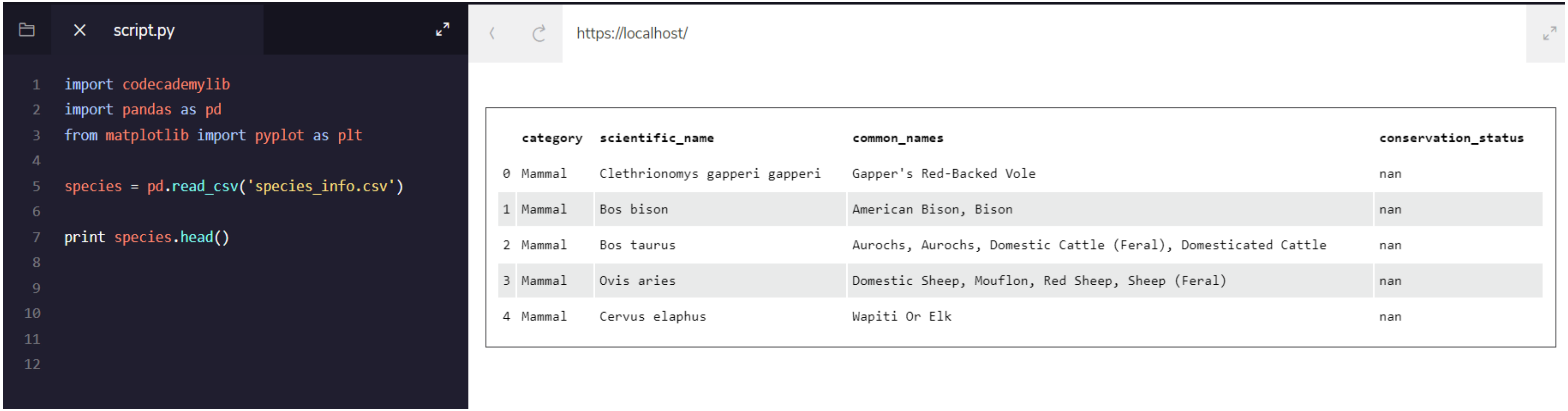
Biodiveristy Capstone Project
Investigating Protected Species

Ruolan Ji

1. Biodiversity Project

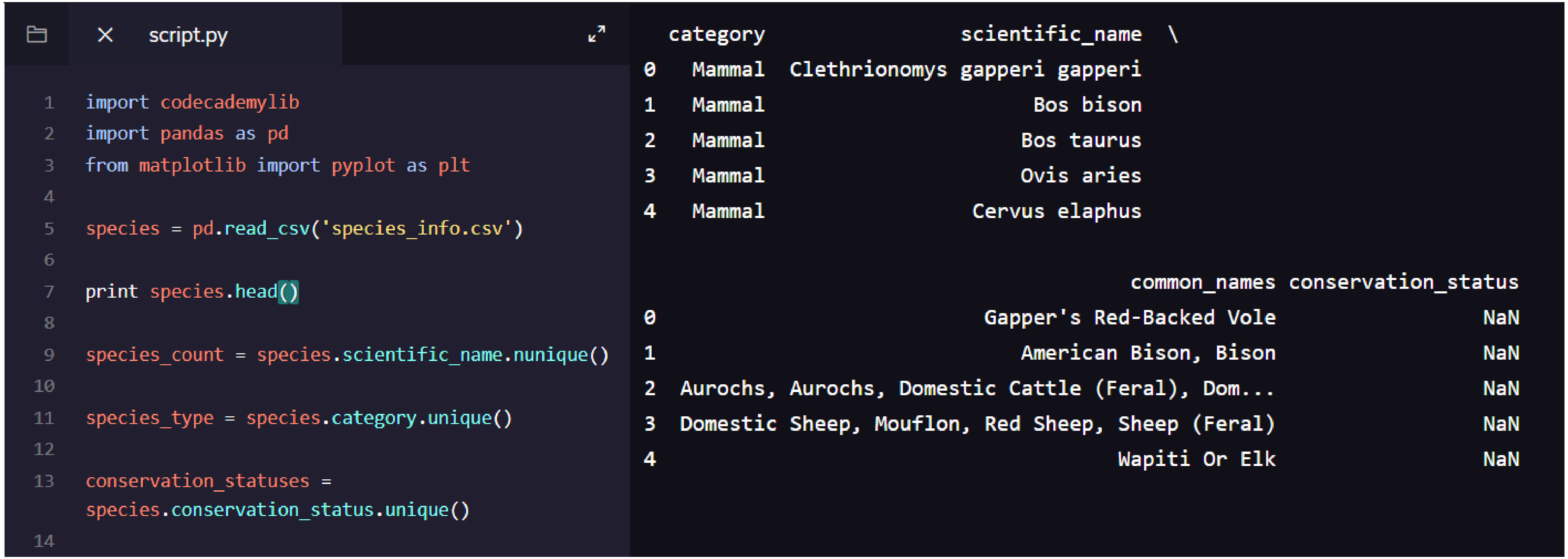


```python
import codecademylib
import pandas as pd
from matplotlib import pyplot as plt

species = pd.read_csv('species_info.csv')

print species.head()
```

	category	scientific_name	common_names	conservation_status
0	Mammal	Clethrionomys gapperi gapperi	Gapper's Red-Backed Vole	nan
1	Mammal	Bos bison	American Bison, Bison	nan
2	Mammal	Bos taurus	Aurochs, Aurochs, Domestic Cattle (Feral), Domesticated Cattle	nan
3	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep, Sheep (Feral)	nan
4	Mammal	Cervus elaphus	Wapiti Or Elk	nan

- Simply loading the file to visualize the raw data

2. Inspected the DataFrame

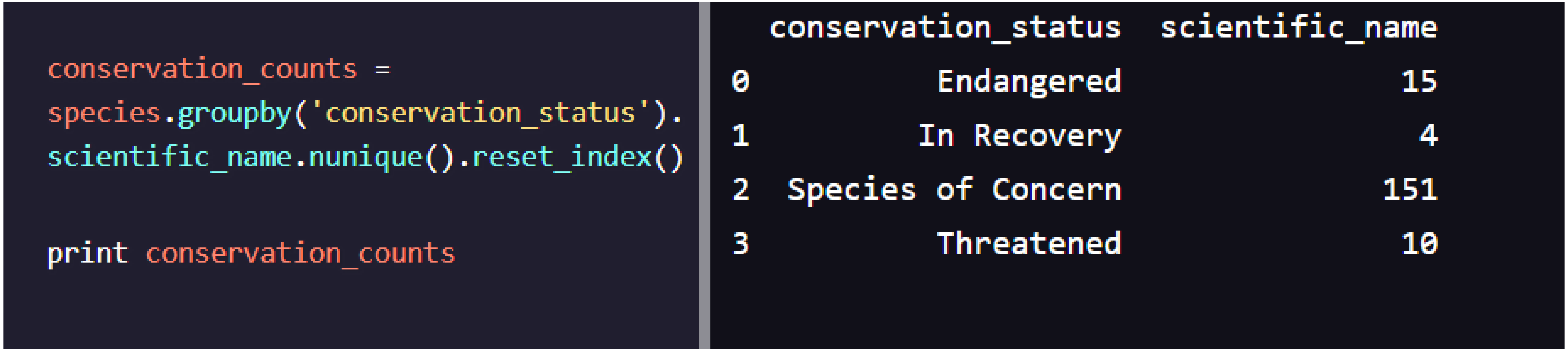


```python
import codecademylib
import pandas as pd
from matplotlib import pyplot as plt

species = pd.read_csv('species_info.csv')

print species.head()

species_count = species.scientific_name.nunique()

species_type = species.category.unique()

conservation_statuses = species.conservation_status.unique()

```

```
  category                scientific_name  \
0   Mammal  Clethrionomys gapperi gapperi
1   Mammal                      Bos bison
2   Mammal                     Bos taurus
3   Mammal                     Ovis aries
4   Mammal                 Cervus elaphus

                                common_names conservation_status
0                   Gapper's Red-Backed Vole                 NaN
1                      American Bison, Bison                 NaN
2  Aurochs, Aurochs, Domestic Cattle (Feral), Dom...         NaN
3  Domestic Sheep, Mouflon, Red Sheep, Sheep (Feral)         NaN
4                              Wapiti Or Elk                 NaN
```

- how many species, how many different values of category and of conservation status.

3. Analyze Species Conservation Status

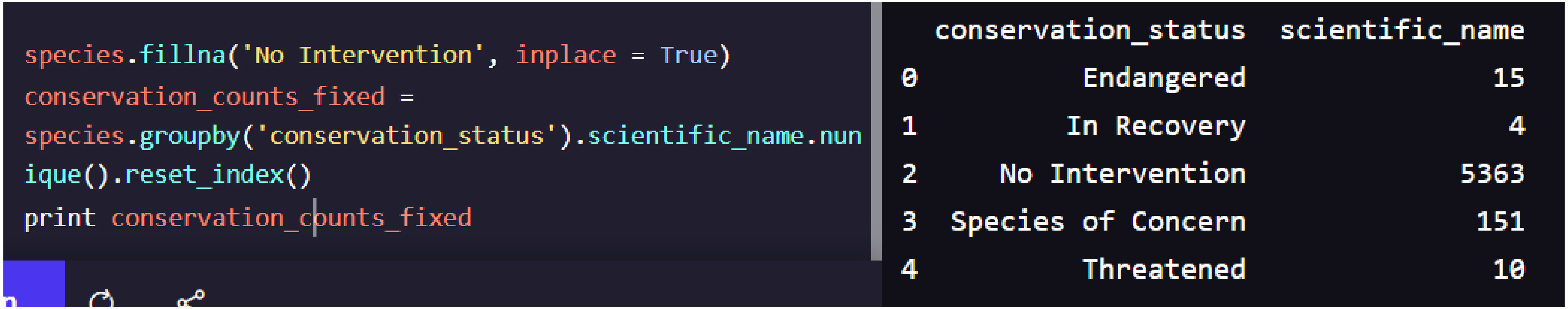


```
conservation_counts =
species.groupby('conservation_status').
scientific_name.nunique().reset_index()

print conservation_counts
```

```
  conservation_status  scientific_name
0          Endangered               15
1         In Recovery                4
2  Species of Concern              151
3          Threatened               10
```

- how many of each species fall into these conservation statuses

4. Analyze Conservation Status II

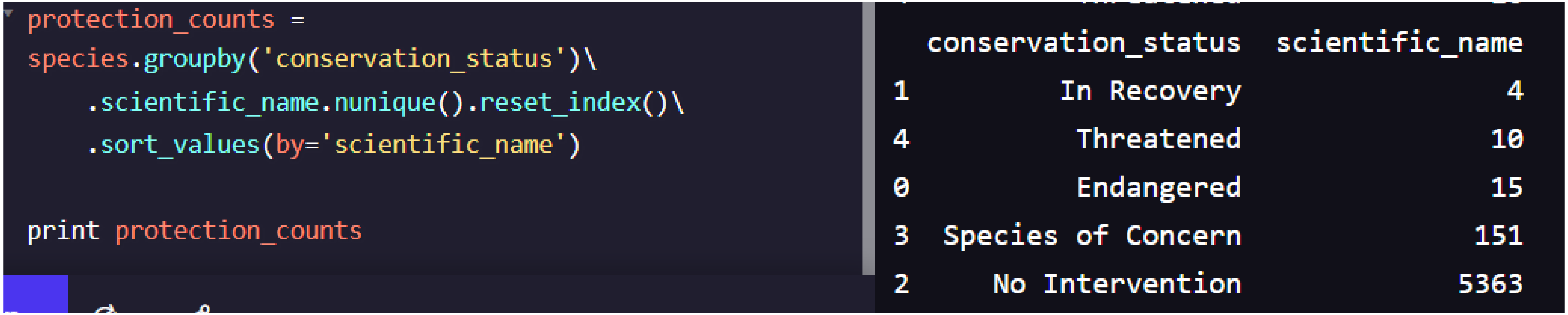


```
species.fillna('No Intervention', inplace = True)
conservation_counts_fixed =
species.groupby('conservation_status').scientific_name.nun
ique().reset_index()
print conservation_counts_fixed
```

```
    conservation_status  scientific_name
0            Endangered               15
1           In Recovery                4
2       No Intervention             5363
3    Species of Concern              151
4            Threatened               10
```

- Clean up the data to be more organized

5. Plotting Conservation Status by Species

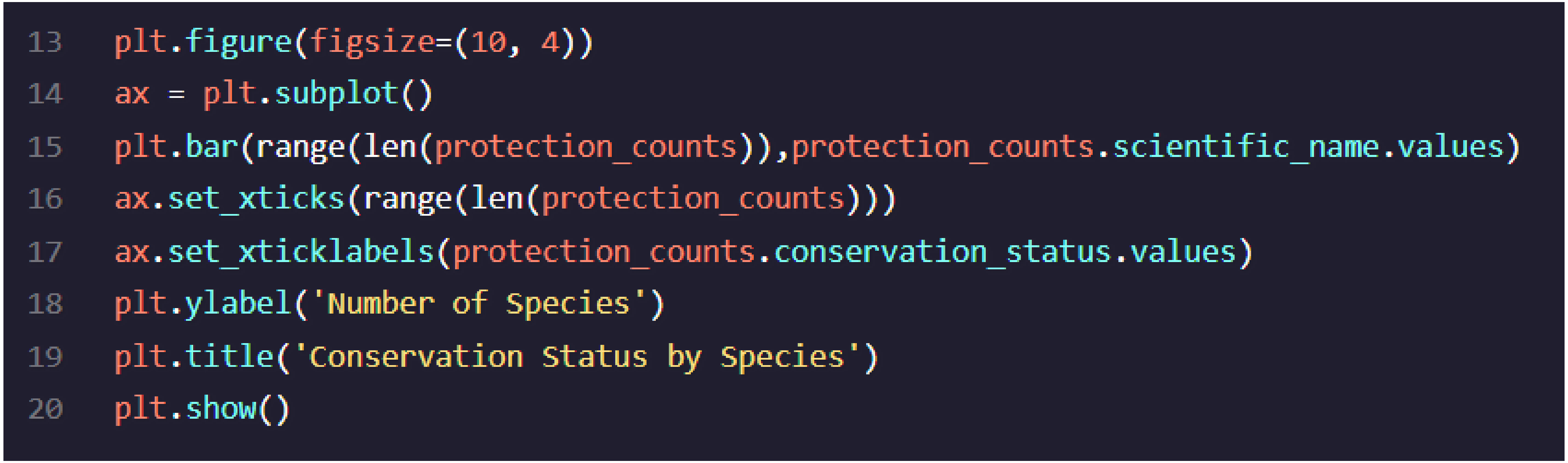


```
protection_counts = 
species.groupby('conservation_status')\
    .scientific_name.nunique().reset_index()\
    .sort_values(by='scientific_name')

print protection_counts
```

```
  conservation_status  scientific_name
1         In Recovery                4
4          Threatened               10
0          Endangered               15
3  Species of Concern              151
2     No Intervention             5363
```

- conservation_counts_fixed indexed version protection_counts
- Visualize protection_counts to plot

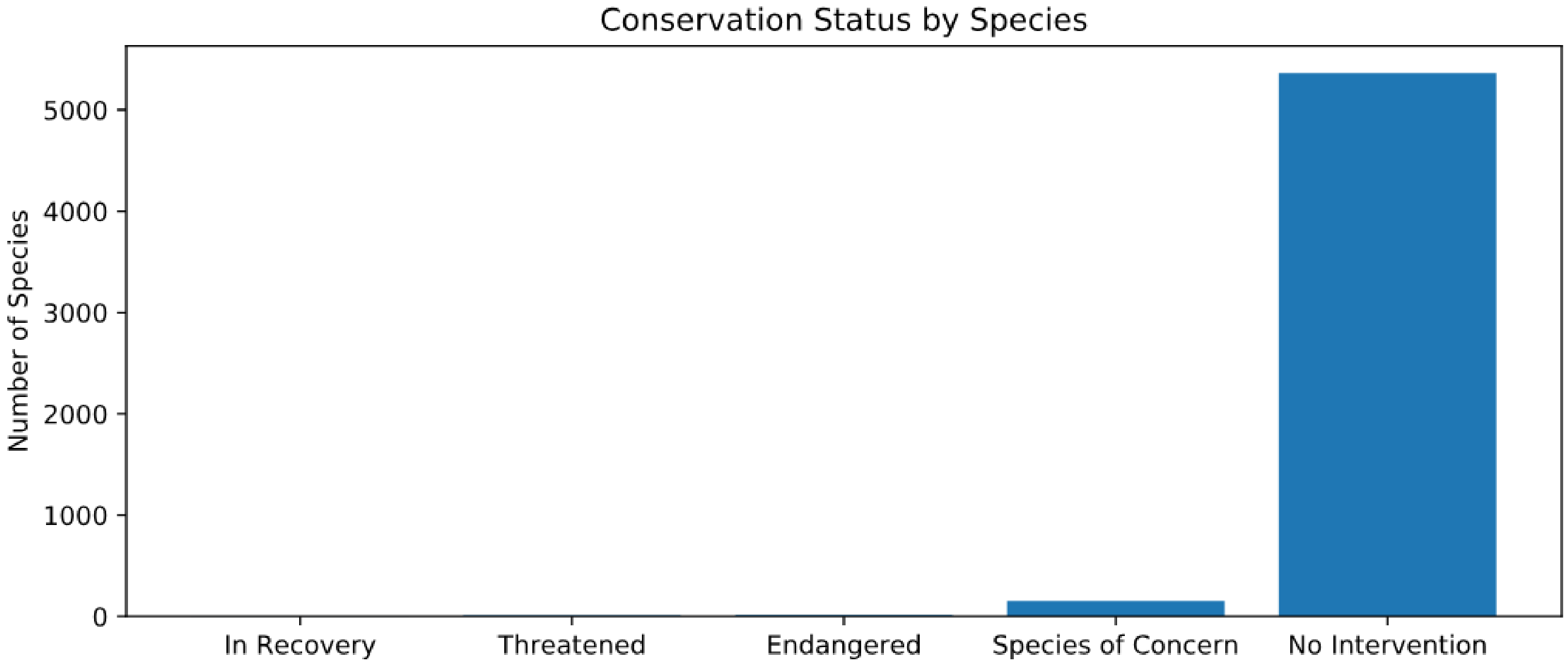
5. Plotting Conservation Status by Species



```python
plt.figure(figsize=(10, 4))
ax = plt.subplot()
plt.bar(range(len(protection_counts)),protection_counts.scientific_name.values)
ax.set_xticks(range(len(protection_counts)))
ax.set_xticklabels(protection_counts.conservation_status.values)
plt.ylabel('Number of Species')
plt.title('Conservation Status by Species')
plt.show()
```

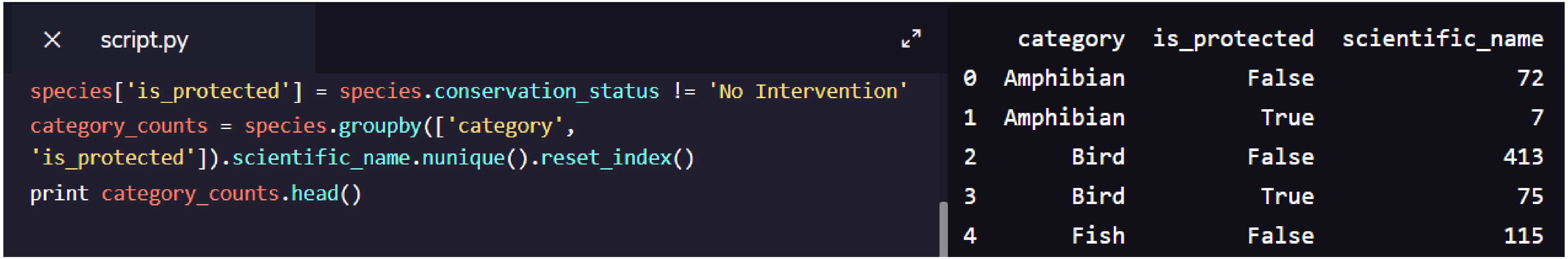
- Plotting a bar chart

5. Plotting Conservation Status by Species



6. Investigating Endangered Species

- **Are certain types of species more likely to be endangered**?



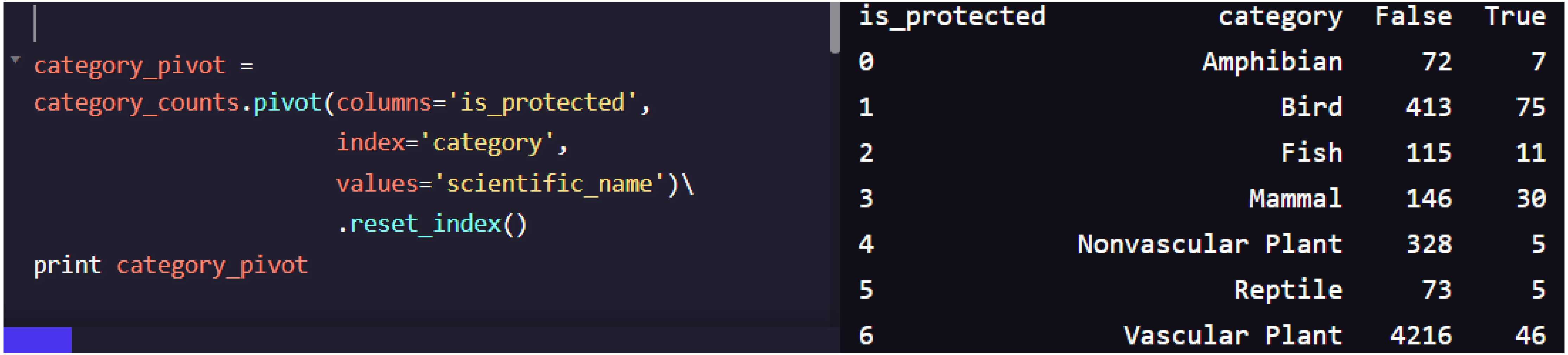
```
species['is_protected'] = species.conservation_status != 'No Intervention'
category_counts = species.groupby(['category',
'is_protected']).scientific_name.nunique().reset_index()
print category_counts.head()
```

```
    category  is_protected  scientific_name
0  Amphibian         False               72
1  Amphibian          True                7
2       Bird         False              413
3       Bird          True               75
4       Fish         False              115
```

Preparing the column is_protected for pivoting

6. Investigating Endangered Species

- Are certain types of species more likely to be endangered?



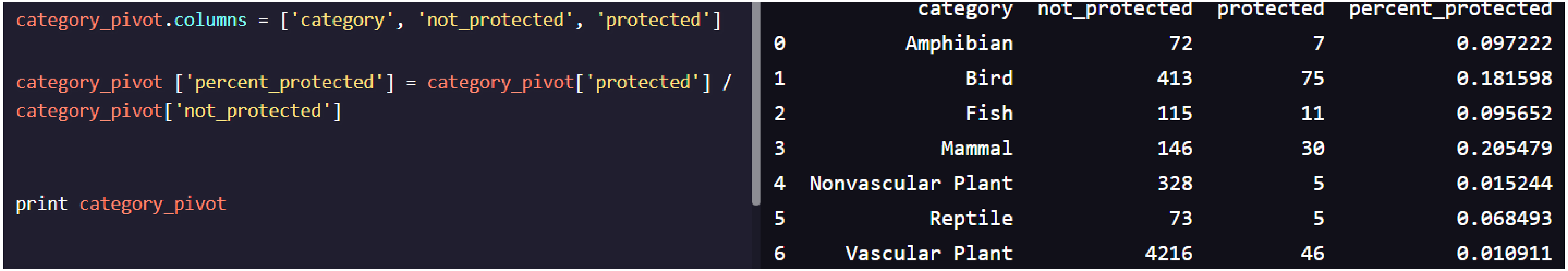
```
category_pivot =
category_counts.pivot(columns='is_protected',
                      index='category',
                      values='scientific_name')\
                      .reset_index()
print category_pivot
```

```
is_protected           category  False  True
0                     Amphibian     72     7
1                          Bird    413    75
2                          Fish    115    11
3                        Mammal    146    30
4             Nonvascular Plant    328     5
5                       Reptile     73     5
6                Vascular Plant   4216    46
```

Pivoting the table category_pivot

7. Investigating Endangered Species II

- Are certain types of species more likely to be endangered?



```
category_pivot.columns = ['category', 'not_protected', 'protected']

category_pivot ['percent_protected'] = category_pivot['protected'] /
category_pivot['not_protected']



print category_pivot
```

```
             category  not_protected  protected  percent_protected
0           Amphibian             72          7           0.097222
1                Bird            413         75           0.181598
2                Fish            115         11           0.095652
3              Mammal            146         30           0.205479
4   Nonvascular Plant            328          5           0.015244
5             Reptile             73          5           0.068493
6      Vascular Plant           4216         46           0.010911
```

Also calculated the percentage of protected species among each category to be aware that Mammals occupies the highest percentage

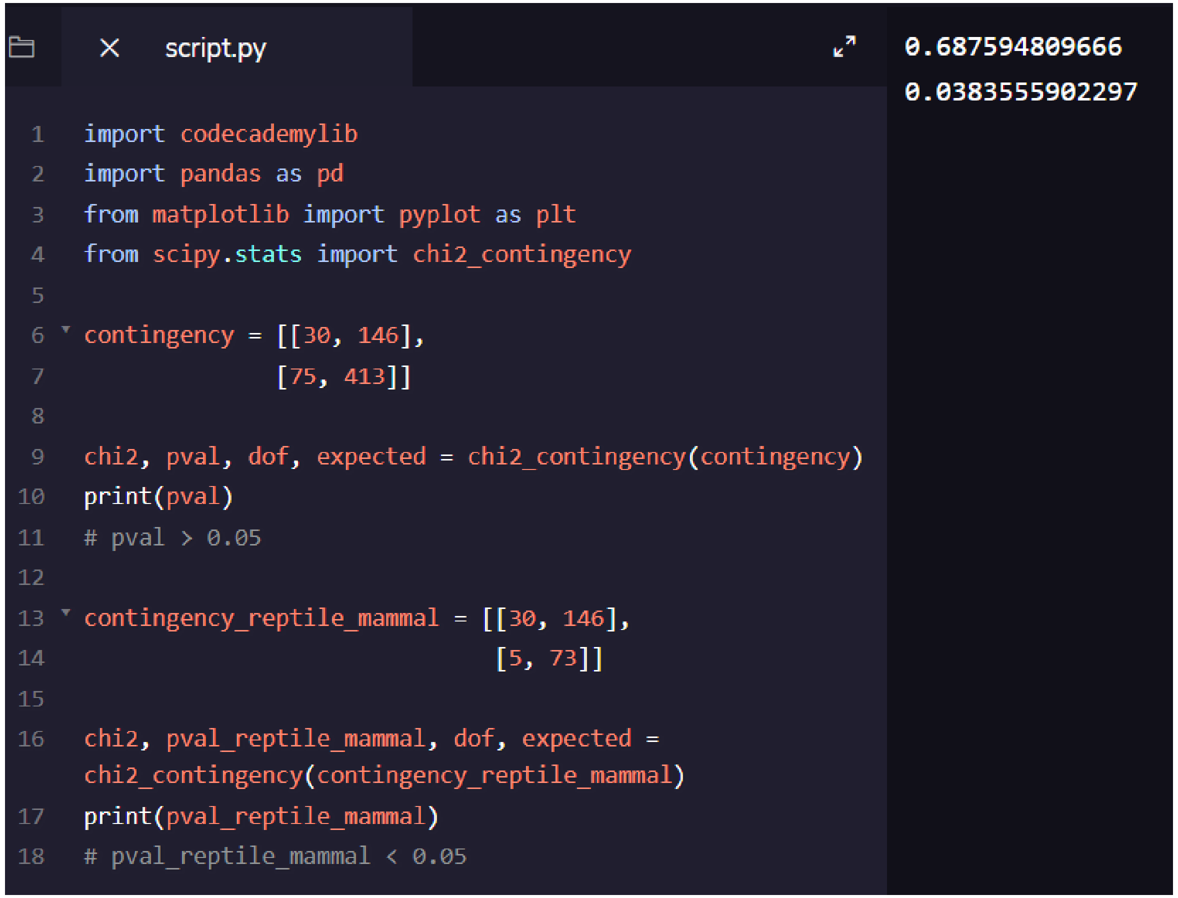
8. Chi-Squared Test for Significance

- **Hypothesis testing**

- Is the data numerical or categorical?
- categorical
- How many pieces of data are you comparing?
- Two pieces

- 0.688, insignificant
- 0.038 significant

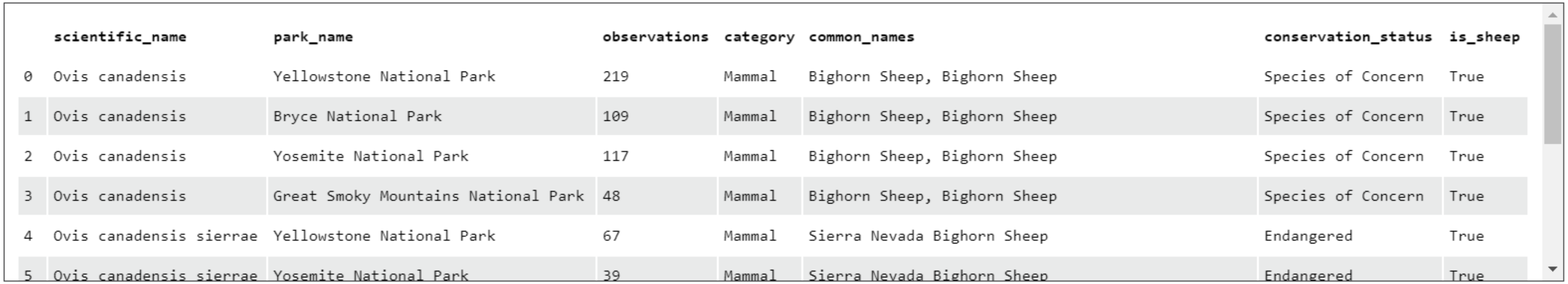
Conservationist will need to put into consideration that Mammals are very likely more endangered than Reptiles



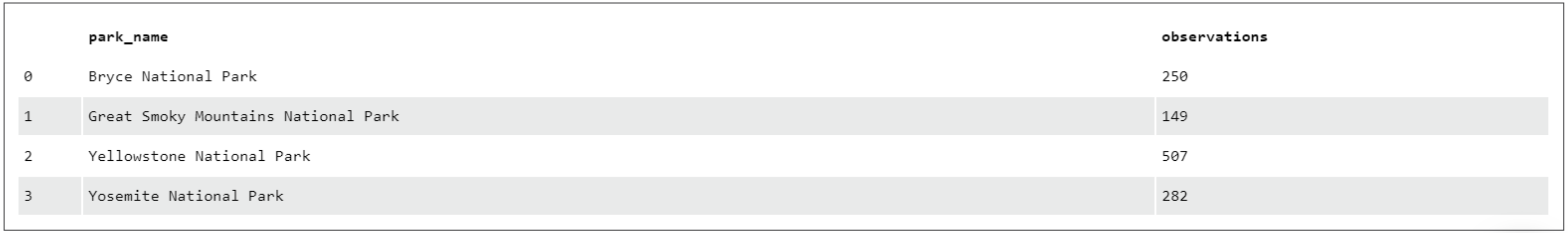
```python
import codecademylib
import pandas as pd
from matplotlib import pyplot as plt
from scipy.stats import chi2_contingency

contingency = [[30, 146],
               [75, 413]]

chi2, pval, dof, expected = chi2_contingency(contingency)
print(pval)
# pval > 0.05

contingency_reptile_mammal = [[30, 146],
                              [5, 73]]

chi2, pval_reptile_mammal, dof, expected = chi2_contingency(contingency_reptile_mammal)
print(pval_reptile_mammal)
# pval_reptile_mammal < 0.05
```

```
0.687594809666
0.0383555902297
```

10. Observations DataFrame



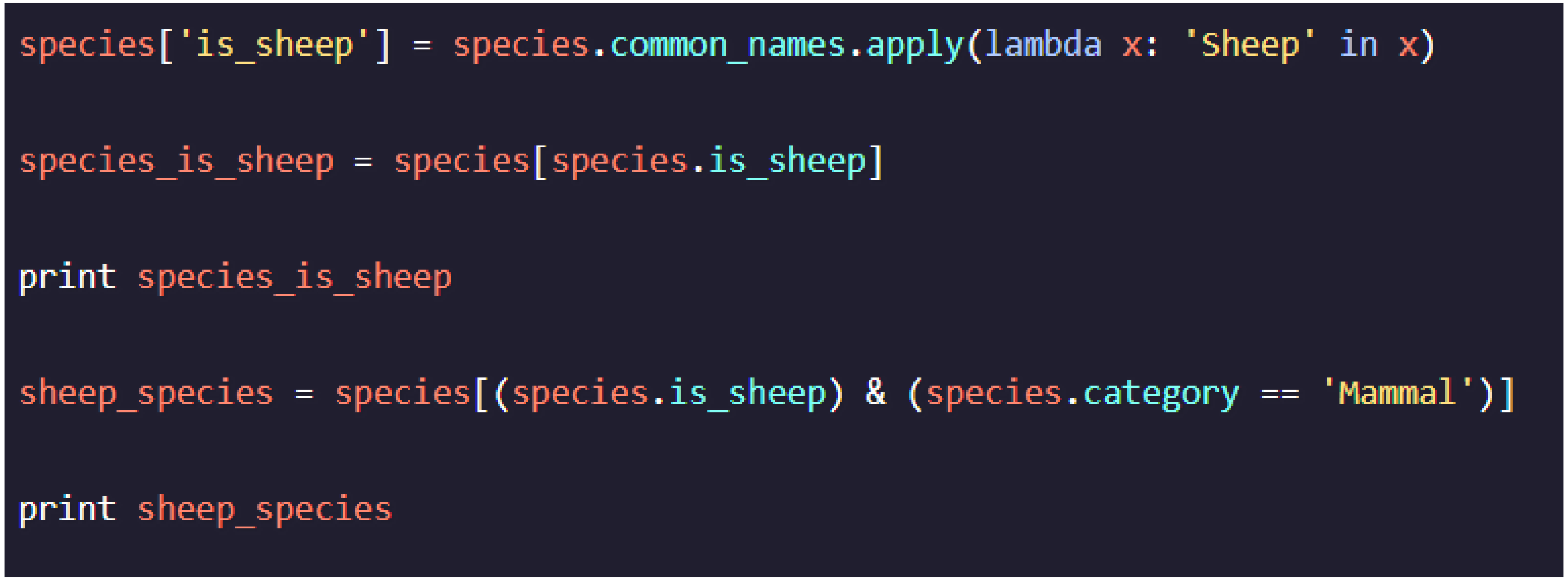
	scientific_name	park_name	observations	category	common_names	conservation_status	is_sheep
0	Ovis canadensis	Yellowstone National Park	219	Mammal	Bighorn Sheep, Bighorn Sheep	Species of Concern	True
1	Ovis canadensis	Bryce National Park	109	Mammal	Bighorn Sheep, Bighorn Sheep	Species of Concern	True
2	Ovis canadensis	Yosemite National Park	117	Mammal	Bighorn Sheep, Bighorn Sheep	Species of Concern	True
3	Ovis canadensis	Great Smoky Mountains National Park	48	Mammal	Bighorn Sheep, Bighorn Sheep	Species of Concern	True
4	Ovis canadensis sierrae	Yellowstone National Park	67	Mammal	Sierra Nevada Bighorn Sheep	Endangered	True
5	Ovis canadensis sierrae	Yosemite National Park	39	Mammal	Sierra Nevada Bighorn Sheep	Endangered	True



	park_name	observations
0	Bryce National Park	250
1	Great Smoky Mountains National Park	149
2	Yellowstone National Park	507
3	Yosemite National Park	282

- Visualize new observations dataFrame

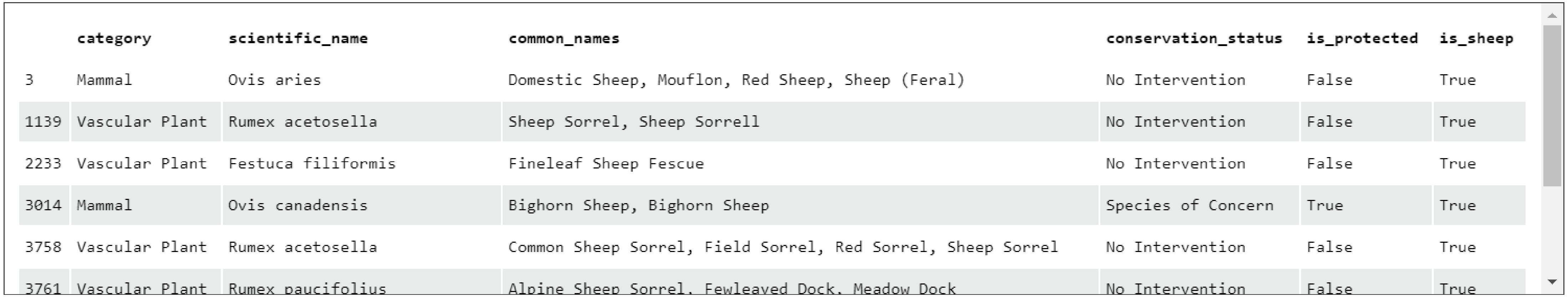
11. In Search of Sheep



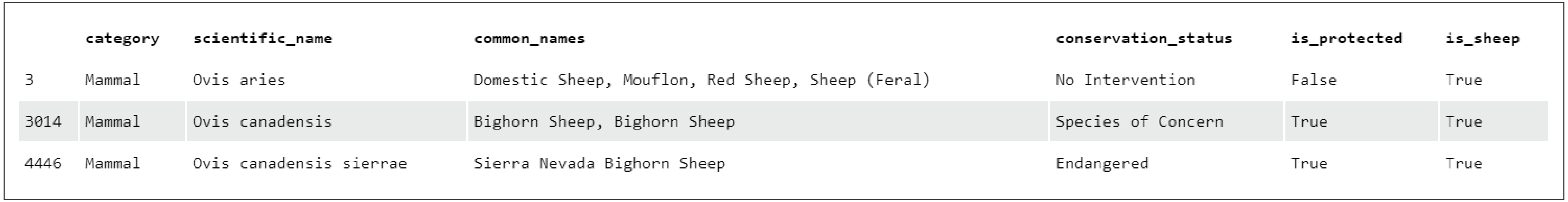
```
species['is_sheep'] = species.common_names.apply(lambda x: 'Sheep' in x)

species_is_sheep = species[species.is_sheep]

print species_is_sheep

sheep_species = species[(species.is_sheep) & (species.category == 'Mammal')]

print sheep_species
```

- Getting only sheep info we need from species

11. In Search of Sheep



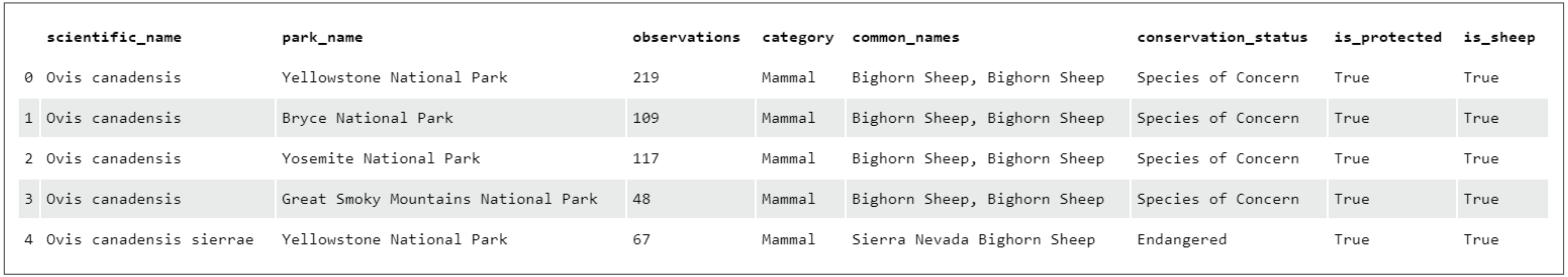
	category	scientific_name	common_names	conservation_status	is_protected	is_sheep
3	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep, Sheep (Feral)	No Intervention	False	True
1139	Vascular Plant	Rumex acetosella	Sheep Sorrel, Sheep Sorrell	No Intervention	False	True
2233	Vascular Plant	Festuca filiformis	Fineleaf Sheep Fescue	No Intervention	False	True
3014	Mammal	Ovis canadensis	Bighorn Sheep, Bighorn Sheep	Species of Concern	True	True
3758	Vascular Plant	Rumex acetosella	Common Sheep Sorrel, Field Sorrel, Red Sorrel, Sheep Sorrel	No Intervention	False	True
3761	Vascular Plant	Rumex paucifolius	Alpine Sheep Sorrel, Fewleaved Dock, Meadow Dock	No Intervention	False	True



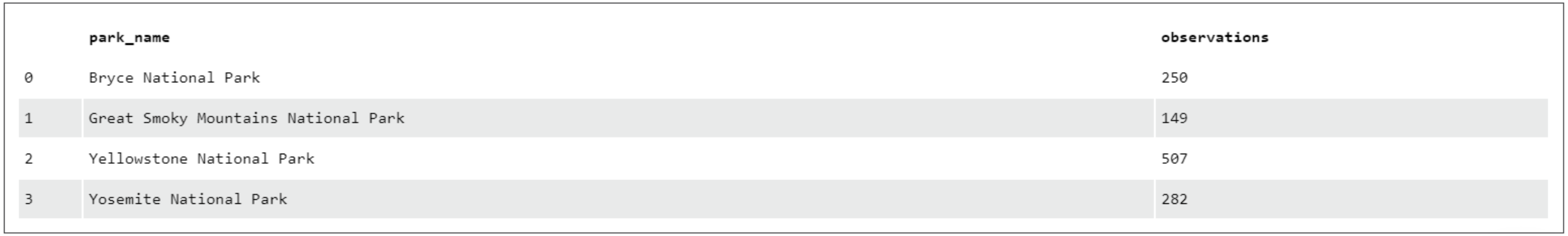
	category	scientific_name	common_names	conservation_status	is_protected	is_sheep
3	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep, Sheep (Feral)	No Intervention	False	True
3014	Mammal	Ovis canadensis	Bighorn Sheep, Bighorn Sheep	Species of Concern	True	True
4446	Mammal	Ovis canadensis sierrae	Sierra Nevada Bighorn Sheep	Endangered	True	True

- The second one is all the sheep is true and is mammal

12.Merging Sheep and Observation DataFrames



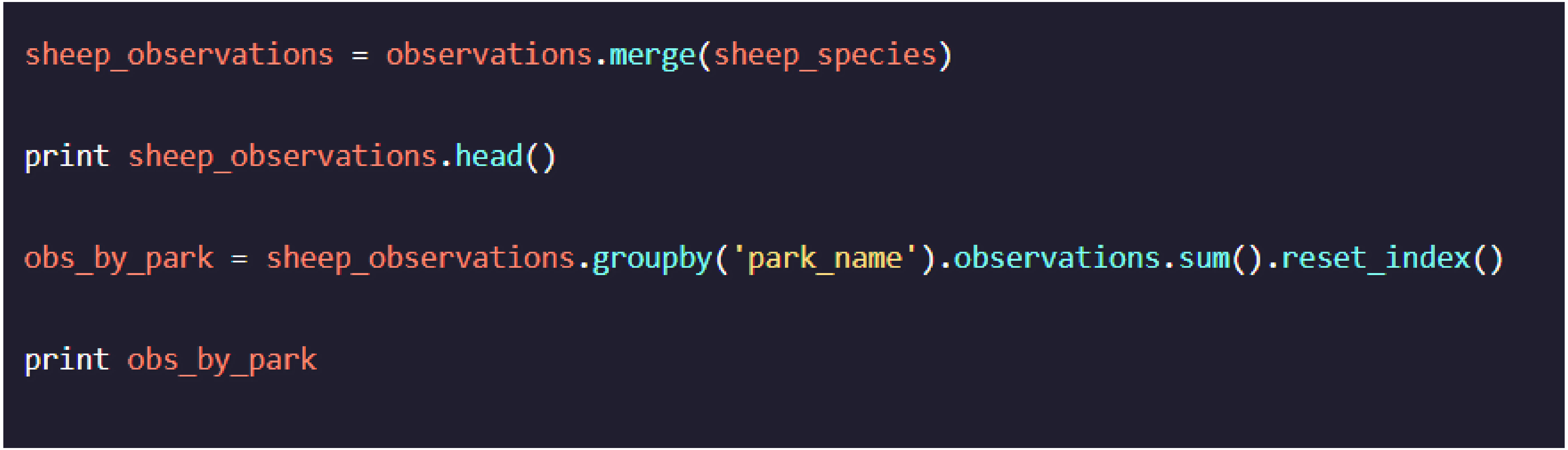
	scientific_name	park_name	observations	category	common_names	conservation_status	is_protected	is_sheep
0	Ovis canadensis	Yellowstone National Park	219	Mammal	Bighorn Sheep, Bighorn Sheep	Species of Concern	True	True
1	Ovis canadensis	Bryce National Park	109	Mammal	Bighorn Sheep, Bighorn Sheep	Species of Concern	True	True
2	Ovis canadensis	Yosemite National Park	117	Mammal	Bighorn Sheep, Bighorn Sheep	Species of Concern	True	True
3	Ovis canadensis	Great Smoky Mountains National Park	48	Mammal	Bighorn Sheep, Bighorn Sheep	Species of Concern	True	True
4	Ovis canadensis sierrae	Yellowstone National Park	67	Mammal	Sierra Nevada Bighorn Sheep	Endangered	True	True



	park_name	observations
0	Bryce National Park	250
1	Great Smoky Mountains National Park	149
2	Yellowstone National Park	507
3	Yosemite National Park	282

- Common column ‘park_name’ to merge

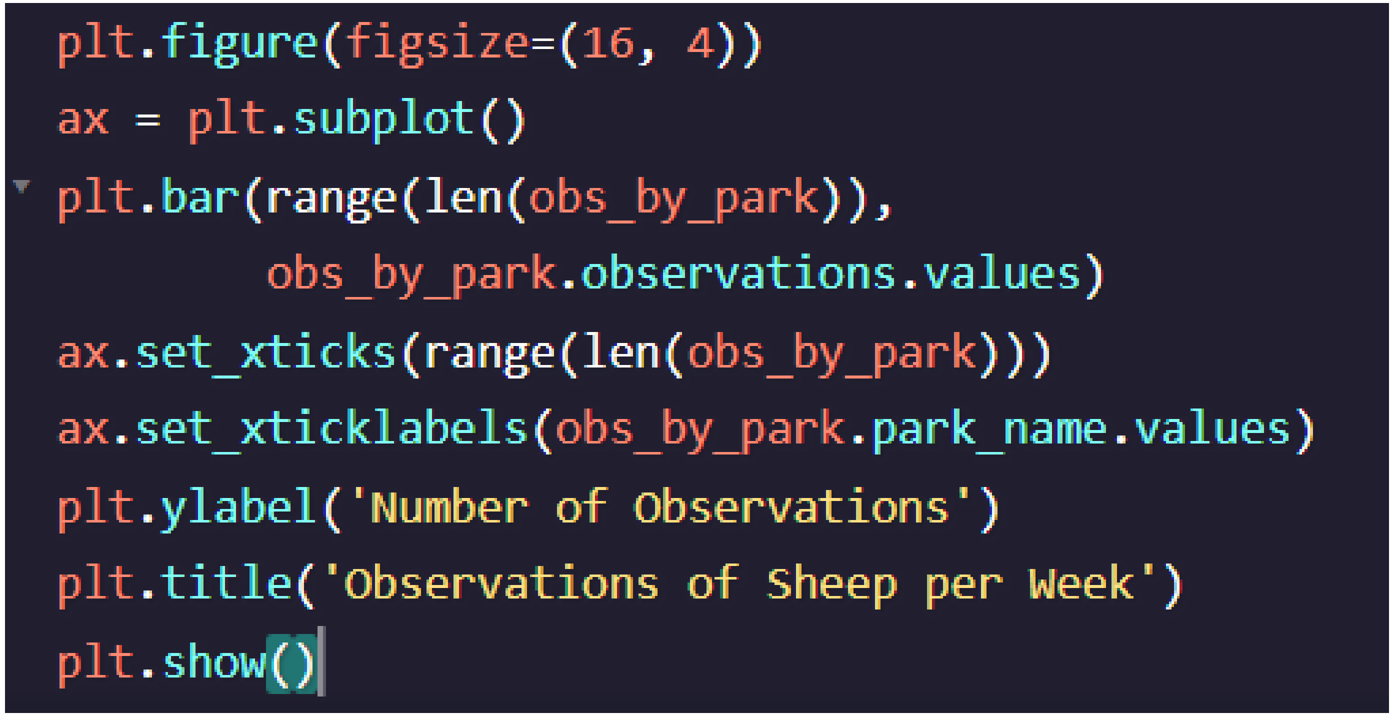
12.Merging Sheep and Observation DataFrames



```
sheep_observations = observations.merge(sheep_species)

print sheep_observations.head()

obs_by_park = sheep_observations.groupby('park_name').observations.sum().reset_index()

print obs_by_park
```

- Sum up the total observations

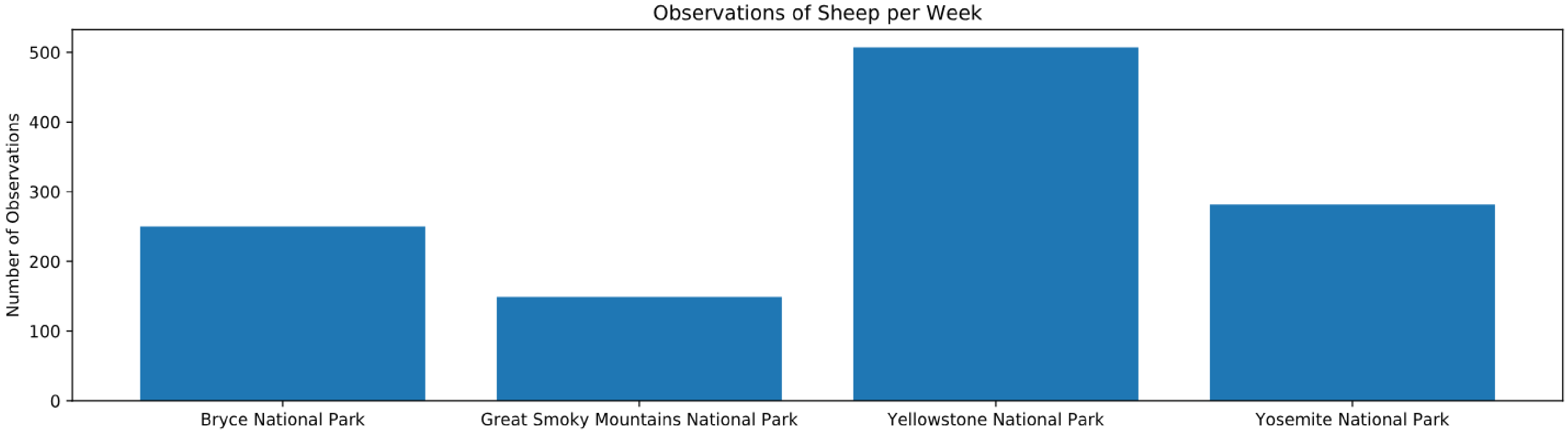
13. Plotting Sheep Sightings



```
plt.figure(figsize=(16, 4))
ax = plt.subplot()
plt.bar(range(len(obs_by_park)),
        obs_by_park.observations.values)
ax.set_xticks(range(len(obs_by_park)))
ax.set_xticklabels(obs_by_park.park_name.values)
plt.ylabel('Number of Observations')
plt.title('Observations of Sheep per Week')
plt.show()
```

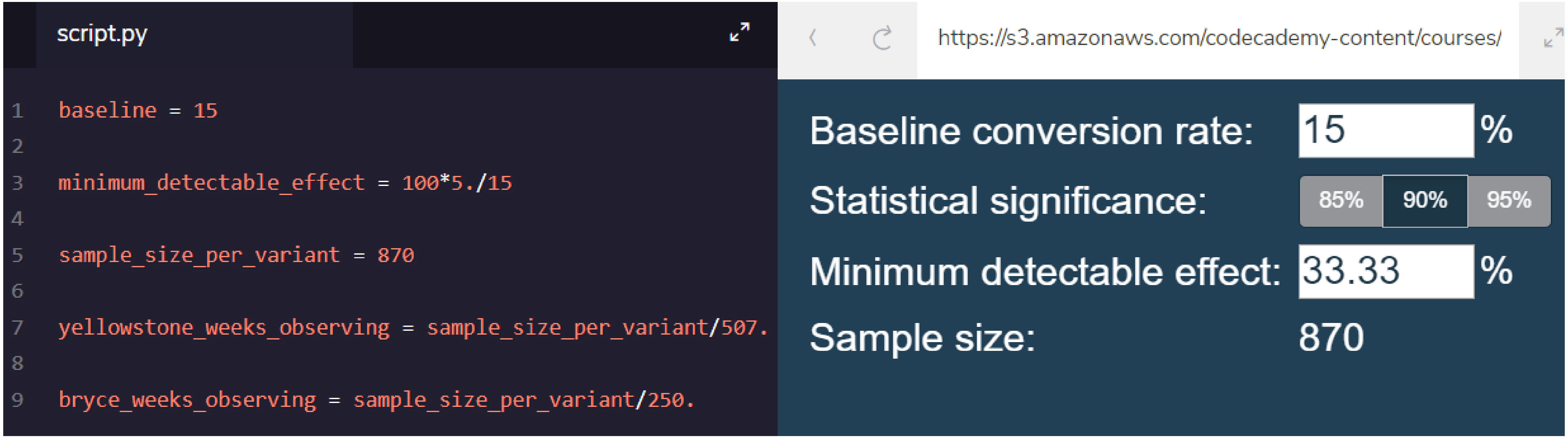
- Bar plotting obs_by_park

13. Plotting Sheep Sightings



- Yellowstone has the most number of observations

14. Foot and Mouth Reduction Effort - Sample Size Determination



```python
baseline = 15

minimum_detectable_effect = 100*5./15

sample_size_per_variant = 870

yellowstone_weeks_observing = sample_size_per_variant/507.

bryce_weeks_observing = sample_size_per_variant/250.
```

Baseline conversion rate: 15 %

Statistical significance: 85% [✓ 90%] 95%

Minimum detectable effect: 33.33 %

Sample size: 870

- Sample size determination helps to estimate how many we need to observe and how long it will take in order to reach our goal