Capstone Option 2: Biodiversity for the National Parks

Code task 4/15

```
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()
print(species_count)
species_type=species.category.unique()
print(species_type)
conservation_statuses=species.conservation_status.unique()
conservation_counts=species.groupby("conservation_status").scientific_name.nunique().reset_index()
print(conservation_counts)
species.fillna('No Intervention', inplace = True)
conservation_counts_fixed=species.groupby("conservation_status").scientific_name.nunique().reset_index()
print(conservation_counts_fixed)
```

2.1. There is 5 541.- different species in DataFrame

2.2. There are 7 different categories:

```
['Mammal' 'Bird' 'Reptile' 'Amphibian' 'Fish'
'Vascular Plant'
 'Nonvascular Plant']
```

2.3. The different values of conservation status are these 5:

```
[nan 'Species of Concern' 'Endangered' 'Threatened'
'In Recovery']
```

3.1, 3.2

By grouping by conservation status we see the number of species in each conservation status

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

4.1, 4.2

After replacing of NaN with “No Intervention” it is able to see that there are 5363 species without intervention

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

Code task 5/15

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

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

species.fillna('No Intervention', inplace = True)

protection_counts = species.groupby('conservation_status')\
    .scientific_name.nunique().reset_index()\
    .sort_values(by='scientific_name')
print(protection_counts)
plt.figure (figsize=(10,4))
ax = plt.subplot
plt.bar (range(len(protection_counts.conservation_status)),protection_counts.scientific_name)
ax.set_xticks(range(len(protection_counts.conservation_status)))
ax.set_xticklabels(protection_counts.conservation_status)
plt.ylabel("Number of Species")
plt.title("Conservation Status by Species")
plt.show()
```

5.1
The new data frame protection_counts was created as follows:

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

5.2
Unfortunately I forgot to copy the graph

Code task 7/15

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

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

species.fillna('No Intervention', inplace = True)

species["is_protected"]=species.conservation_status.apply(lambda x: True if x !="No Intervention" else False)
category_counts=species.groupby(["category", "is_protected"]).scientific_name.nunique().reset_index()
print(category_counts.head())
category_pivot=category_counts.pivot(columns="is_protected", index = "category", values = "scientific_name").reset_index()
print(category_pivot)
category_pivot.columns=["category", "not_protected", "protected"]
category_pivot["percent_protected"]=category_pivot.protected/(category_pivot.protected+category_pivot.not_protected)
print(category_pivot)
```

6.2, 6.3

After grouping by category and is protected we have a table as follows:

	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

6.4, 6.5

The created pivot from mentioned table is:

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

7.1, 7.2, 7.3

The pivot after renaming and calculation of percentage is:

The percentage in the pivot is the percent of protected species from all species in each category.

	category	not_protected	protected	percent_protected
0	Amphibian	72	7	0.088608
1	Bird	413	75	0.153689
2	Fish	115	11	0.087302
3	Mammal	146	30	0.170455
4	Nonvascular Plant	328	5	0.015015
5	Reptile	73	5	0.064103
6	Vascular Plant	4216	46	0.010793

Code task 8/15

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

# Loading the Data
species = pd.read_csv('species_info.csv')

# print species.head()

# Inspecting the DataFrame
species_count = len(species)

species_type = species.category.unique()

conservation_statuses = species.conservation_status.unique()

# Analyze Species Conservation Status
conservation_counts =
species.groupby('conservation_status').scientific_name.count().reset_index()

# print conservation_counts

# Analyze Species Conservation Status II
species.fillna('No Intervention', inplace = True)

conservation_counts_fixed =
species.groupby('conservation_status').scientific_name.count().reset_index()

# Plotting Conservation Status by Species
protection_counts = species.groupby('conservation_status')\
    .scientific_name.count().reset_index()\
    .sort_values(by='scientific_name')

# 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')
```

```
# labels = [e.get_text() for e in ax.get_xticklabels()]
# print ax.get_title()
# plt.show()

species['is_protected'] = species.conservation_status != 'No Intervention'

category_counts = species.groupby(['category', 'is_protected'])\
                  .scientific_name.count().reset_index()

# print category_counts.head()

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

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

category_pivot['percent_protected'] = category_pivot.protected / (category_pivot.protected +
category_pivot.not_protected)

print category_pivot.head()
contingency=[[30,146],[75, 413]]
from scipy.stats import chi2_contingency
chi2, pval, dof, expected=chi2_contingency(contingency)
contingency2=[[30,146],[5, 73]]
chi2, pval, dof, expected=chi2_contingency(contingency2)
pval_reptile_mammal=pval
print(pval_reptile_mammal)
```

8.1, 8.2, 8.3, 8.4

The Pval from chi-squared test:

0.688 - The difference between Mammal and Bird is not significant

0.038 - The difference between Mammal and Reptile is significant

Code task 12/15

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

species = pd.read_csv('species_info.csv')
species.fillna('No Intervention', inplace = True)
species['is_protected'] = species.conservation_status != 'No Intervention'

observations=pd.read_csv("observations.csv")
print(observations.head(10))
species["is_sheep"]=species.common_names.apply(lambda x: True if "Sheep" in x else False)
species_is_sheep = species[species.is_sheep == True]
print(species_is_sheep.head(10))
sheep_species = species[(species.is_sheep == True) & (species.category == "Mammal")]
print(sheep_species.head(10))
sheep_observations=pd.merge(sheep_species, observations)
print(sheep_observations.head(10))
obs_by_park=sheep_observations.groupby("park_name").observations.sum().reset_index()
print(obs_by_park)
```

10.1, 10.2

Observation table include scietific_name, park_name and number of observations:

	scientific_name	park_name	observations
0	Vicia benghalensis	Great Smoky Mountains National Park	68
1	Neovison vison	Great Smoky Mountains National Park	77
2	Prunus subcordata	Yosemite National Park	138
3	Abutilon theophrasti	Bryce National Park	84
4	Githopsis specularioides	Great Smoky Mountains National Park	85
5	Elymus virginicus var. virginicus	Yosemite National Park	112

11.1, 11.2, 11.3

The table after using lambda creating new column is_sheep which is true when a name contains “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

11.4, 11.5

Table “sheep species”after selecting only category “Mammal” and is_sheep “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

12.1, 12.2

Table “sheep observations” after merging tables “sheep species” with “observations”

	category	scientific_name	common_names	conservation_status	is_protected	is_sheep	park_name	observatio
0	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep, Sheep (Feral)	No Intervention	False	True	Yosemite National Park	126
1	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep, Sheep (Feral)	No Intervention	False	True	Great Smoky Mountains National Park	76
2	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep, Sheep (Feral)	No Intervention	False	True	Bryce National Park	119
3	Mammal	Ovis aries	Domestic Sheep, Mouflon, Red Sheep, Sheep (Feral)	No Intervention	False	True	Yellowstone National Park	221
4	Mammal	Ovis canadensis	Bighorn Sheep, Bighorn Sheep	Species of Concern	True	True	Yellowstone National Park	219
5	Mammal	Ovis canadensis	Bighorn Sheep, Bighorn Sheep	Species of Concern	True	True	Bryce National Park	109

12.3, 12.4

Table obs_by_park after grouping observations for each park name

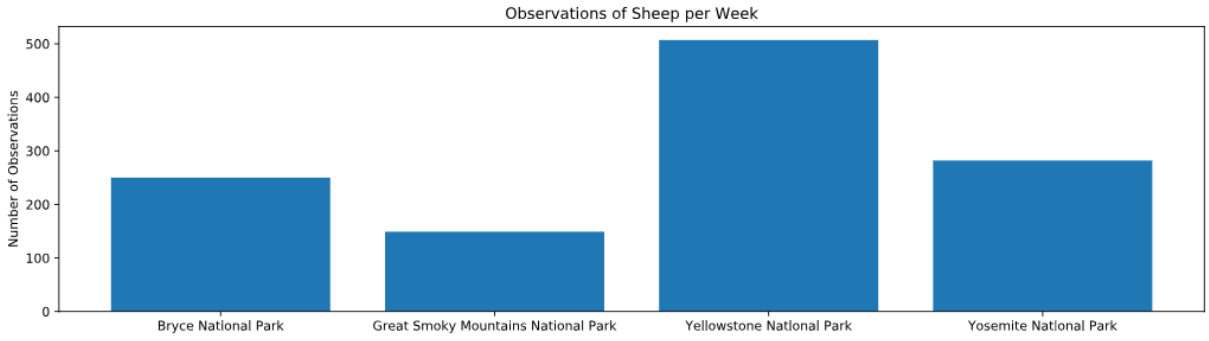
	park_name	observations
0	Bryce National Park	250
1	Great Smoky Mountains National Park	149
2	Yellowstone National Park	507
3	Yosemite National Park	282

Code task 13/15

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

species = pd.read_csv('species_info.csv')
species['is_sheep'] = species.common_names.apply(lambda x: 'Sheep' in x)
sheep_species = species[(species.is_sheep) & (species.category == 'Mammal')]

observations = pd.read_csv('observations.csv')

sheep_observations = observations.merge(sheep_species)

obs_by_park = sheep_observations.groupby('park_name').observations.sum().reset_index()
plt.figure(figsize=(16, 4))
ax=plt.subplot()
parks=obs_by_park.park_name.unique
week_observations=obs_by_park.observations
plt.bar(range(len(parks())), week_observations)
ax.set_xticks(range(len(obs_by_park)))
ax.set_xticklabels(obs_by_park.park_name)
plt.ylabel("Number of Observations")
plt.title("Observations of Sheep per Week")
plt.show()
```

13.1

Bar chart showing observation of Sheeps per week in each park:



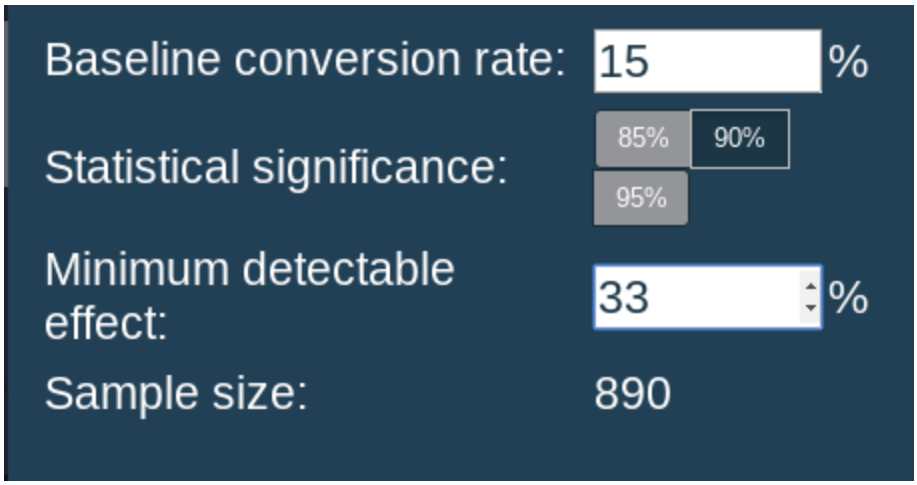
14.1
baseline=15

- 15% of sheep at Bryce National Park have foot and mouth disease

14.2
minimum_detectable_effect=.05/.15

- Calculation for detection of reduction by 5%

14.3
sample_size_per_variant=890



14.4
yellowstone_weeks_observing=890.0/507

14.5
bryce_weeks_observing=890.0/250