

---

Read chapter 4 in [Python Data Science Handbook](#) and complete the following exercises. In this assignment, you will use the `pandas` and `matplotlib` package to visualize the cleaned produce data.

Each part will define what it takes to earn a ‘B’ with suggestions for ways to improve it to earn a higher grade. At the top of each Python file, include your name and a list of the extra features you implemented.

**Complete computer setup by following the directions here:**

<http://cs.appstate.edu/~rmp/cs5245/setup.pdf>

Complete the following activities using the **exact function names** and save the files using the **exact file names** as specified. Test your programs using the command line or an IDE to check their output.

## 1 Description

In this assignment, you will use `pandas` and `matplotlib` to visualize a cleaned version of the produce data. You will create two programs. Each will start with a main function like and you will add code to accomplish the various tasks, saving a PNG file as your result.

```
def main():  
    print('it works!')  
  
if __name__ == '__main__':  
    main()
```

Download the following and put into the directory where you will work on this assignment.

1. [food\\_cleaned.csv](#)

## 2 Constructing a Single Bar Graph: `bar.py`

This program plots one bar graph containing the aggregated units sold for each month in 2019. To earn a ‘B’:

1. Prompt the user for the desired category
2. Only use sales from 2019
3. Aggregate the sales for each month
4. Plot a bar graph showing total units sold each month
5. Months should be ordered January to December on the  $x$ -axis
6. Label the  $x$ -axis and  $y$ -axis
7. Provide a title that includes the units and selected subcategory

To earn additional credit:

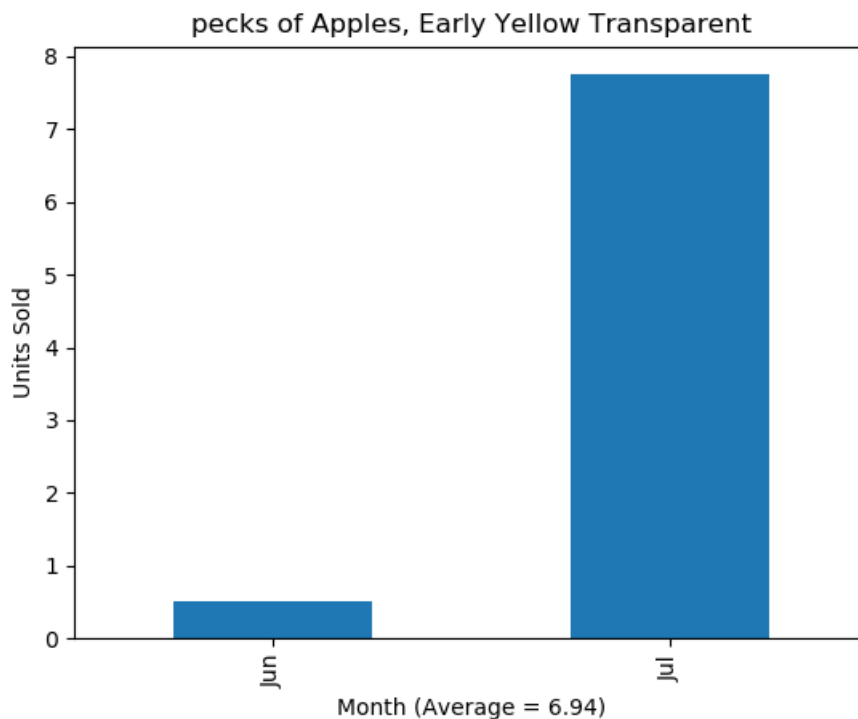
1. Compute the expected month of sale for a unit of the selected produce and display it in the figure (see below).
2. Use the names of the months on the  $x$ -axis
3. Implement other additional features for credit.

The **expected month of purchase** is the average month of each sale weighted by the number of units sold. For example, if 2 units were sold in March and 3 units in April, the weighted average for the month would be:

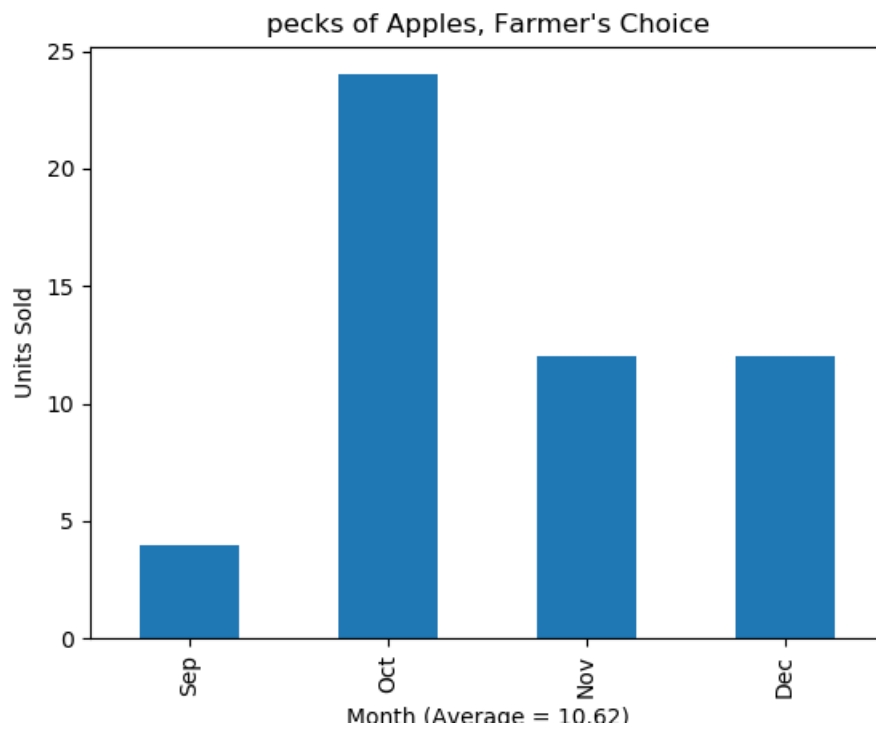
$$\frac{(2 \text{ units}) \times (\text{March} = 3) + (3 \text{ units}) \times (\text{April} = 4)}{2 \text{ units} + 3 \text{ units} = 5 \text{ total units}} = \frac{6 + 12}{5} = 3.6 \text{ (closer to April than March)}$$

For example, it should look something like this:

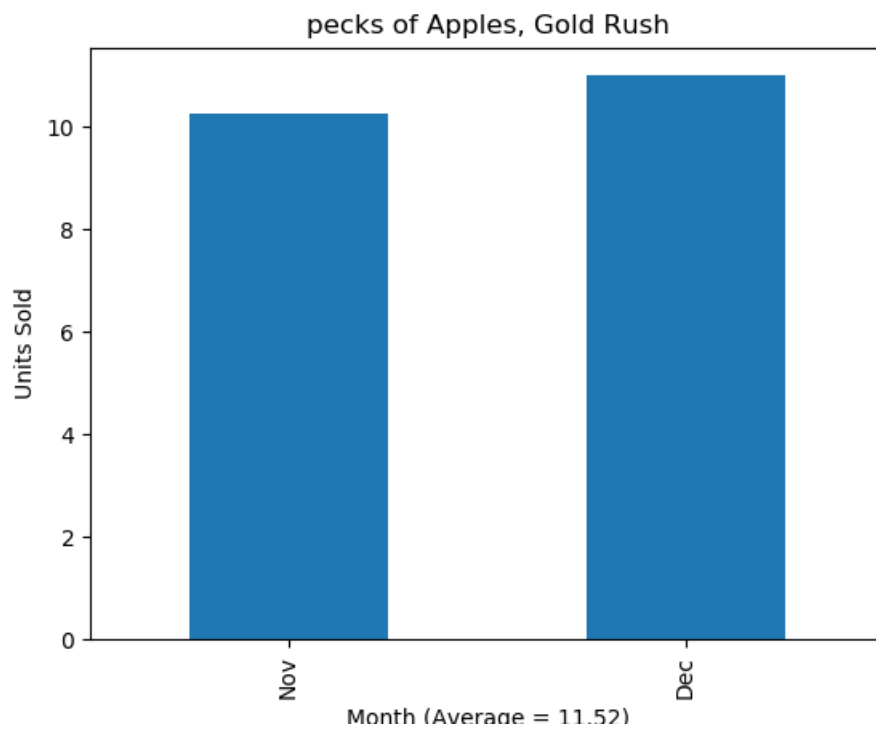
```
$ python bar.py
Enter SubCategory: Apples, Early Yellow Transparent
```



```
$ python bar.py
Enter SubCategory: Apples, Farmer's Choice
```



```
$ python bar.py  
Enter SubCategory: Apples, Gold Rush
```



### 3 Constructing Multiple Bar Graphs: `bars.py`

This program stacks multiple normalized bar graphs on top of each other to show relative distribution of purchase dates. To earn a ‘B’ it must implement the following features:

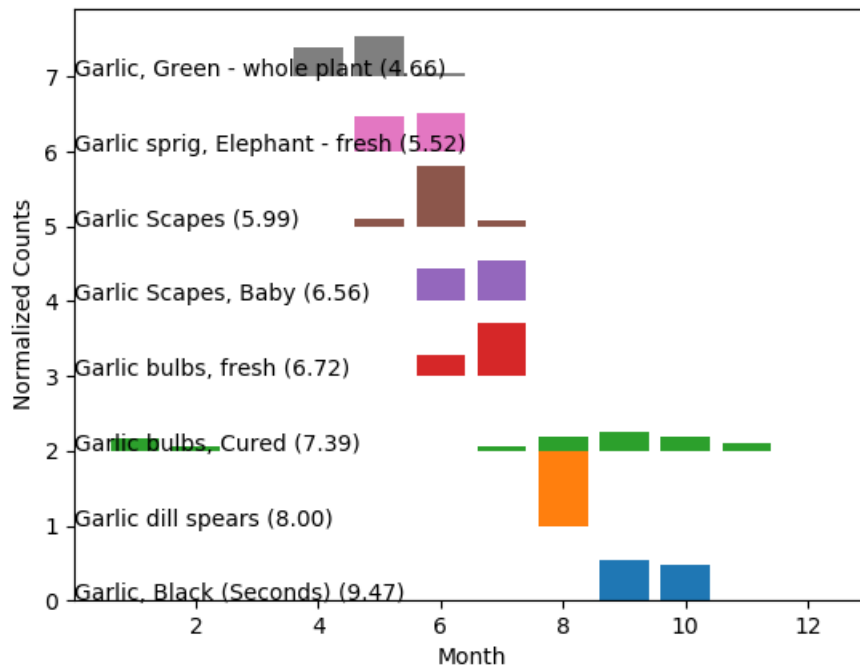
1. Prompt the user for a “starts with” string to indicate which subcategories to include
2. Only include sales from 2019
3. Only include bar graphs for subcategories that start with the “starts with” string
4. Aggregate sales for each month
5. Plot a bar graph for each subcategory in the same axes
6. Months should be ordered January to December on the  $x$ -axis
7. Label the  $x$ -axis and  $y$ -axis
8. Label each subcategory

To earn additional credit:

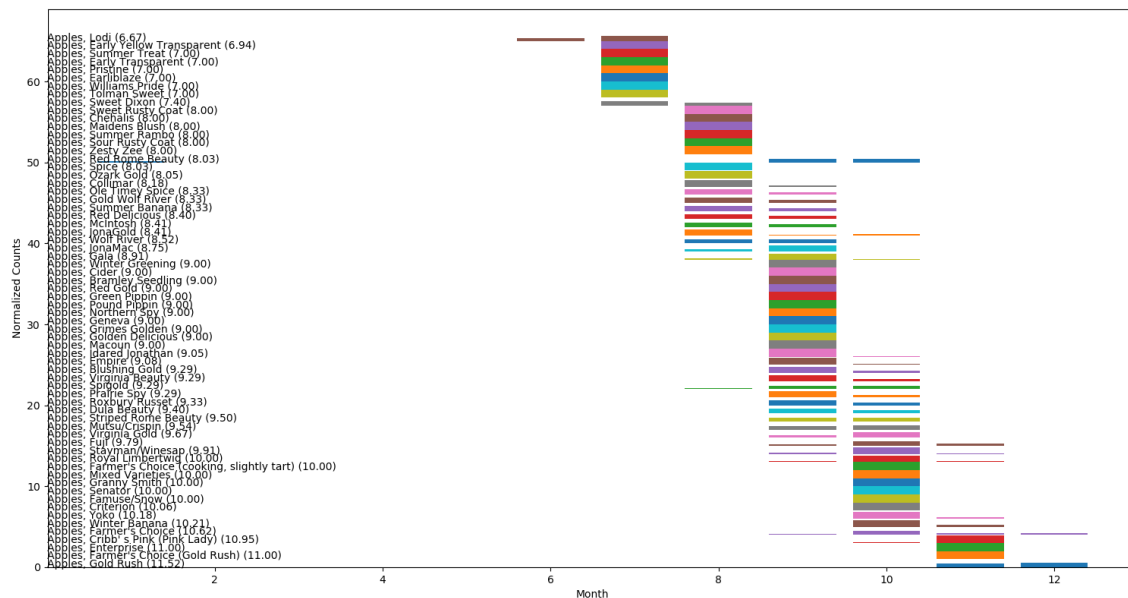
1. Show the average purchase month for each subcategory
2. Sort the subcategories by average purchase month
3. Do a better job of text placement so that it doesn’t overlap the bars
4. Implement other additional features for credit.

For example, it might look like this:

```
$ python bars.py
Keep subcategories that start with: Garlic
```



\$ python bars.py  
Keep subcategories that start with: Apples



## 4 Submit to Asulearn

1. Include a comment at the top of each file with your name, a list of resources that you used, and any additional features you implemented.

2. Upload the python files `bar.py` and `bars.py` in a ZIP file
3. Login to <http://asulearn.appstate.edu> and submit a ZIP file with your `bar.py` and `bars.py` file for grading.