

# Notebook

September 11, 2020

Local date & time is : 09/11/2020 19:31:11 PDT

```
In [92]: import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from scripts.espm_module import *
import json
from pandas import json_normalize
from IPython.core.display import display, HTML
plt.style.use('seaborn')
%matplotlib inline
```

YOUR RESPONSE HERE  
YOUR RESPONSE HERE  
YOUR RESPONSE HERE  
YOUR RESPONSE HERE

```
In [112]: records_df = json_normalize(records)  # converts the JSON above to a dataframe
records_df.head()                             # prints the first five rows of
```

```
Out[112]:
```

	key	datasetKey	\
0	2596088009	50c9509d-22c7-4a22-a47d-8c48425ef4a7	
1	2597716824	50c9509d-22c7-4a22-a47d-8c48425ef4a7	
...	Omitting 43 lines	...	...
4	NaN	NaN	NaN

[5 rows x 122 columns]

```
In [113]: def get_species_records_df(scientific_name):
records = get_species_records(scientific_name) # Get the records using the f
records_df = json_normalize(records) # Convert the raw records into a DataFrm
return records_df
```

YOUR RESPONSE HERE  
YOUR RESPONSE HERE  
YOUR RESPONSE HERE  
YOUR RESPONSE HERE  
YOUR RESPONSE HERE

<b>EXERCISE 1:</b>  
<br />

Edit the code below to search for a species you're interested in, then use the graphing cell.  
Please note that a species name has two parts, the first part is the genus name and the second part is the species name.

**Hint:** Here's what the code looks like if we used *Argia agrioides* again:

```
my_records_df = get_species_records_df('Argia agrioides')
my_records_df.head()
```

```
In [130]: # Replace the ... in the line below with your species
my_records_df = get_species_records_df("...")
my_records_df.head() # Show the first 5 records
```

```
Out[130]:
```

	key	datasetKey	\
0	2596088009	50c9509d-22c7-4a22-a47d-8c48425ef4a7	
1	2597716824	50c9509d-22c7-4a22-a47d-8c48425ef4a7	
... Omitting 43 lines ...			
4	NaN	NaN	NaN

[5 rows x 122 columns]

Note: If the species you chose is distributed globally, the above plot may look a bit odd or the cell may have errored. Uncomment the cell below (you can highlight the entire cell and enter Control + / on PC or Command + / on Mac on your keyboard) and run it to see the same chart but with countries instead of counties.

```
In [136]: # my_records_df.groupby(["country", "collectionCode"])['collectionCode'].count()
# plt.title('Species Distribution by Country and Collection');
# plt.xlabel('Count');
# plt.ylabel('Country');
```

YOUR RESPONSE HERE

<b>QUESTION 11:</b>

<br />

What is the elevation range of your organism?

<br/>

<i>Note:</i> Depending on the species you have chosen, your data frame might not have any v

YOUR RESPONSE HERE

<b>QUESTION 12:</b>

<br />

Which has more: observations or preserved specimens? Why might this be? Are there any addit

YOUR RESPONSE HERE

---

**Make sure that you've answered questions 1-12. Also make sure you've done the 1 code exercise.**

You are finished with this notebook! Please run the following cell to generate a download link for your submission file to submit on bCourses. *It is very likely that this download link will not work. If the download link does not work, please use the alternate download method described below.*

Alternate download instructions: - open a new tab and go to <https://datahub.berkeley.edu> - go to the "fall2019" folder - click the box next to notebook1\_submission.pdf - click the "Download" link below the menu bar

**Check the PDF before submitting and make sure all of your answers & code changes are shown.**