

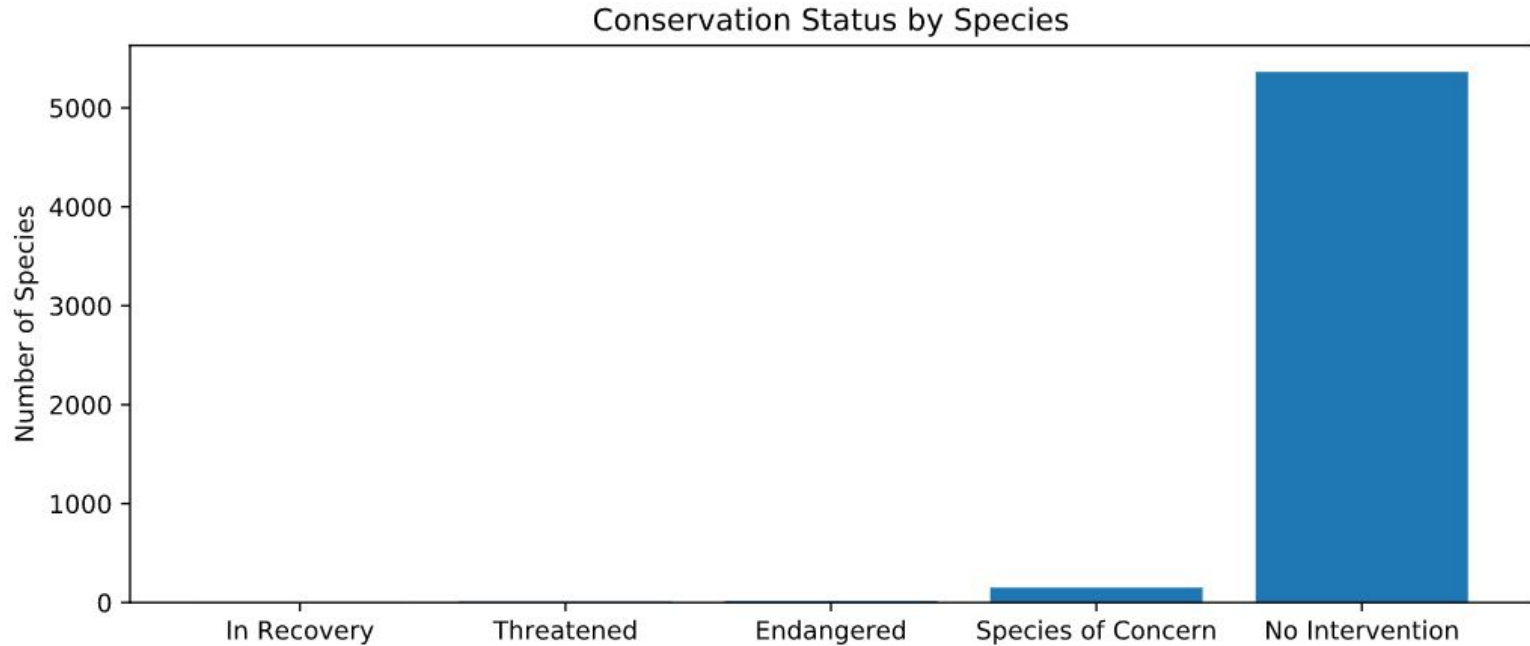
# Biodiversity!

**Codecademy Capstone:  
JamieHo**

**Bias in protection?**

---

**Don't forget them!!**



There's some species need our special attention

# And those attention is NOT well distributed

```
11
12 category_counts =
   species.groupby(["category",
   "is_protected"]).scientific_name.count().
   reset_index()
13 #category_counts =
   species.groupby(['category',
   'is_protected']).scientific_name.nunique(
   ).reset_index()
14
15 print category_counts.head()
16
17 category_pivot =
   category_counts.pivot(columns='is_protect
   ed',
18                        index='category',
```

	is_protected	category	not_protected	protected	percent_protected
0		Amphibian	73	7	0.087500
1		Bird	442	79	0.151631
2		Fish	116	11	0.086614
3		Mammal	176	38	0.177570
4		Nonvascular Plant	328	5	0.015015
5		Reptile	74	5	0.063291
6		Vascular Plant	4424	46	0.010291

Bird and Mammal is in the spotlight.

Don't forget others!

# Truely!

```
))
category_pivot["percent_protected"]=category_p
pivot["protected"]/(category_pivot["protected"]
category_pivot["not_protected"])

print category_pivot

from scipy.stats import chi2_contingency

contingency = [[38, 176],
               [79, 443]]

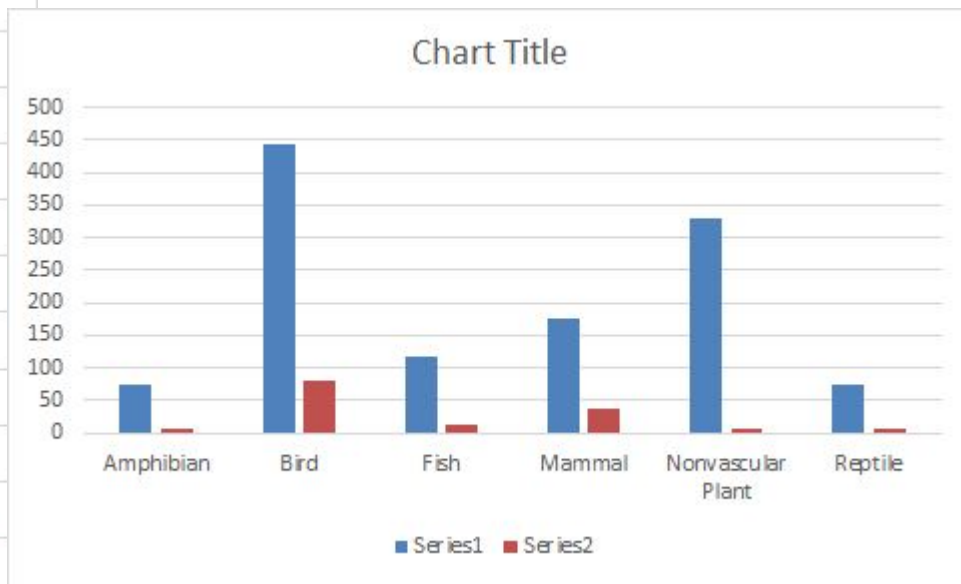
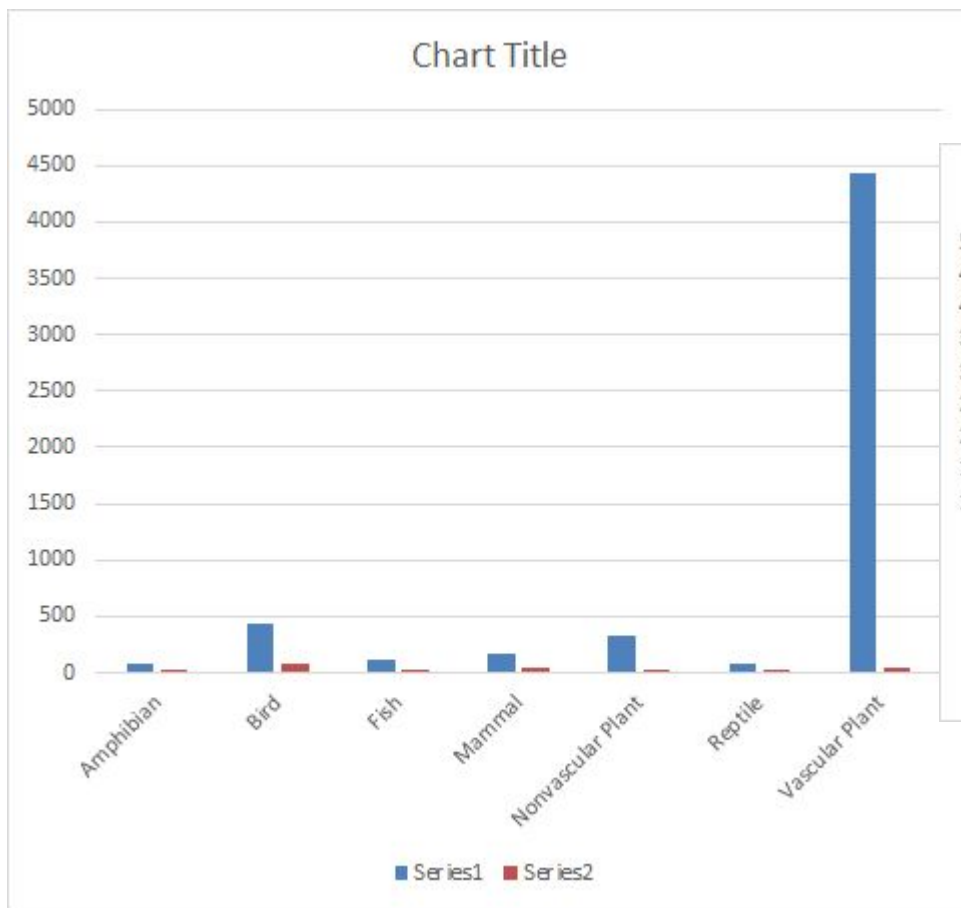
pval = chi2_contingency(contingency)[1]
print(pval)

No significant difference because pval >
0.05

contingency_reptile_mammal = [[38, 176],
                              [5, 74]]
```

is_protected	category	not_protected	protected	percent_protected
0	Amphibian	73	7	0.08750
1	Bird	442	79	0.15163
2	Fish	116	11	0.08661
3	Mammal	176	38	0.17757
4	Nonvascular Plant	328	5	0.01501
5	Reptile	74	5	0.06329
6	Vascular Plant	4424	46	0.01029

0.43967179738  
**0.0233846521487**



Series1: not\_protected

Series2: protected

**How many week ?  
For Baaa!**

---

# There's the sheep

codecademy

Biodiversity Capstone Project

script.py

```
1 import codecademylib
2 import pandas as pd
3 from matplotlib import pyplot as plt
4
5 species = pd.read_csv('species_info.csv')
6 species.fillna('No Intervention', inplace = True)
7 species['is_protected'] = species.conservations_status !=
8 'No Intervention'
9
10 observations = pd.read_csv('observations.csv')
11
12 species['is_sheep'] = species.common_names.apply(lambda
13 x: 'Sheep' in x)
14
15 #species['is_sheep'] = species.common_names.apply(lambda
16 x: "True" if "Sheep" in x else "False")
17
18 species_is_sheep = species[species.is_sheep==True]
19
20 #We taught this before??!!
21 #species_is_sheep = species[species.is_sheep]
22
23 print species_is_sheep
24
25 sheep_species = species[(species.is_sheep==True) &
26 (species.category == 'Mammal')]
```

Did your lambda correctly identify if `common_name` contained 'Sheep'?

Run



https://localhost/

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

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



# Sheep it is! In Yellowstone, which I plan to go some day.

< Biodiversity Capstone Project

script.py

```
10
11 species['is_sheep'] = species.common_names.apply(lambda x:
12 'Sheep' in x)
13 #species["is_sheep"]=species.common_names.apply(lambda
14 x:"True" if "Sheep" in x else "False")
15 species_is_sheep = species[species.is_sheep==True]
16
17 #We taught this before??!!
18 #species_is_sheep = species[species.is_sheep]
19
20 print species_is_sheep
21
22 sheep_species = species[(species.is_sheep==True) &
23 (species.category == 'Mammal')]
24
25 print sheep_species
26
27 sheep_observations = pd.merge(sheep_species, observations,
28 how='left')
29
30 print sheep_observations.head()
31
32 obs_by_park =
33 sheep_observations.groupby("park_name").observations.sum().r
34 eset_index()
35
36 print obs_by_park
```

Run



< ↺ https://localhost/ ↻

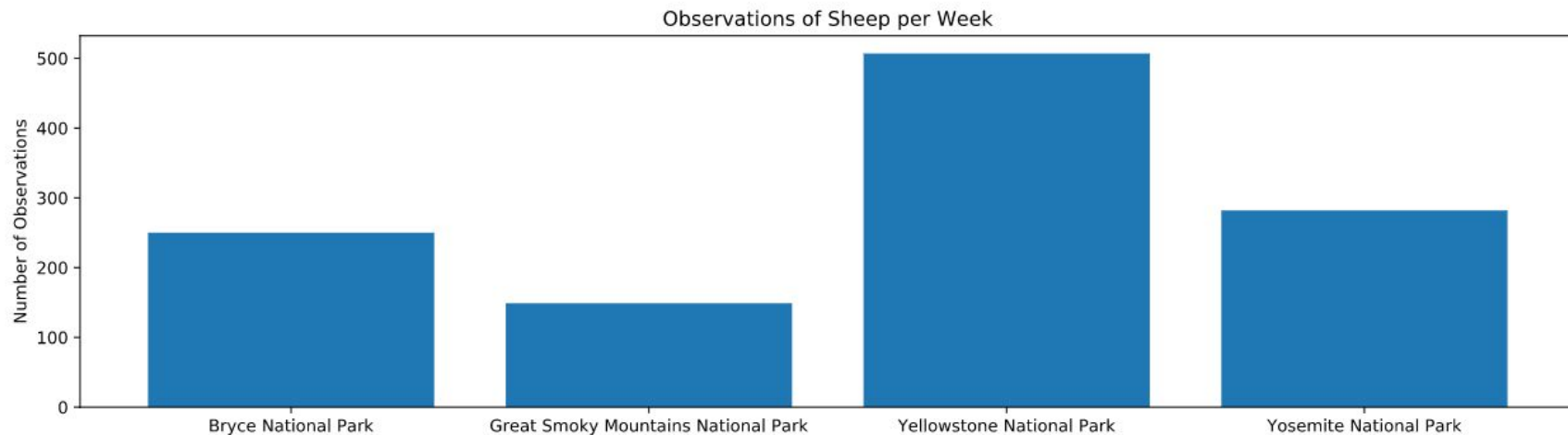
	category	scientific_name	common_names
--	----------	-----------------	--------------

0	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep,
1	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep,
2	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep,
3	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep,
4	Mammal	Ovis canadensis	Bighorn Sheep, Bighorn Sheep

	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

# Sheep chart



# For meaningful out come, we have to stay for 1.7 week in YS

For sample size 870, we have to stay in Yellowstone for 1.72 week; Bryce, 3.48!

script.py

```
1 baseline = 15
2
3 minimum_detectable_effect = 100*5./15
4
5 sample_size_per_variant = 870
6
7 yellowstone_weeks_observing = sample_size_per_variant/507.
8
9 bryce_weeks_observing = sample_size_per_variant/250.
10
11 print(sample_size_per_variant, yellowstone_weeks_observing,
12       bryce_weeks_observing)
13 #Bug??
14 #baseline = 15
15 #minimum_detectable_effect=100*5/baseline
16 #sample_size_per_variant=870
17 #yellowstone_weeks_observing=870/507
18 #bryce_weeks_observing=870/250
```

(870, 1.7159763313609468, 3.48)