Bring Data In

November 15, 2021

Table of Contents

[ ]:
```
#!pip install sodapy
#!pip install boto3
#!pip install s3fs      # Not used directly but used by pandas 'under the hood'
######## Not required
#!pip install socrata
```

[ ]:
```
from google.colab import drive
drive.mount('/content/drive')
```

1 Reading in a file from github

[ ]:
```
import pandas as pd
diabetes = pd.read_csv('https://bitbucket.org/jimcody/sampledata/raw/
↪b2aa6df015816ec35afc482b53df1b7ca7a31f80/diabetes_for_plotly.csv')
diabetes.head()
```

2 Write to a local disk

[ ]:
```
diabetes2 = diabetes[diabetes.month==2]
diabetes2.to_csv('diabetes2.csv')
```

3 Reading from the database

[ ]:
```
import sqlalchemy
from sqlalchemy.sql import select
from sqlalchemy import create_engine
import pandas as pd
```

[ ]:
```
db_string = 'postgresql://XXXXXXXXX:XXXXXXXXX@diabetes-do-user-10225574-0.b.db.
↪ondigitalocean.com:25060/diabetes'
db = create_engine(db_string)
```

[ ]:
```
result_set = db.execute("SELECT * FROM state")
for r in result_set:
    print(r)
```

[ ]:
```
states = pd.read_sql("""
           select * from state
            """, con = db)
states.head()
```

4 Using an API

[ ]:
```
import pandas as pd
import requests
from sodapy import Socrata
```

[ ]:
```
client = Socrata('data.cdc.gov',
                 '', # AppToken
                 username= '',
                 password='')
results = client.get("7rci-qmm9", limit = 150000)
tss = pd.DataFrame(results)
```

[ ]:
```
tss.head()
```

5 Reading from AWS S3

[ ]:
```
import boto3
import pandas as pd
```

[ ]:
```
# Creating the low level functional client
client = boto3.client(
    's3',
    aws_access_key_id = '',
    aws_secret_access_key= '',
    region_name = 'us-east-2'
```

```
)

# Creating the high level object oriented interface
resource = boto3.resource(
    's3',
    aws_access_key_id = '',
    aws_secret_access_key= '',
    region_name = 'us-east-2'
)
```

[ ]:
```
# Fetch the list of existing buckets
clientResponse = client.list_buckets()

# Print the bucket names one by one
print('Printing bucket names...')
for bucket in clientResponse['Buckets']:
    print(f'Bucket Name: {bucket["Name"]}')
```

[ ]:
```
# Create the S3 object
obj = client.get_object(
    Bucket = 'jcody.class',
    Key = 'outbreaks2.csv'
)

# Read data from the S3 object
outbreaks = pd.read_csv(obj['Body'])

# Print the data frame
#print('Printing the data frame...')
#print(outbreaks)
```

[ ]:
```
outbreaks.head()
```

[ ]:
```
outbreaks.shape
```

[ ]:
```
# Add column names
outbreaks.columns = ['year', 'month', 'state', 'location','food', 'ingredient',␣
 ↪'species', 'serotype','status',
                            'illnesses', 'hospitalizations','fatalities']
```

[ ]:
```
outbreaks.head()
```

[ ]:
```
outbreaks.shape
```

6 Write to S3

[7]:
```
# Having issues here
```

[ ]:
```

```