Chocolate

1. The best countries to source raw materials
2. Get to know the most important global manufacturers and what type of beans they use in their manufacturing process
3. Establish if there's a correlation between the origin of the bean and the quality of the final product
4. Which markets are more specialized in manufacturing chocolate
5. Explore the relationship between the "purity" of the chocolate and the quality of the final product

Flavors of Cacao Rating System:

- 5= Elite (Transcending beyond the ordinary limits)
- 4= Premium (Superior flavor development, character and style)
- 3= Satisfactory(3.0) to praiseworthy(3.75) (well made with special qualities)
- 2= Disappointing (Passable but contains at least one significant flaw)
- 1= Unpleasant (mostly unpalatable)

Vasudha Nair
Kathryn Buckwalter
Mark Gu

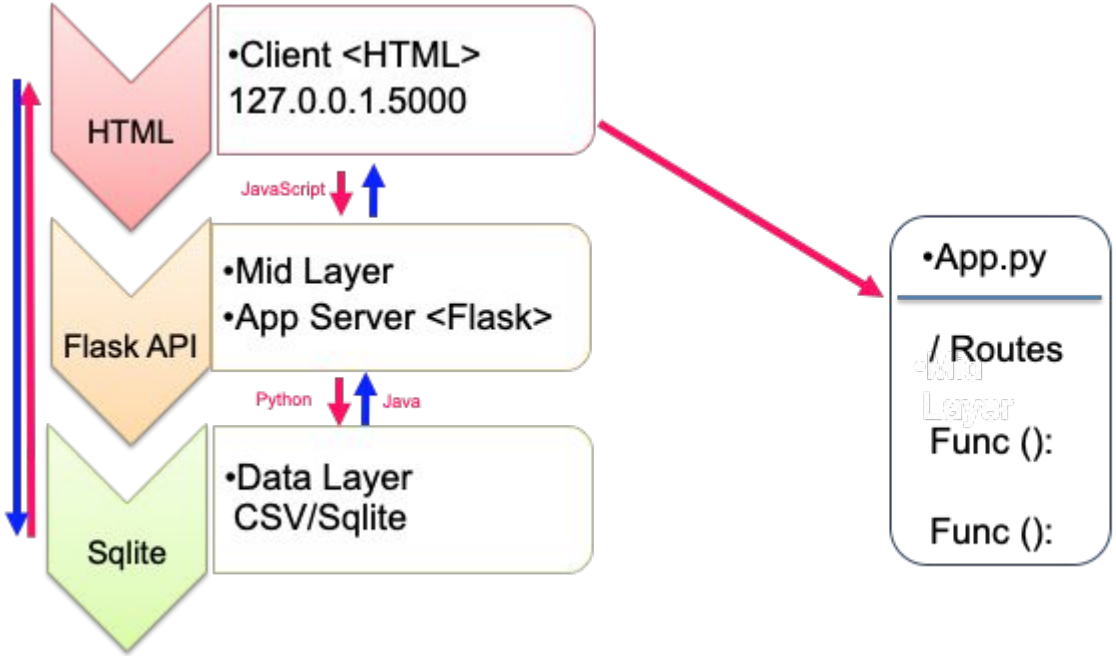
Dependencies , Libraries and Dataset

- Pandas, Numpy, sqlalchemy, sqlite and Flask API
- Javascript- D3, Plotly, Leaflet and RawGraphs
- HTML, CSS

Dataset Chosen: https://www.kaggle.com/rtatman/chocolate-bar-ratings

Data Munging Techniques

Flow chart displaying the broad schema



Coding Approach

Data Cleaning

Steps

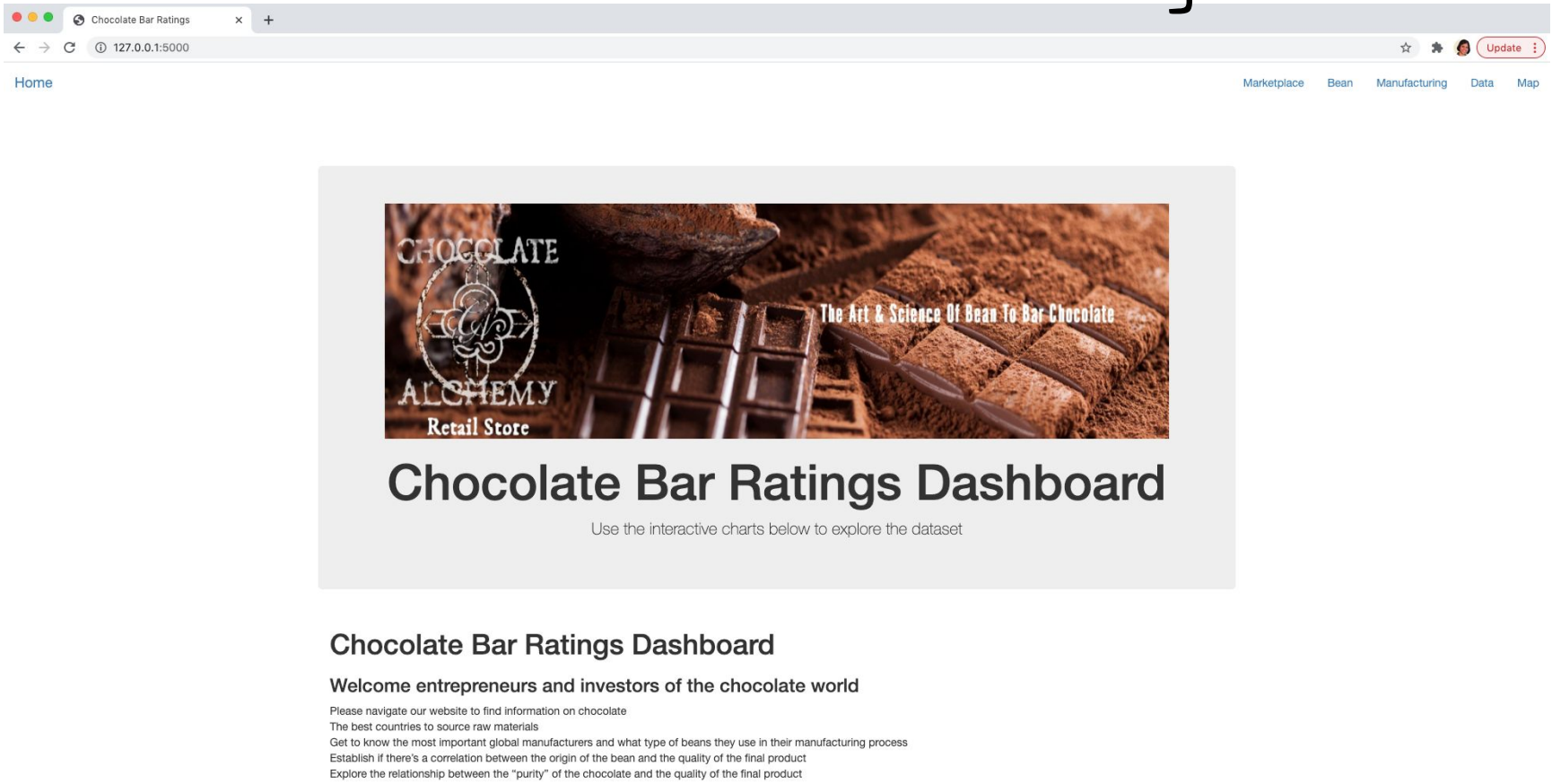
1. Import data from csv
2. Cleaned the data set by selecting columns and dropna
3. Exported cleaned data in to new csv file
4. Exported the cleaned data as an SQlite file

```
# Export file as a CSV, without the Pandas index, but with the header
cacaobean_dropna_df.to_csv("cacao_clean_withbean.csv", index=True, header=True)
```

```
import csv_to_sqlite
# all the usual options are supported
options = csv_to_sqlite.CsvOptions(typing_style="full", encoding="utf-8")
input_files = ["cacao_clean_withbean.csv"] # pass in a list of CSV files
csv_to_sqlite.write_csv(input_files, "cacao_bean.sqlite", options)
```

```
Written 884 rows into 1 tables in 0.206 seconds
884
```

Dashboard-Home or Index Page



Marketplace

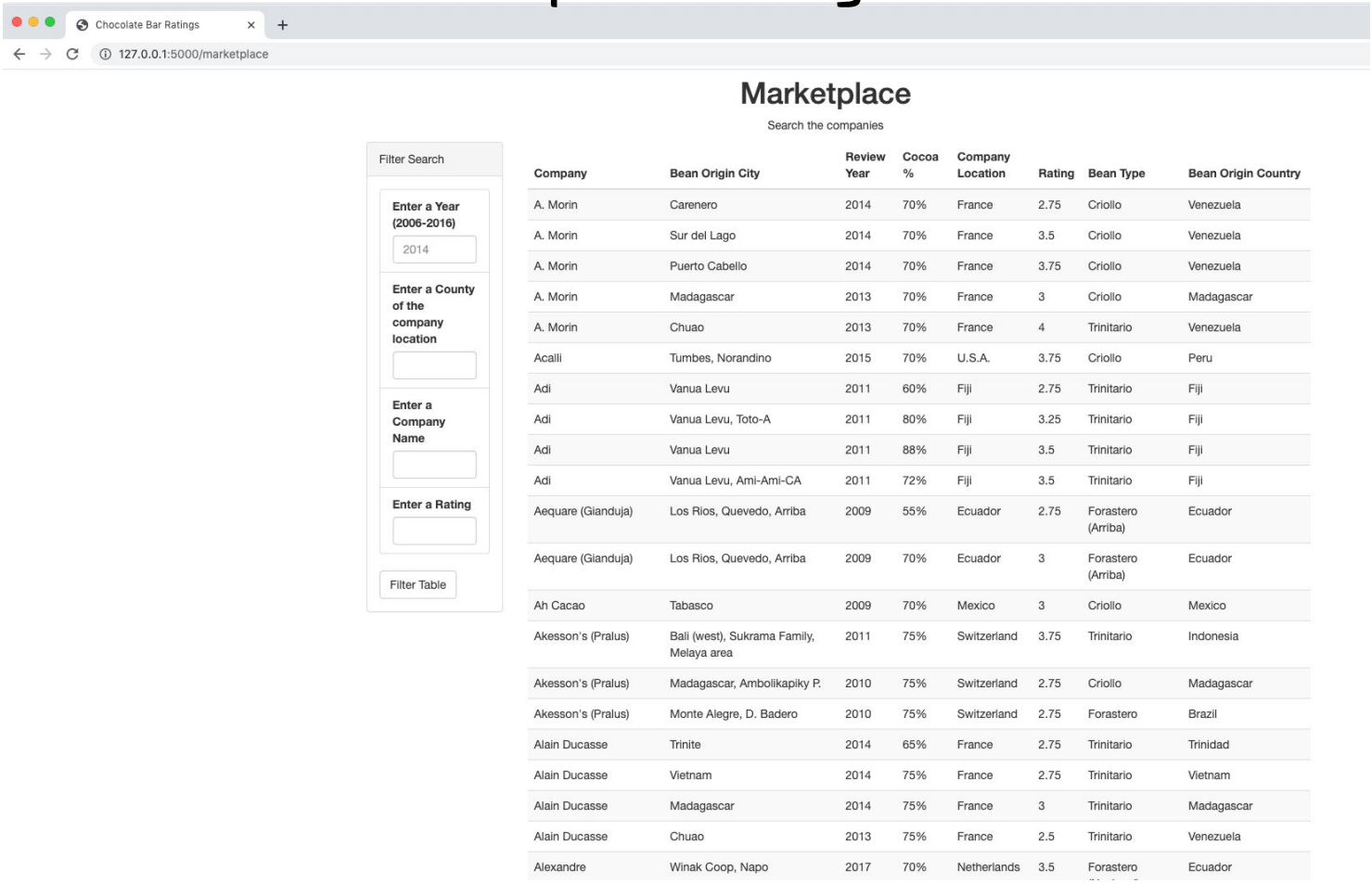
Used JavaScript to create a table with filterable results

Steps

1. Linked a marketplace.html to the index
2. Used D3 to read in data
3. Created a button in JS to filter the data
4. Created a variable to allow the data in the table to reflect the user selects

```
// Prevent the page from refreshing
d3.event.preventDefault();

// Select the input element and get the raw HTML node
    var inputElement = d3.select("#datetime");
    var inputCompanyElement = d3.select("#company");
    var inputCountryElement = d3.select("#country");
    var inputRatingElement = d3.select("#rating");

// Get the value property of the input element
    var inputValue = inputElement.property("value");
    var inputCompanyValue = inputCompanyElement.property("value");
    var inputCountryValue = inputCountryElement.property("value");
    var inputRatingValue = inputRatingElement.property("value");

    //console.log(inputValue);
    //clear the old search content
    tbody.html("")

    var filteredData = tableData.filter(data => data.review_date === inputValue ||
        data.company===inputCompanyValue || data.country===inputCountryValue
        || data.rating===inputRatingValue);
```

Marketplace Page



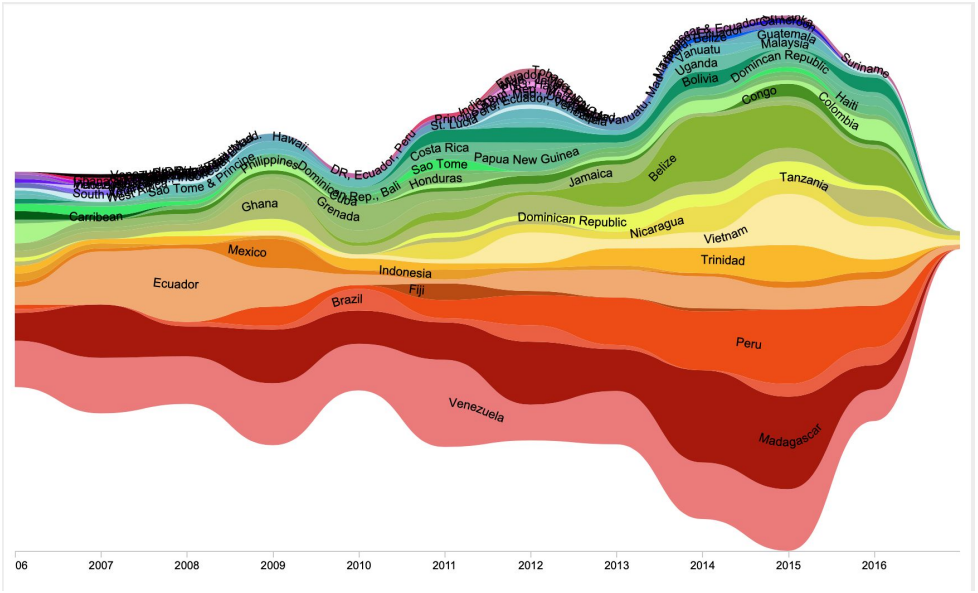
Company	Bean Origin City	Review Year	Cocoa %	Company Location	Rating	Bean Type	Bean Origin Country
A. Morin	Carenero	2014	70%	France	2.75	Criollo	Venezuela
A. Morin	Sur del Lago	2014	70%	France	3.5	Criollo	Venezuela
A. Morin	Puerto Cabello	2014	70%	France	3.75	Criollo	Venezuela
A. Morin	Madagascar	2013	70%	France	3	Criollo	Madagascar
A. Morin	Chuao	2013	70%	France	4	Trinitario	Venezuela
Acalli	Tumbes, Norandino	2015	70%	U.S.A.	3.75	Criollo	Peru
Adi	Vanua Levu	2011	60%	Fiji	2.75	Trinitario	Fiji
Adi	Vanua Levu, Toto-A	2011	80%	Fiji	3.25	Trinitario	Fiji
Adi	Vanua Levu	2011	88%	Fiji	3.5	Trinitario	Fiji
Adi	Vanua Levu, Ami-Ami-CA	2011	72%	Fiji	3.5	Trinitario	Fiji
Aequare (Gianduja)	Los Rios, Quevedo, Arriba	2009	55%	Ecuador	2.75	Forastero (Arriba)	Ecuador
Aequare (Gianduja)	Los Rios, Quevedo, Arriba	2009	70%	Ecuador	3	Forastero (Arriba)	Ecuador
Ah Cacao	Tabasco	2009	70%	Mexico	3	Criollo	Mexico
Akesson's (Pralus)	Bali (west), Sukrama Family, Melaya area	2011	75%	Switzerland	3.75	Trinitario	Indonesia
Akesson's (Pralus)	Madagascar, Ambolikapiky P.	2010	75%	Switzerland	2.75	Criollo	Madagascar
Akesson's (Pralus)	Monte Alegre, D. Badero	2010	75%	Switzerland	2.75	Forastero	Brazil
Alain Ducasse	Trinite	2014	65%	France	2.75	Trinitario	Trinidad
Alain Ducasse	Vietnam	2014	75%	France	2.75	Trinitario	Vietnam
Alain Ducasse	Madagascar	2014	75%	France	3	Trinitario	Madagascar
Alain Ducasse	Chuao	2013	75%	France	2.5	Trinitario	Venezuela
Alexandre	Winak Coop, Napo	2017	70%	Netherlands	3.5	Forastero	Ecuador

Beans

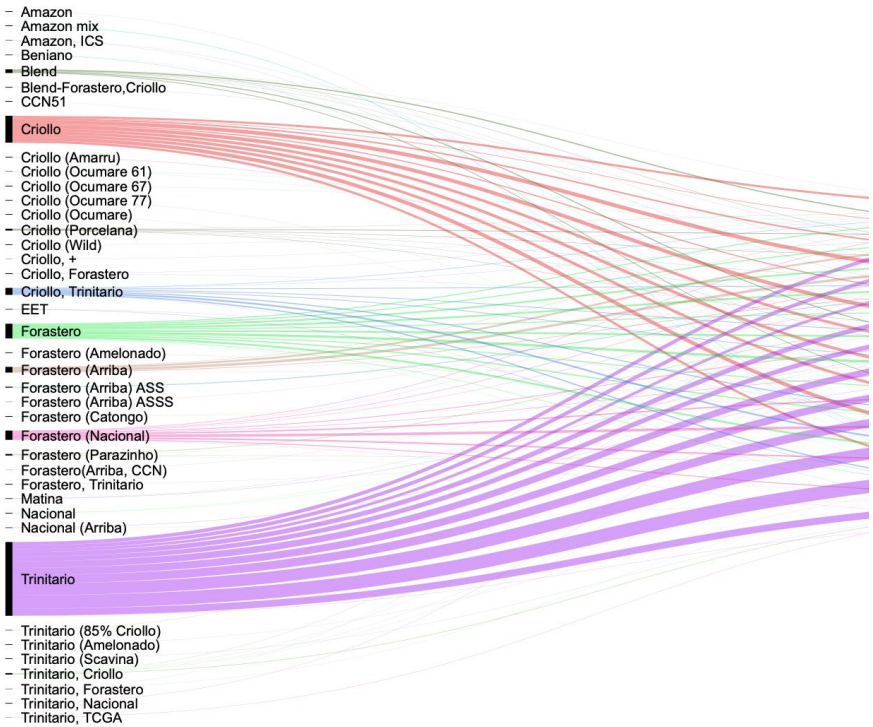
Used new JS library, RAW, to create graphs

RAW Graphs is an open source data visualization framework built with the goal of making the visual representation of complex data easy for everyone.

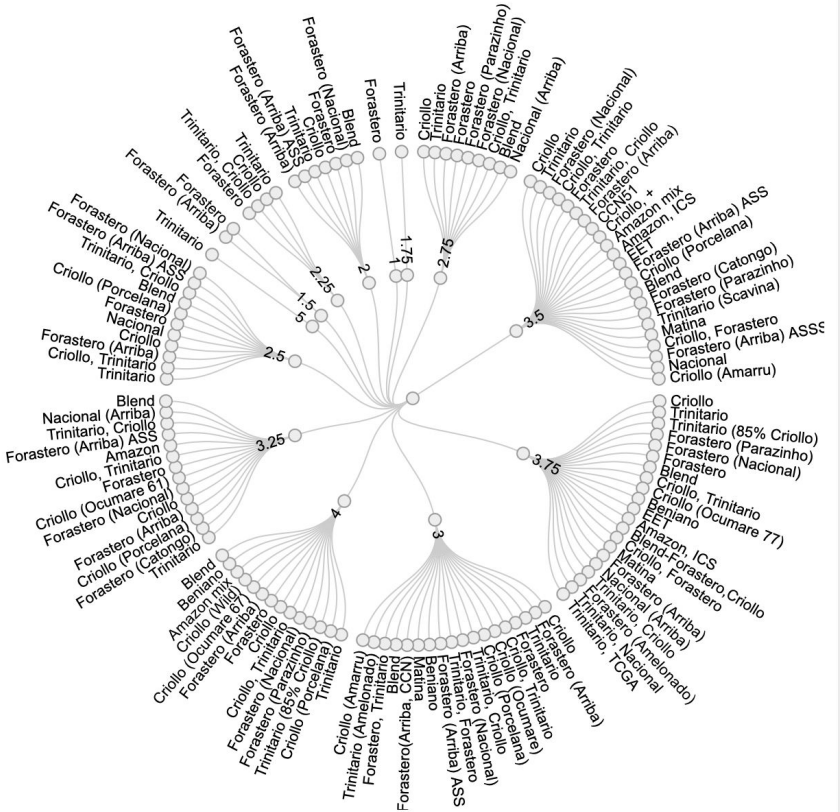
1. Used the graphs to represent how the bean country of origins rated over the rating time period.



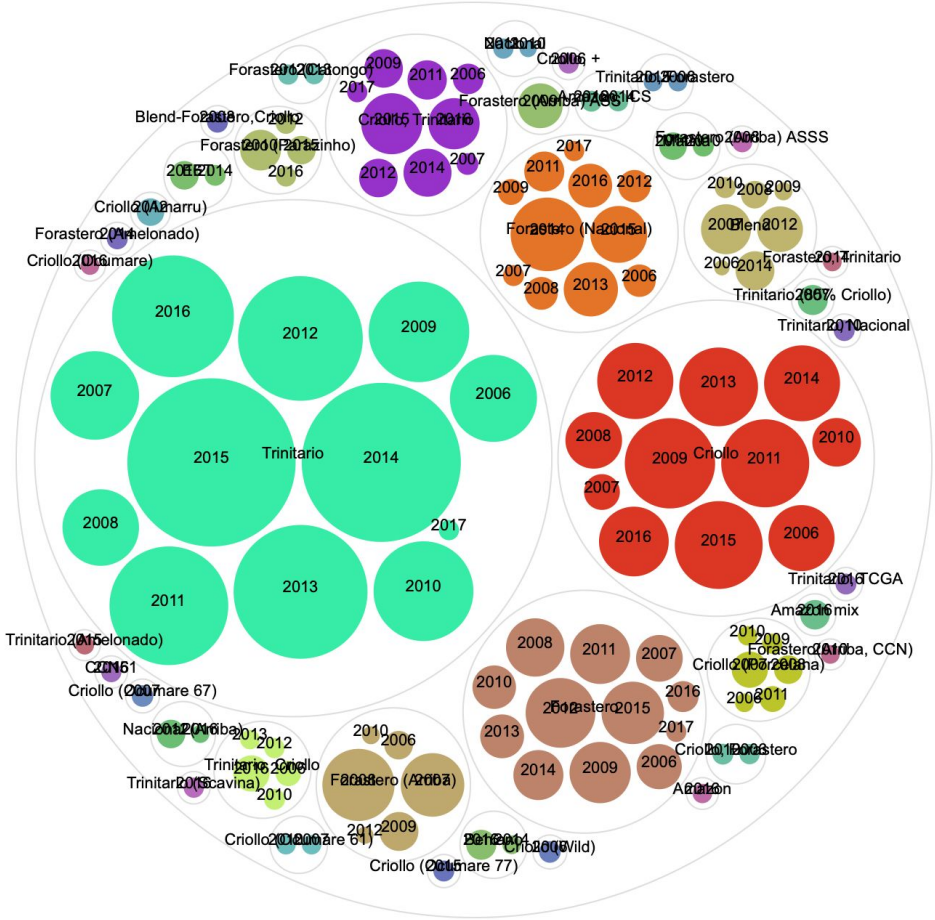
Alluvial Year and Bean Type with rating as size of line



Dendrogram of Ratings by Bean Type



Circle Cluster of Rating by Bean Type



Manufacturing

Used JavaScript, Python and Plotly to read in data and create interactive Bubble plot

Steps

1. Linked a manufacturing.html to the index
2. Used D3 to read in data
3. Used flask to render the route and Jsonify the data
4. Created a variable to using or to allow the data in the table to reflect the user selects

```python
@app.route("/manufacturing/<location>")
def manufacturing(location):
    """Return the MetaData for a given location."""
    # results = cur.execute(f"SELECT * FROM cacao_clean_withbean WHERE company_location='{location}'").fetcha
    # Create a dictionary entry for each row of metadata information
    cacao_data = jsonify(cacao_table)
    return cacao_data
```

```javascript
d3.json("/data").then((data) => {
  console.log(data);
  var companyLocation = [];

  data.forEach((point) => {
    if (companyLocation.indexOf(point[5]) > -1) {
      // do nothing
    } else {
      companyLocation.push(point[5]);
    };
  });
  companyLocation.forEach((location) => {
    dropdown.append("option").text(location).property("value", location);
  });
  console.log(companyLocation);
  // Use the First Sample from the List to Build Initial Plots

  buildCharts(companyLocation[0]);
  getData(companyLocation[0]);
})
}

function optionChanged(new_location) {
  // Fetch New Data Each Time a New Sample is Selected
  buildCharts(new_location);
  getData(new_location);
}

// Initialize the Dashboard
init();
```

127.0.0.1:5000/manufacturing

Retail Store

Manufacturing Dashboard

Use the interactive charts below to explore the dataset

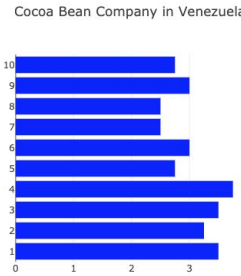
Company Country Location

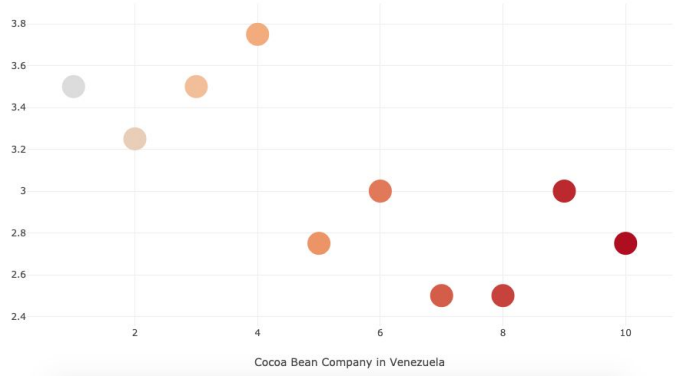
Venezuela

Cocoa Bean Info

Comany Location : Venezuela

Average Rating : 3.05





Data

Steps

1. Used jupyter notebook to convert csv to html
2. Linked the data.html to the index.html to display our data.

```
import pandas as pd
```

```
data_df=pd.read_csv("data/cacao_clean_withbean.csv")
data_df.head()
```

	company	specific_bean_origin	review_date	cocoa_percent	company_location	rating	bean_type	bean_origin_country
0	A. Morin	Carenero	2014	70%	France	2.75	Criollo	Venezuela
1	A. Morin	Sur del Lago	2014	70%	France	3.50	Criollo	Venezuela
2	A. Morin	Puerto Cabello	2014	70%	France	3.75	Criollo	Venezuela
3	A. Morin	Madagascar	2013	70%	France	3.00	Criollo	Madagascar
4	A. Morin	Chuao	2013	70%	France	4.00	Trinitario	Venezuela

```
html=data_df.to_html()
html
```

Map

Steps

1. Used Leaflet to create a map displaying the country and the average chocolate ratings
2. Linked the map.html to the index.html to display our data.