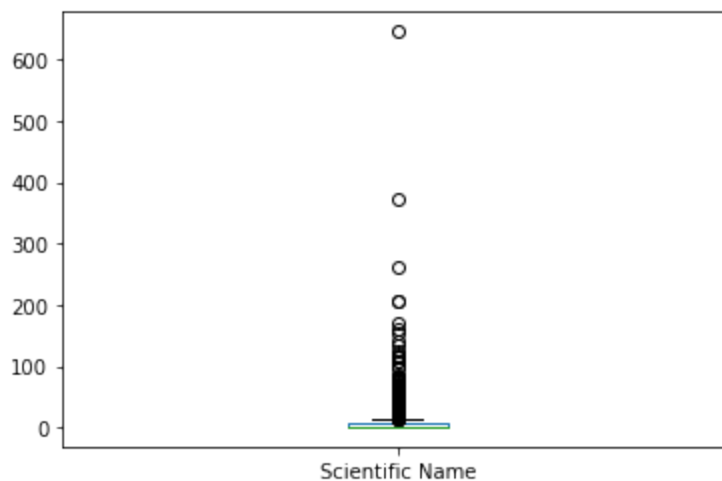
```
In [12]: data['Federal Listing Status'].unique()
```

```
Out[12]: array(['Not Listed', 'Threatened', 'Endangered',
        'Under Review in the Candidate or Petition Process',
        'Proposed Endangered', 'Recovery', 'Proposed Threatened',
        'Original Data in Error - Not a listable entity', 'Extinction',
        'Similarity of Appearance to a Threatened Taxon',
        'Original Data in Error - Taxonomic Revision',
        'Experimental Population, Non-Essential',
        'Pre-Act Delisting (or clearance--removal from the Lists)',
        'Original Data in Error - New Information Discovered', 'Candidate',
        'Original Data in Error - Act Amendment',
        'Original Data in Error - Erroneous Data',
```

```
'Proposed Similarity of Appearance to a Threatened Taxon'],
dtype=object)
```

```
In [13]: data['Scientific Name'].groupby(data['Family']).count().plot(kind='box')
```

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



```
In [14]: regions
```

```
Out[14]:
```

	Number	Lead Region
0	0	Pacific
1	1	Southwest
2	2	Midwest
3	3	Southeast
4	4	Northeast
5	5	Mountain Prairie
6	6	Alaska
7	7	Pacific Southwest
8	8	Washington Office

```
In [15]: regions.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9 entries, 0 to 8
Data columns (total 2 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   Number          9 non-null     int64
1   Lead Region     9 non-null     object
dtypes: int64(1), object(1)
memory usage: 272.0+ bytes
```

```
In [16]: regions
```

```
Out [16]:
```

	Number	Lead Region
0	0	Pacific
1	1	Southwest
2	2	Midwest
3	3	Southeast
4	4	Northeast
5	5	Mountain Prairie
6	6	Alaska
7	7	Pacific Southwest
8	8	Washington Office

```
import pandas as pd
regions=pd.read_csv('/Users/Rana/Desktop/Lead_Region.csv', usecols=[0]) .squeeze()
regions.str.split(',')
regions=regions.str.replace('Region','',regex=True)
regions=regions.str.replace('','',regex=True)
regions=regions.str.replace('','',regex=True)
#regions.to_csv('species_regions')
print(regions)
```

```
In [22]: type(regions.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9 entries, 0 to 8
Data columns (total 2 columns):
#   Column          Non-Null Count  Dtype
---  ---
0   Number          9 non-null      int64
1   Lead Region     9 non-null      object
dtypes: int64(1), object(1)
memory usage: 272.0+ bytes
```

```
Out[22]: NoneType
```

```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```