


[learn-co-curriculum](#) / [dsc-exploring-and-transforming-json-schemas-lab](#) Public View license 0 stars  196 forks Star Watch ▾[Code](#) [Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) master ▾

...



hoffm386 fix objectives spacing ...

on Aug 1 ⌚ 16

[View code](#) README.md

Exploring and Transforming JSON Schemas - Lab

Introduction

In this lab, you'll practice exploring a JSON file whose structure and schema is unknown to you. We will provide you with limited information, and you will explore the dataset to answer the specified question.

Objectives

You will be able to:

- Use the `json` module to load and parse JSON documents
- Explore and extract data using unknown JSON schemas
- Convert JSON to a pandas dataframe

Your Task: Create a Bar Graph of the Top 10 States with the Highest Asthma Rates for Adults Age 18+

The information you need to create this graph is located in `disease_data.json`. It contains both data and metadata.

You are given the following codebook/data dictionary:

- The actual data values are associated with the key `'DataValue'`
- The state names are associated with the key `'LocationDesc'`
- To filter to the appropriate records, make sure:
 - The `'Question'` is `'Current asthma prevalence among adults aged >= 18 years'`
 - The `'StratificationCategoryID1'` is `'OVERALL'`
 - The `'DataValueTypeID'` is `'CRDPREV'`
 - The `'LocationDesc'` is NOT `'United States'`

The provided JSON file contains both data and metadata, and you will need to parse the metadata in order to understand the meanings of the values in the data.

No further information about the structure of this file is provided.

Load the JSON File

Load the data from the file `disease_data.json` into a variable `data`.

```
# Your code here
```

Explore the Overall Structure

What is the overall data type of `data`?

```
# Your code here
```

What are the keys?

```
# Your code here
```

What are the data types associated with those keys?

```
# Your code here (data)
```

```
# Your code here (metadata)
```

Perform additional exploration to understand the contents of these values. For dictionaries, what are their keys? For lists, what is the length, and what does the first element look like?

```
# Your code here (add additional cells as needed)
```

As you likely identified, we have a list of lists forming the `'data'`. In order to make sense of that list of lists, we need to find the meaning of each index, i.e. the names of the columns.

Identify the Column Names

Look through the metadata to find the *names* of the columns, and assign that variable to `column_names`. This should be a list of strings. (If you just get the values associated with the `'columns'` key, you will have a list of dictionaries, not a list of strings.)

```
# Your code here (add additional cells as needed)
```

The following code checks that you have the correct column names:

```
# Run this cell without changes

# 42 total columns
assert len(column_names) == 42

# Each name should be a string, not a dict
assert type(column_names[0]) == str and type(column_names[-1]) == str

# Check that we have some specific strings
assert "DataValue" in column_names
assert "LocationDesc" in column_names
assert "Question" in column_names
assert "StratificationCategoryID1" in column_names
assert "DataValueTypeID" in column_names
```

Filter Rows Based on Columns

Recall that we only want to include records where:

- The 'Question' is 'Current asthma prevalence among adults aged ≥ 18 years'
- The 'StratificationCategoryID1' is 'OVERALL'
- The 'DataValueTypeID' is 'CRDPREV'
- The 'LocationDesc' is not 'United States'

Combining knowledge of the data and metadata, filter out the rows of data that are not relevant.

(You may find the `pandas` library useful here.)

```
# Your code here (add additional cells as needed)
```

You should have 54 records after filtering.

Extract the Attributes Required for Plotting

For each record, the only information we actually need for the graph is the 'DataValue' and 'LocationDesc'. Create a list of records that only contains these two attributes.

Also, make sure that the data values are numbers, not strings.

```
# Your code here (create additional cells as needed)
```

Find Top 10 States

Sort by 'DataValue' and limit to the first 10 records.

```
# Your code here (add additional cells as needed)
```

Separate the Names and Values for Plotting

Assign the names of the top 10 states to a list-like variable `names`, and the associated values to a list-like variable `values`. Then the plotting code below should work correctly to make the desired bar graph.

```
# Replace None with appropriate code
```

```
names = None  
values = None
```

```
# Run this cell without changes
```

```
import matplotlib.pyplot as plt  
fig, ax = plt.subplots()
```

```
ax.barh(names[::-1], values[::-1]) # Values inverted so highest is at top  
ax.set_title('Adult Asthma Rates by State in 2016')  
ax.set_xlabel('Percent 18+ with Asthma');
```

Summary

Well done! In this lab you got some extended practice exploring the structure of JSON files and visualizing data!

Releases

No releases published

Packages

No packages published

Contributors 4



mathymitchell



lmcm18



LoreDirick Lore Dirick



hoffm386 Erin R Hoffman

Languages

● Jupyter Notebook 100.0%