Thoughts about BigQuery:

I think that for storing data, the methods from most to least convenient are: BitBucket/GitHub, BigQuery, and then data from Google Drive. The reasoning behind this is that BitBucket and GitHub are fairly straightforward. To use BigQuery, it takes a long time to set up credentials and create projects. There's a lot of clicking around to manually create the data (by making a table for the project and then setting it to public). With the pandas API in Jupyter notebook, there's a lot of lines of code that are needed to use the data set. The difficulty with Google Drive data is that the data will have to be public and the user will need to save it to their own drive and allow access to their drive which is a complicated process.

Visualizing BigQuery data in a Jupyter notebook
https://cloud.google.com/bigquery/docs/visualize-jupyter

Downloading BigQuery data to pandas using the BigQuery Storage API
https://cloud.google.com/bigquery/docs/bigquery-storage-python-pandas

TO UPLOAD DATA TO BIGQUERY
Quickstart using the web UI in the Google Cloud Console (to create a dataset and loading the data into a new table; then, make public by going to "share dataset" and adding allUsers or allAuthenticatedUsers as a BigQuery Data Viewer)
https://cloud.google.com/bigquery/docs/quickstarts/quickstart-web-ui

jupyter notebook example:

```
%load_ext google.cloud.bigquery


import google.auth
from google.cloud import bigquery
from google.cloud import bigquery_storage_v1beta1

# Explicitly create a credentials object. This allows you to use the same
# credentials for both the BigQuery and BigQuery Storage clients, avoiding
# unnecessary API calls to fetch duplicate authentication tokens.
credentials, your_project_id = google.auth.default(
    scopes=["https://www.googleapis.com/auth/cloud-platform"]
)

# Make clients.
bqclient = bigquery.Client(
```

```
    credentials=credentials,
    project=your_project_id,
)
bqstorageclient = bigquery_storage_v1beta1.BigQueryStorageClient(
    credentials=credentials
)



# Download query results.
query_string = """
SELECT California, Hawaii
FROM
  `dailycases.dailynewcases`
ORDER BY
  date DESC
LIMIT
  10
"""

dataframe = (
    bqclient.query(query_string)
    .result()
    .to_dataframe(bqstorage_client=bqstorageclient)
)
print(dataframe)
```