**PROJECT-**

**1.Market Basket Analysis-** Using Apriori/Association Rules on dataset Online Retail for analyzing the frequent purchase of items and which items was purchased more.

**2. Consumer Behavior Analysis-** Using K-Means Clustering (Unsupervised Learning)on dataset Mall Customer for analyzing the behavior of customers.

**3. HR Analytics-**

1. **Market Basket Analysis-**

Have you ever asked yourself how the store managers decide on product shelf placement in retail stores? There must be some strategy behind it, right? It can’t be just a random choice. Almost on daily basis, you receive product purchase recommendations from variety of sources where you have left your “digital fingerprint”. In many cases these recommendations make sense, what leaves you puzzled, how did they figured it out?

The Market Basket Analysis is perhaps the most famous method in Association Mining techniques arsenal. It’s all about finding frequent pairs, triples, quadruples of products from historical transactions or market baskets.

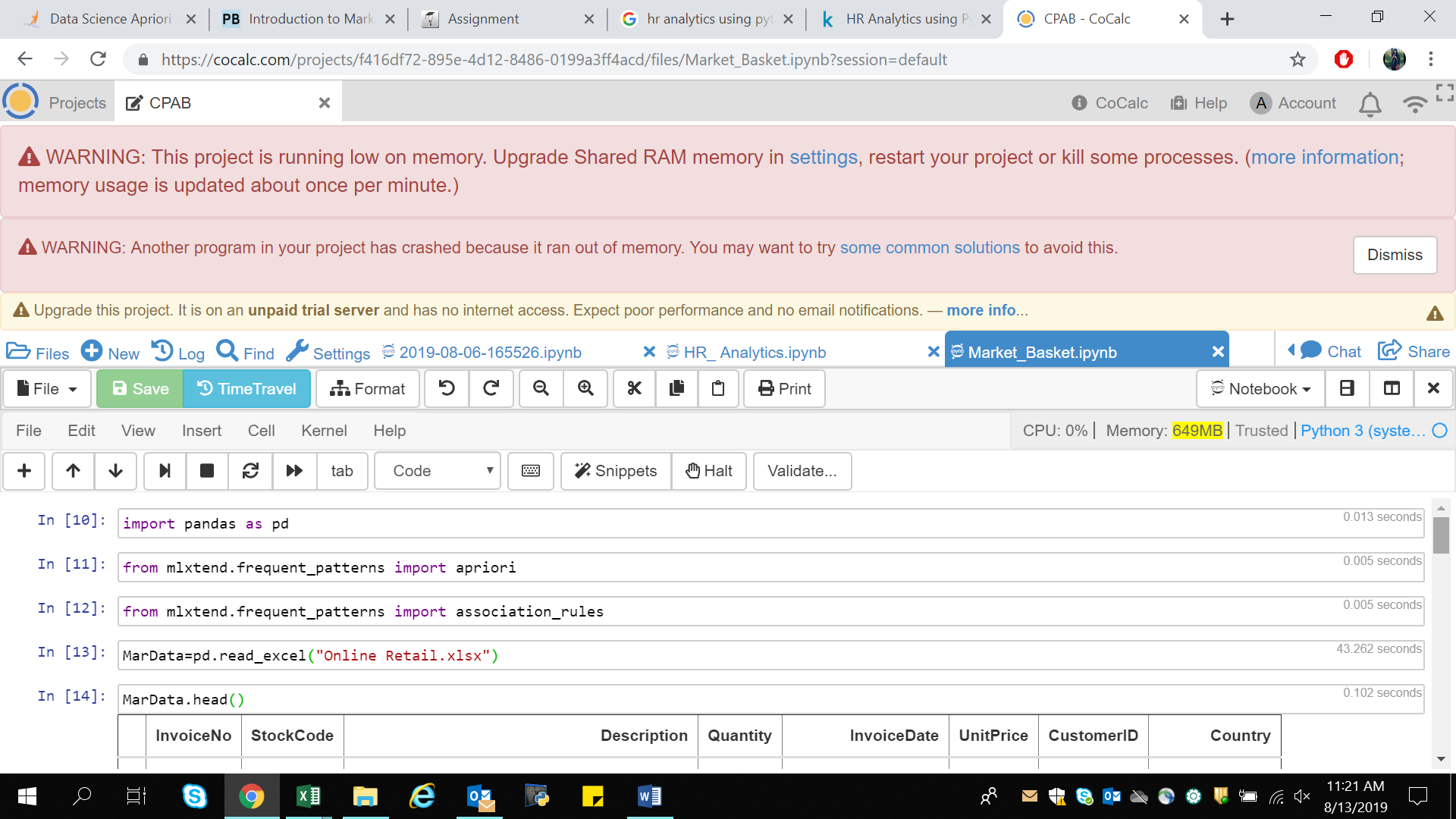
Here I have Used dataset of ONLINE RETAIL and perform Market Basket Analysis.

About the Dataset-

It is a transactional data set which contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK-based and registered non-store online retail. The company mainly sells unique all-occasion gifts; many customers of the company are wholesalers.

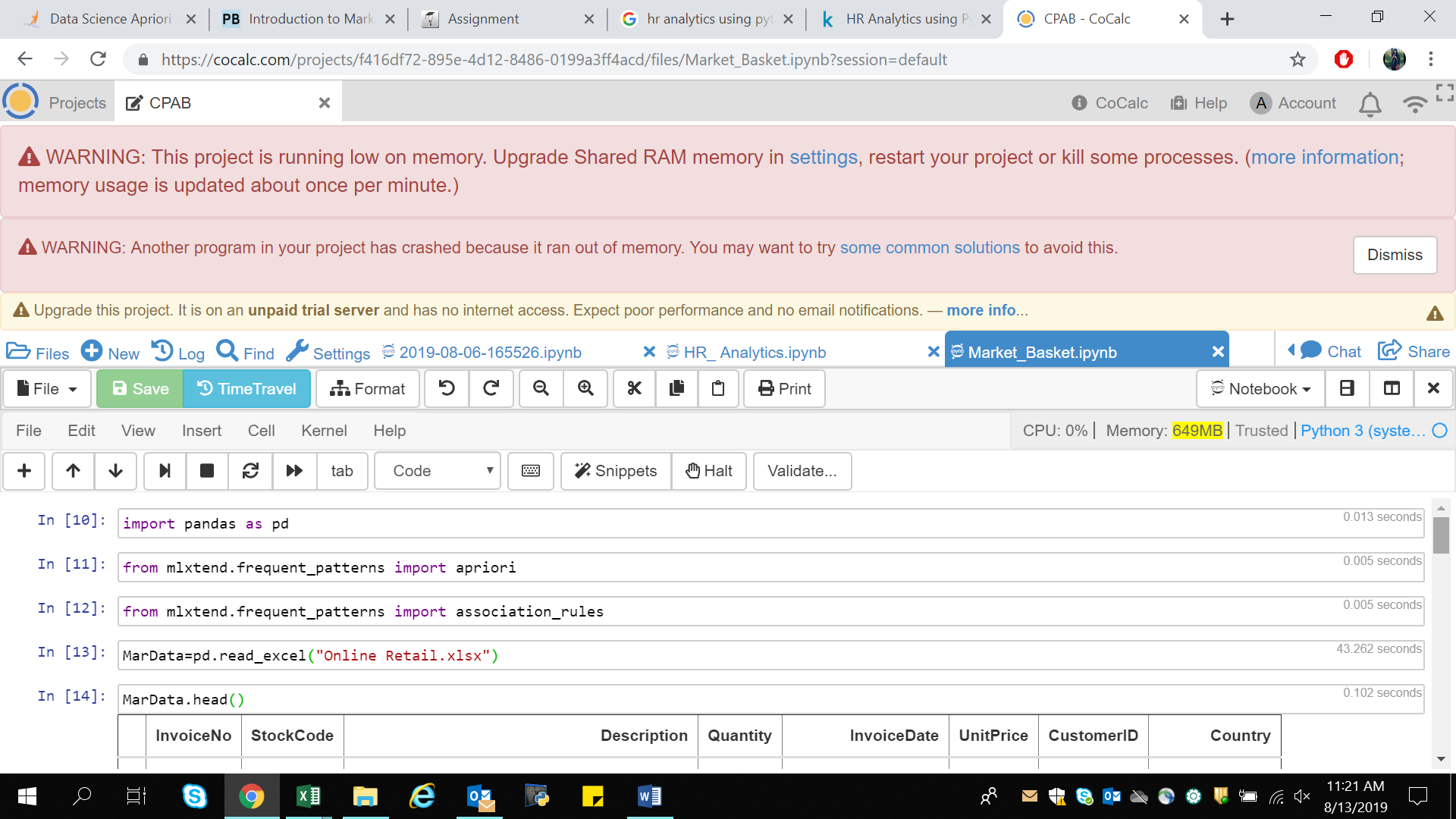
My ultimate goal is to do Market Basket Analysis on this data and figure out the association rules.

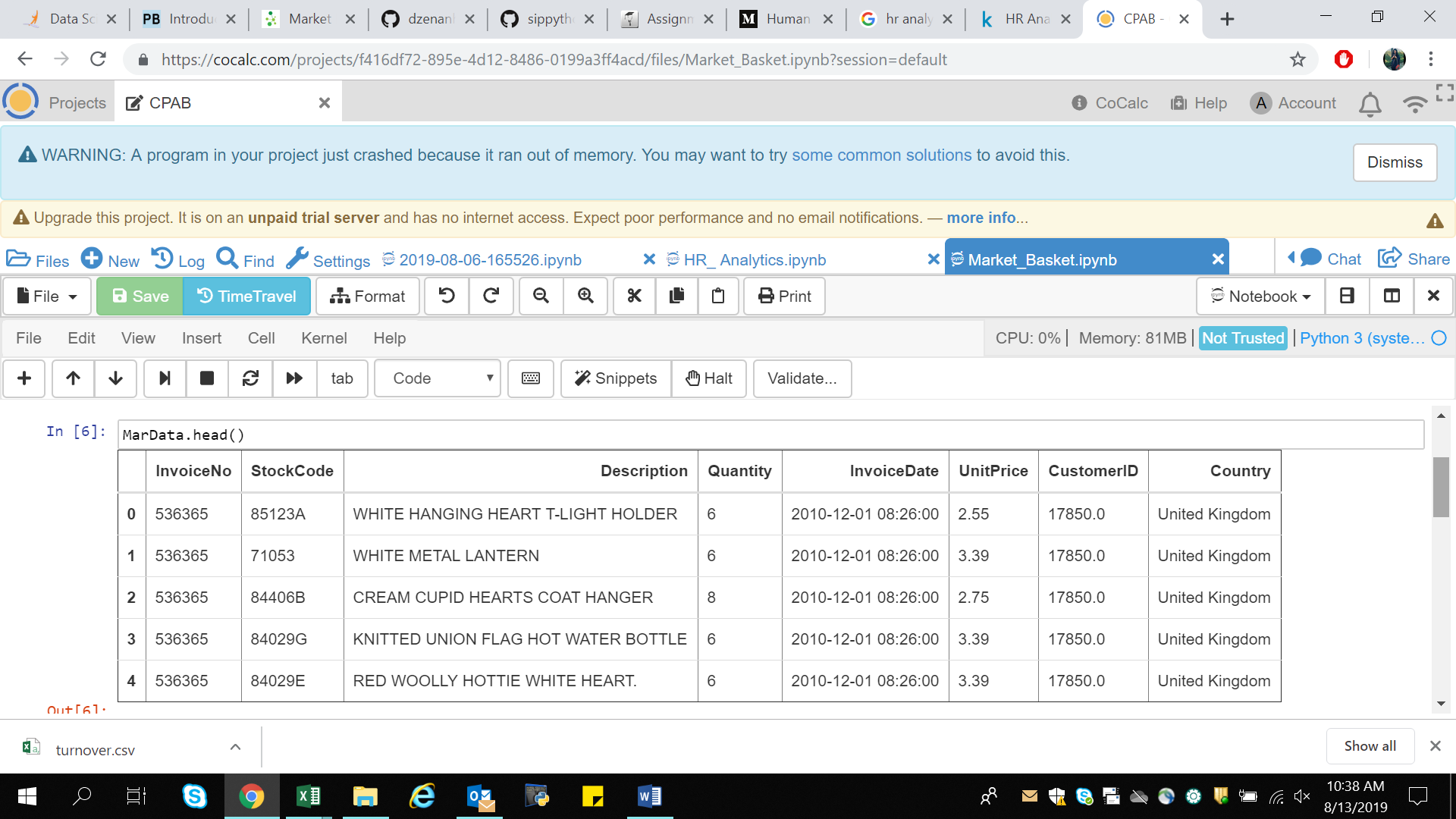
Library Used-



1. Pandas- for data manipulation and data analysis.
2. Apriori- for market basket analysis.
3. Association\_rules- for market basket analysis

Data is imported using pandas library under name MarData.

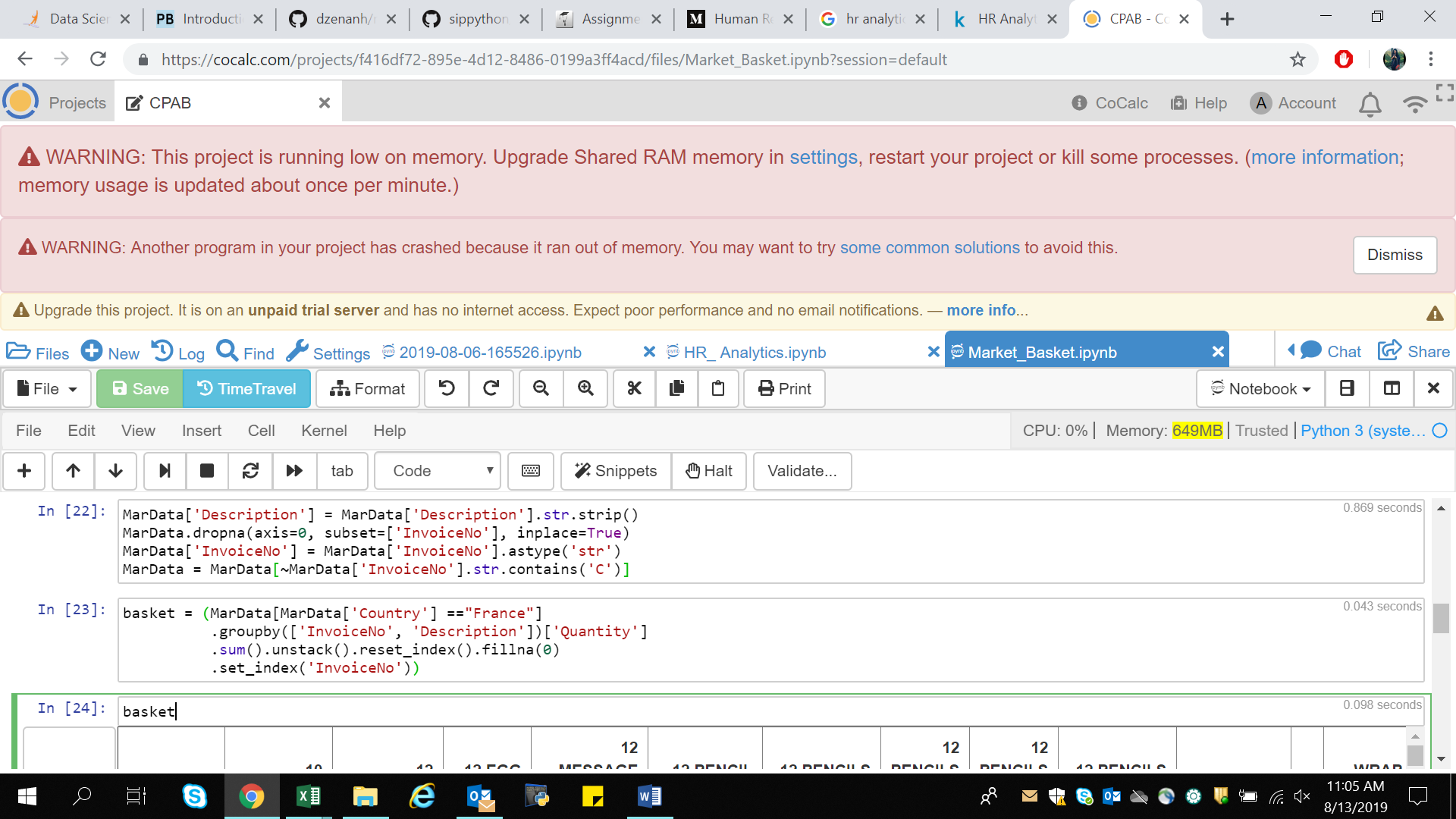




The above picture shows the data contained by the dataset.

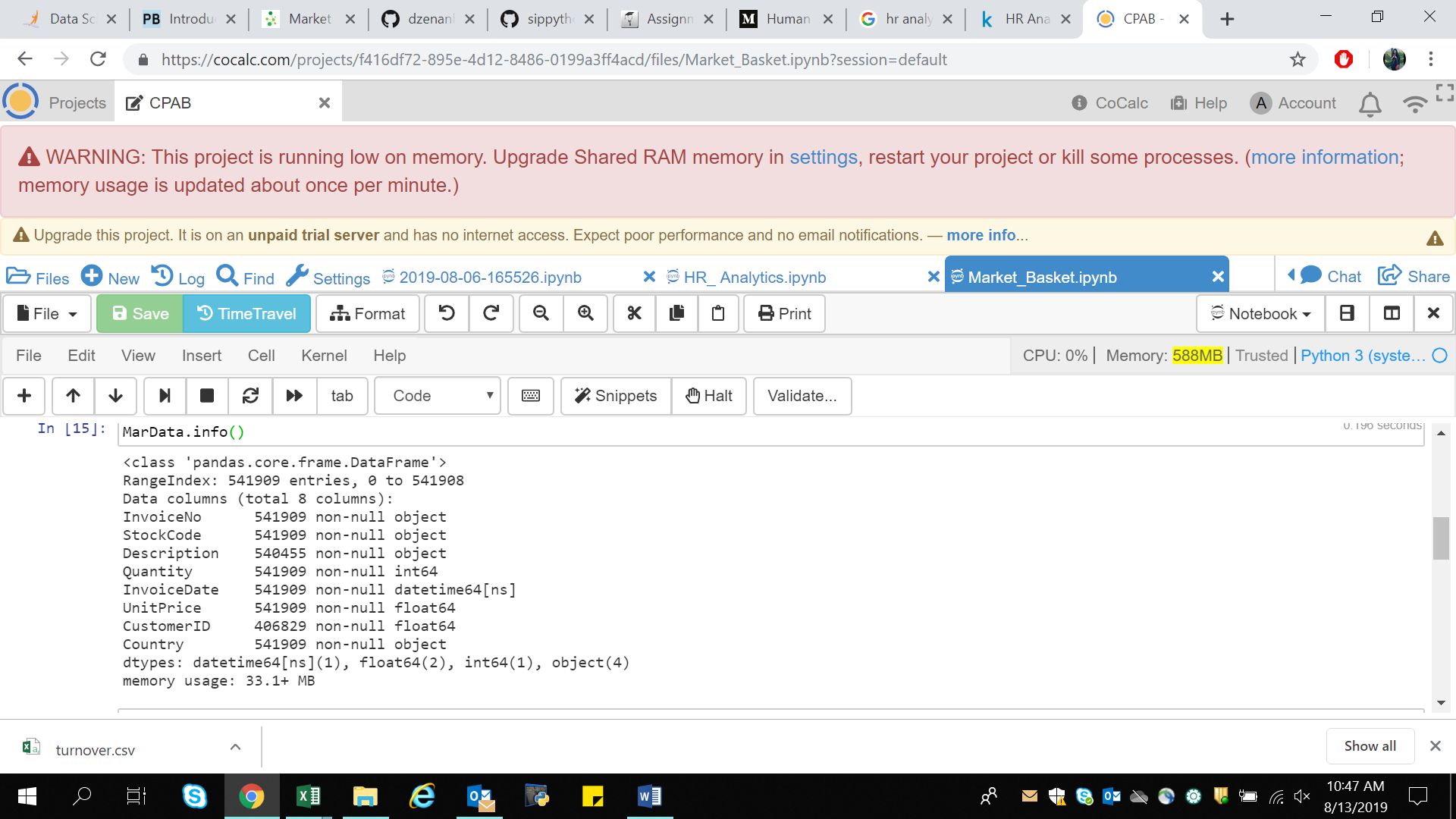
Data Cleaning-

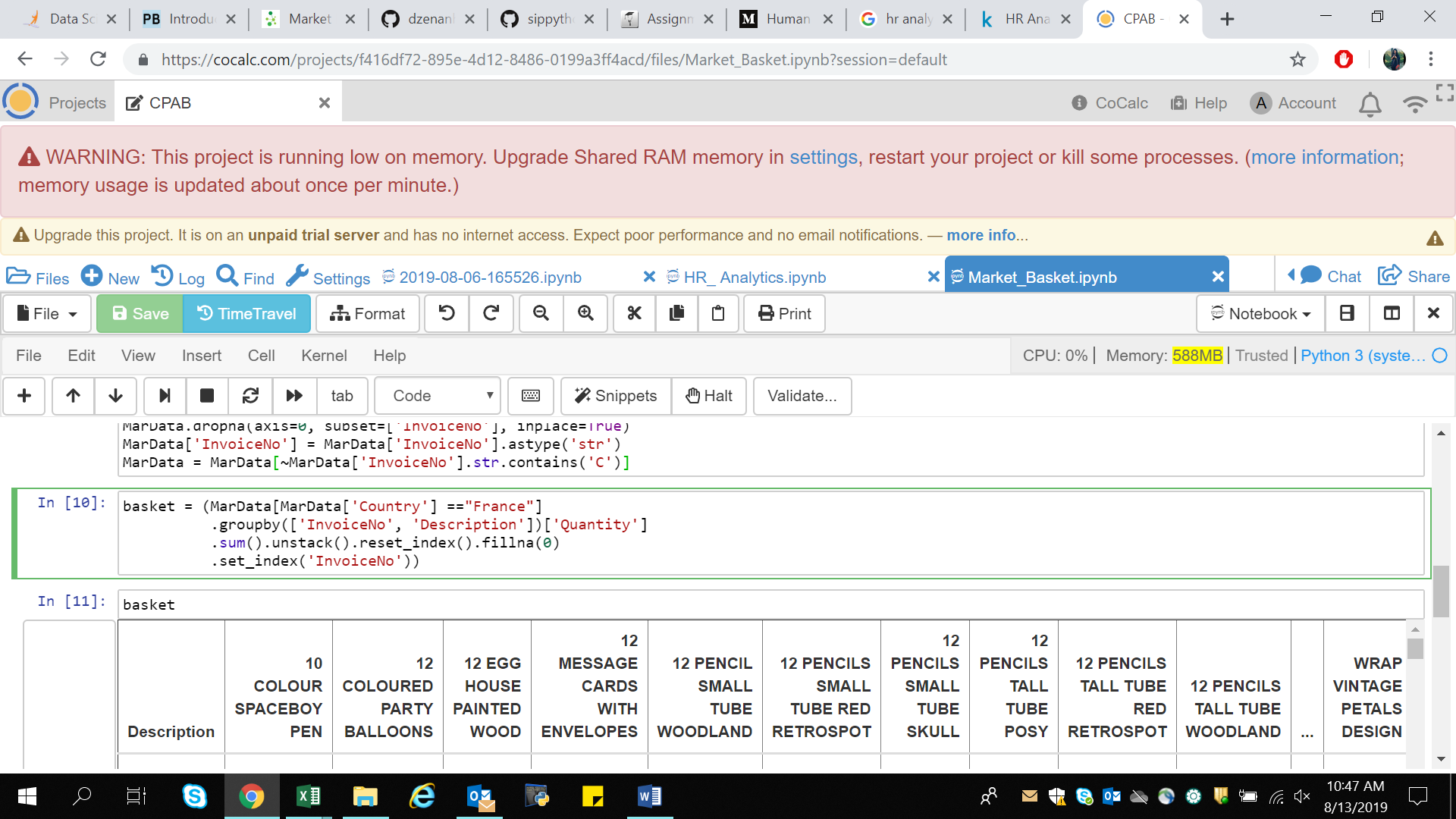
To check data contain any white spaces or Null values can be viewed using dataobject.info() functions.

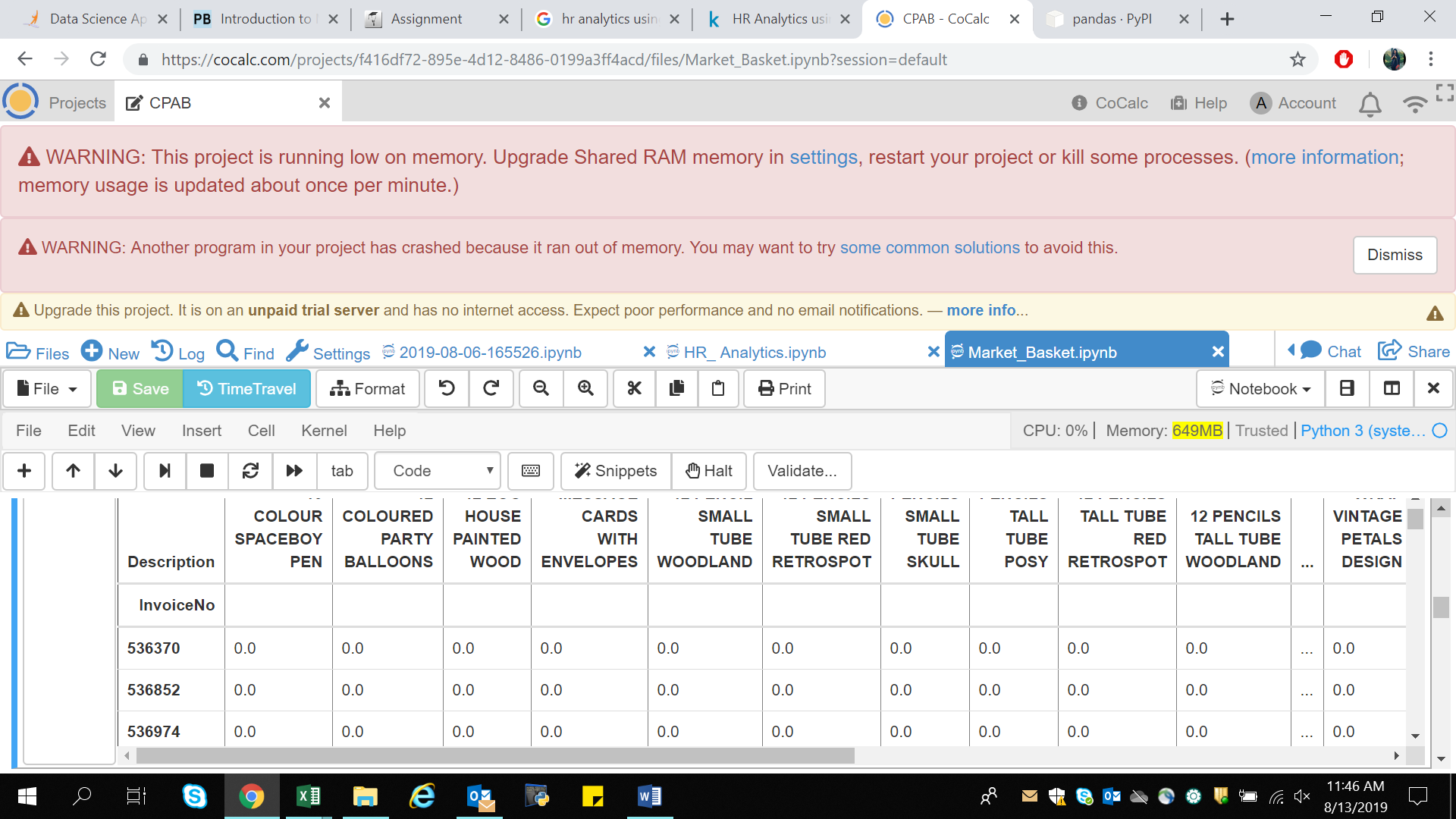


1. str.strip - to remove Leading and trailing spaces.
2. dropna - to remove null values.
3. astype - to cast entire pandas object to the same type.
4. ~MarData[‘InvoiceNo’].str.contains(‘C’) – to remove credit transaction from InvoiceNo

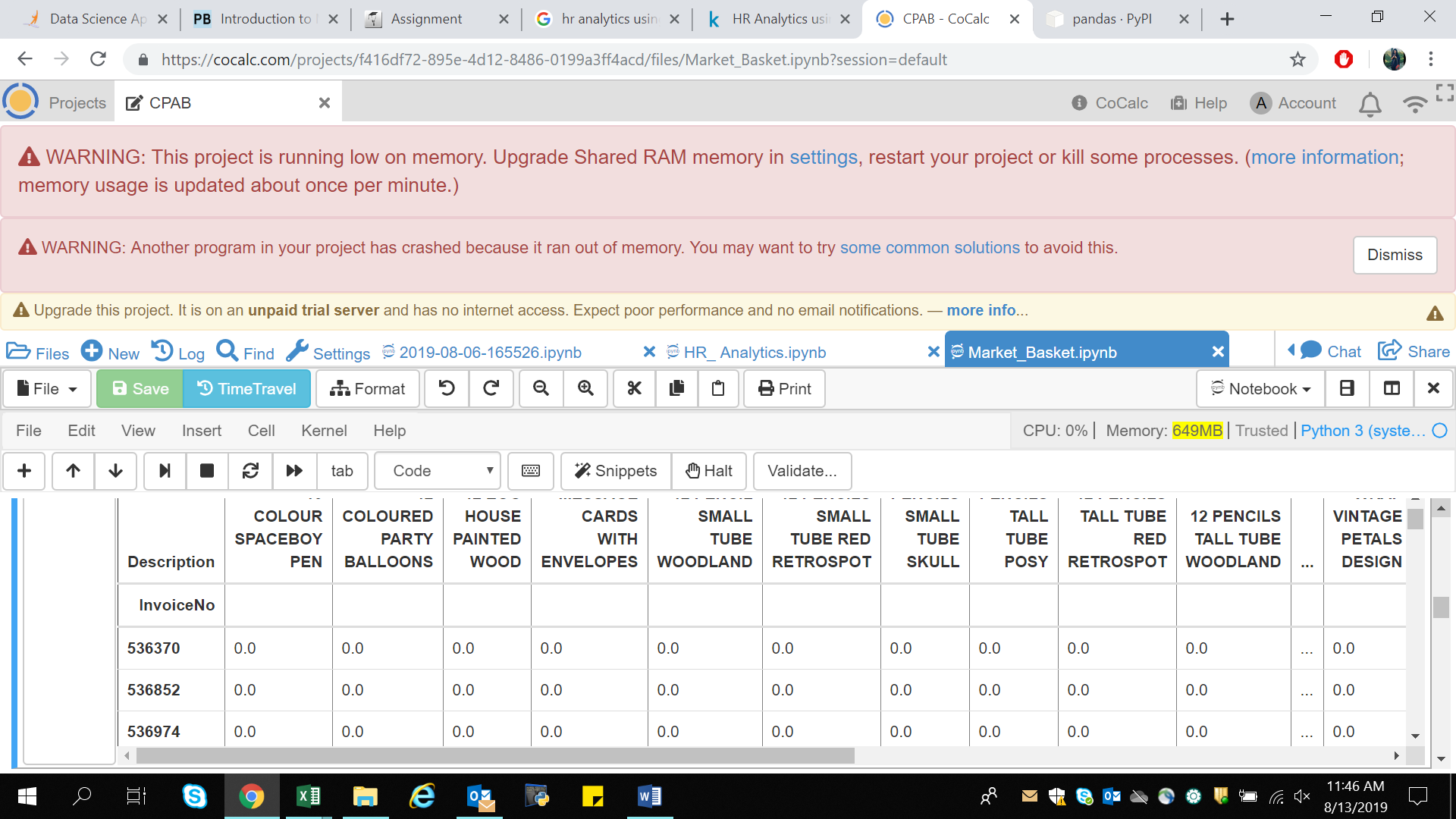
After data cleaning the dataset looks like in the following format.



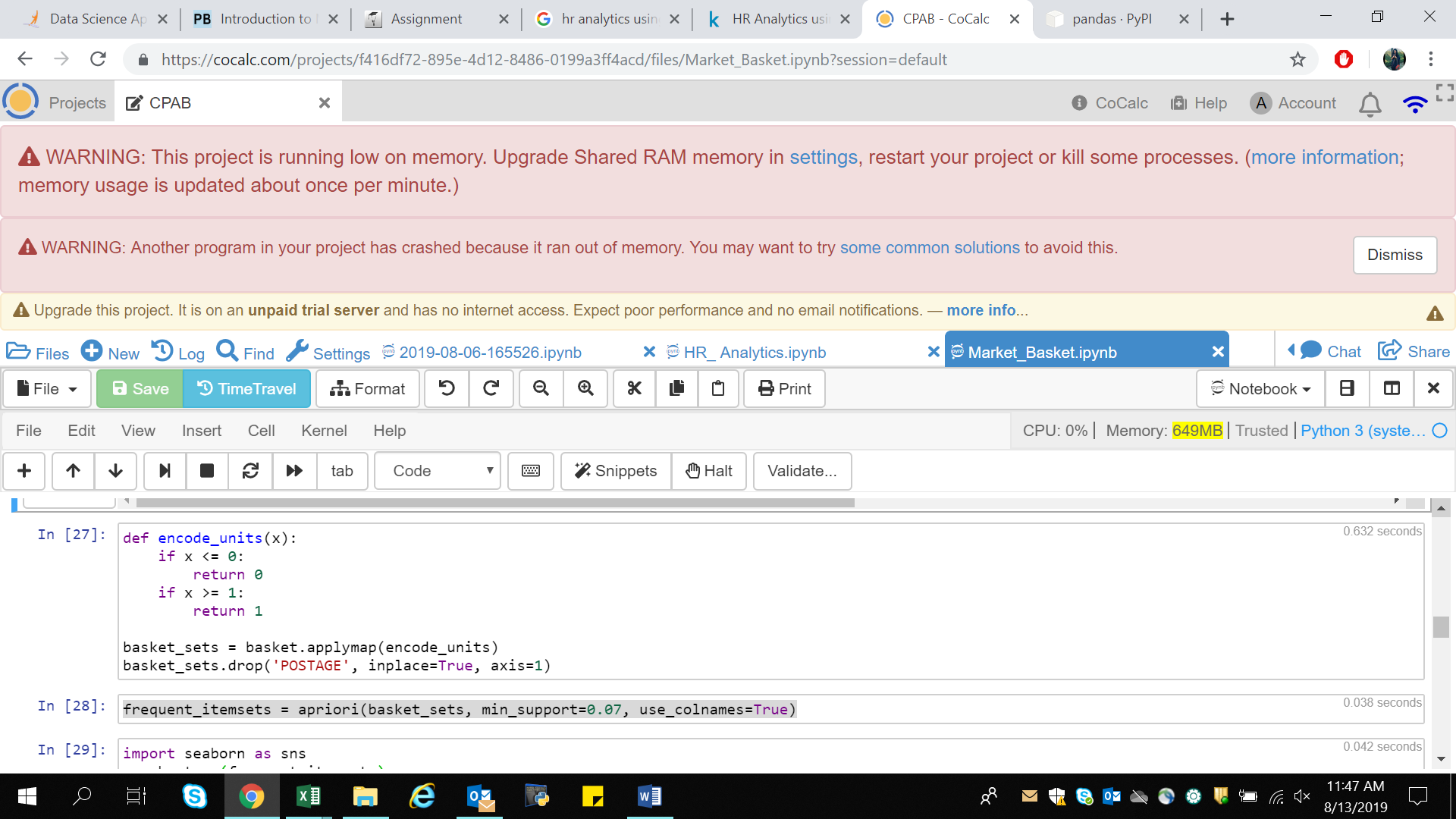
After Data cleaning to consolidate the items into 1 transaction per row with each product. Here, I will be performing analysis for the sales in France (similarly can do for other Country just replacing the name of the country with other say United Kingdom).



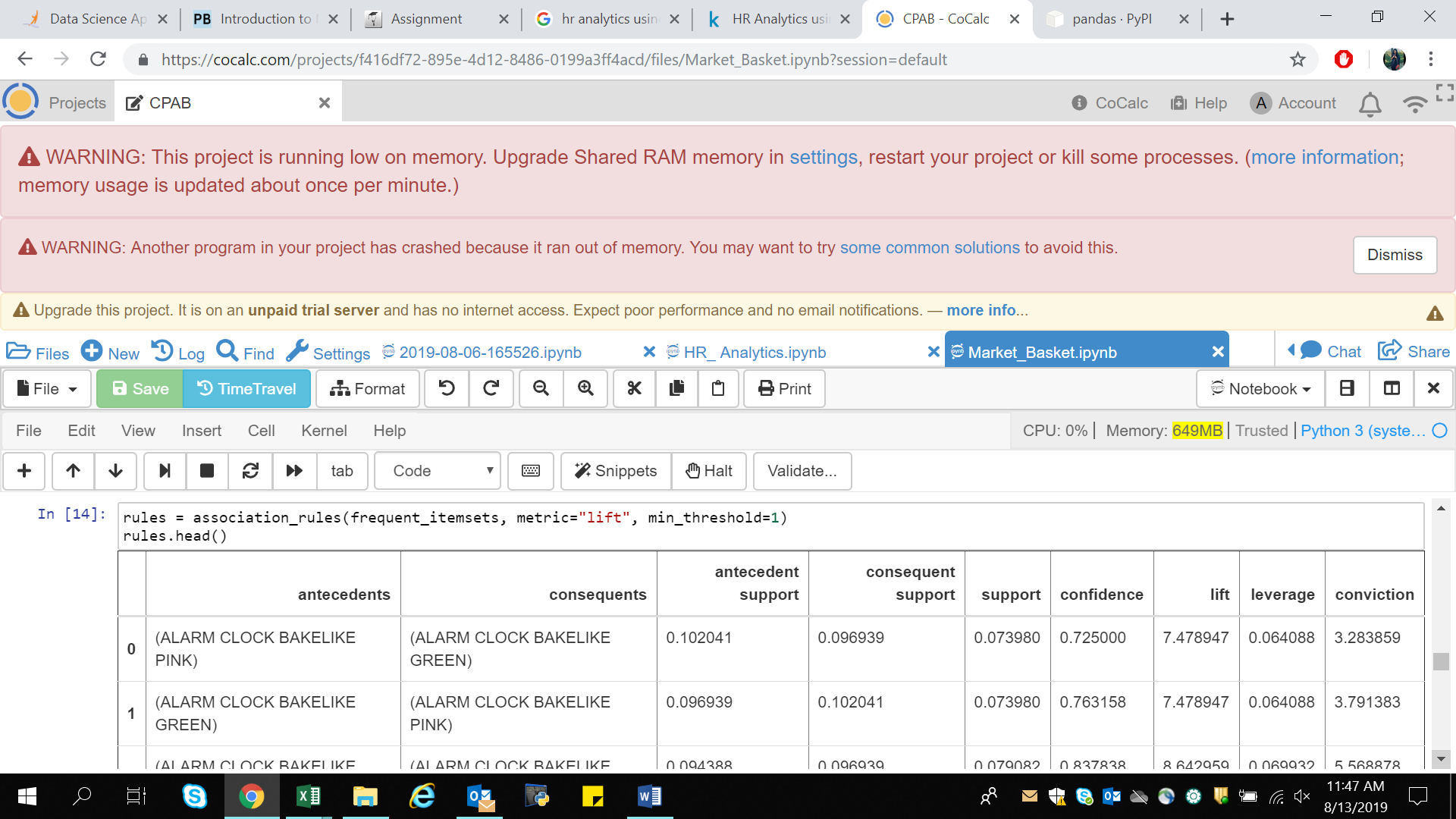
There are a lot of zero’s in the above data but we also need to make sure any positive values are converted to a 1 and anything less than 0 is set to 0. This step will complete the one encoding of the data and remove the postage column (since that charge is not one we wish to explore).



Now that the data is structured properly, we can generate frequent item sets that have a support of at least 7%.

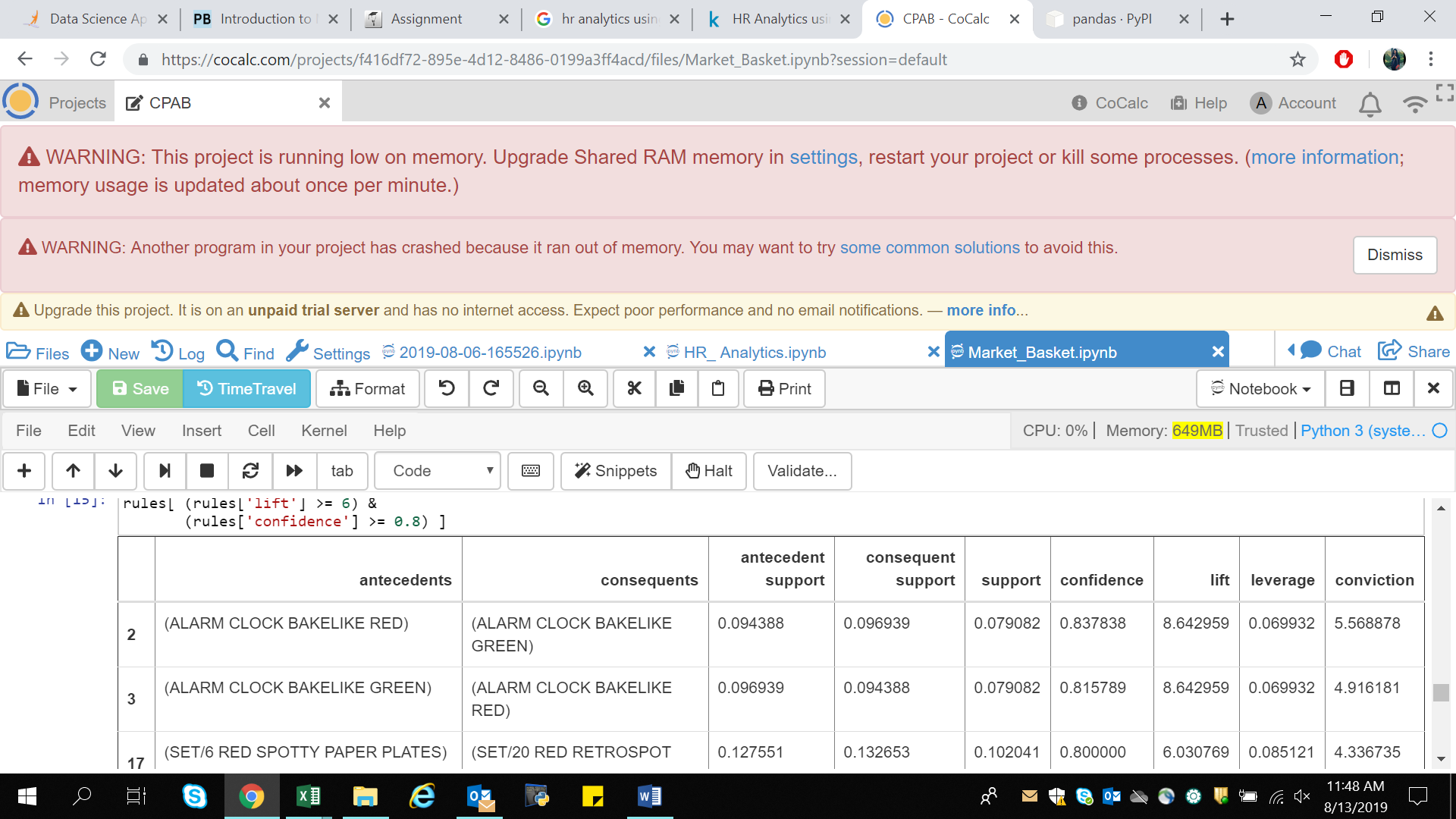


The final step is to generate the rules with their corresponding support, confidence and lift:



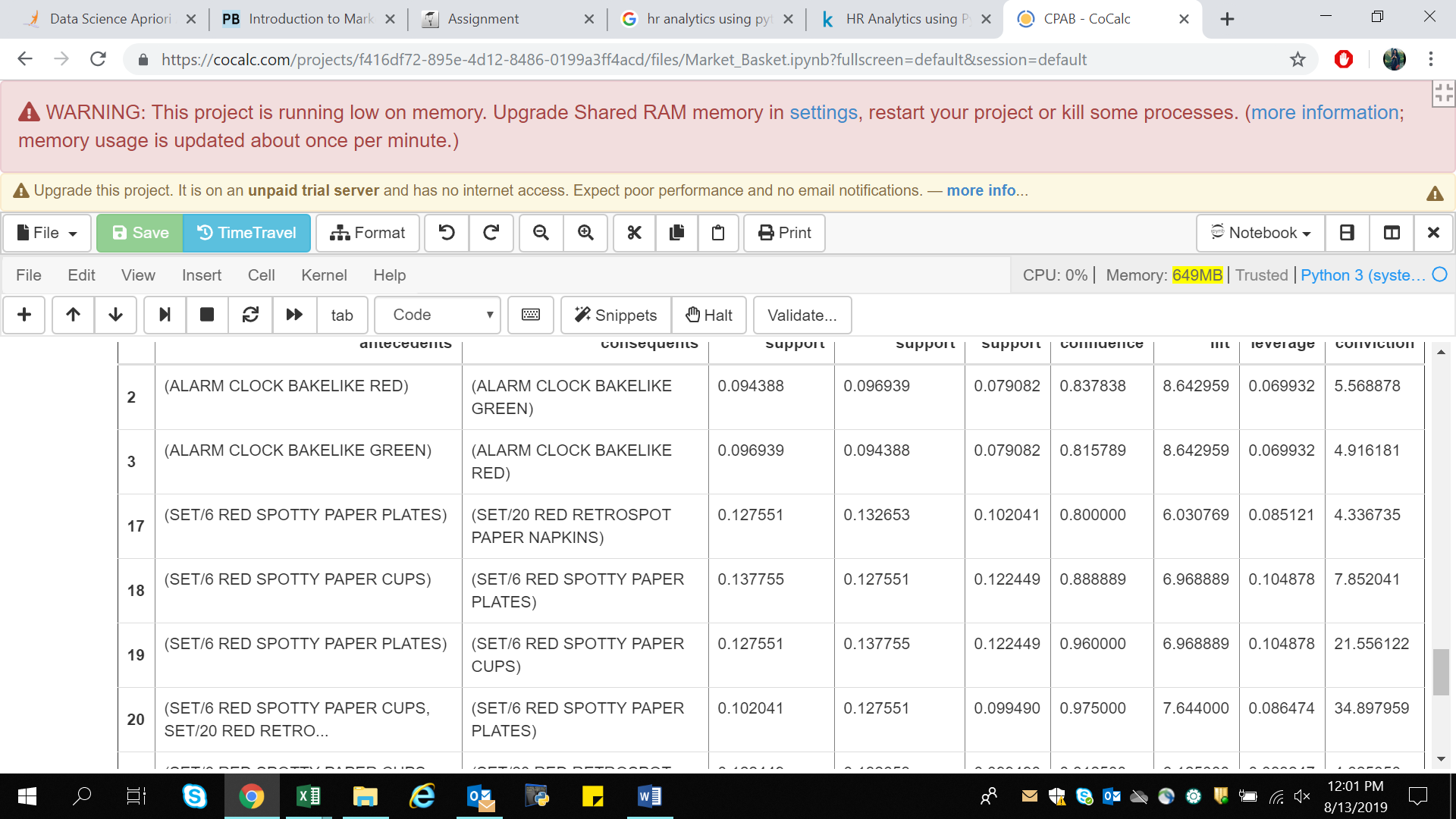
That’s all there is to it! Build the frequent items using apriori then build the rules with association rules.

We can filter the dataframe using standard pandas code. In this case, look for a large lift (6) and high confidence (0.8):



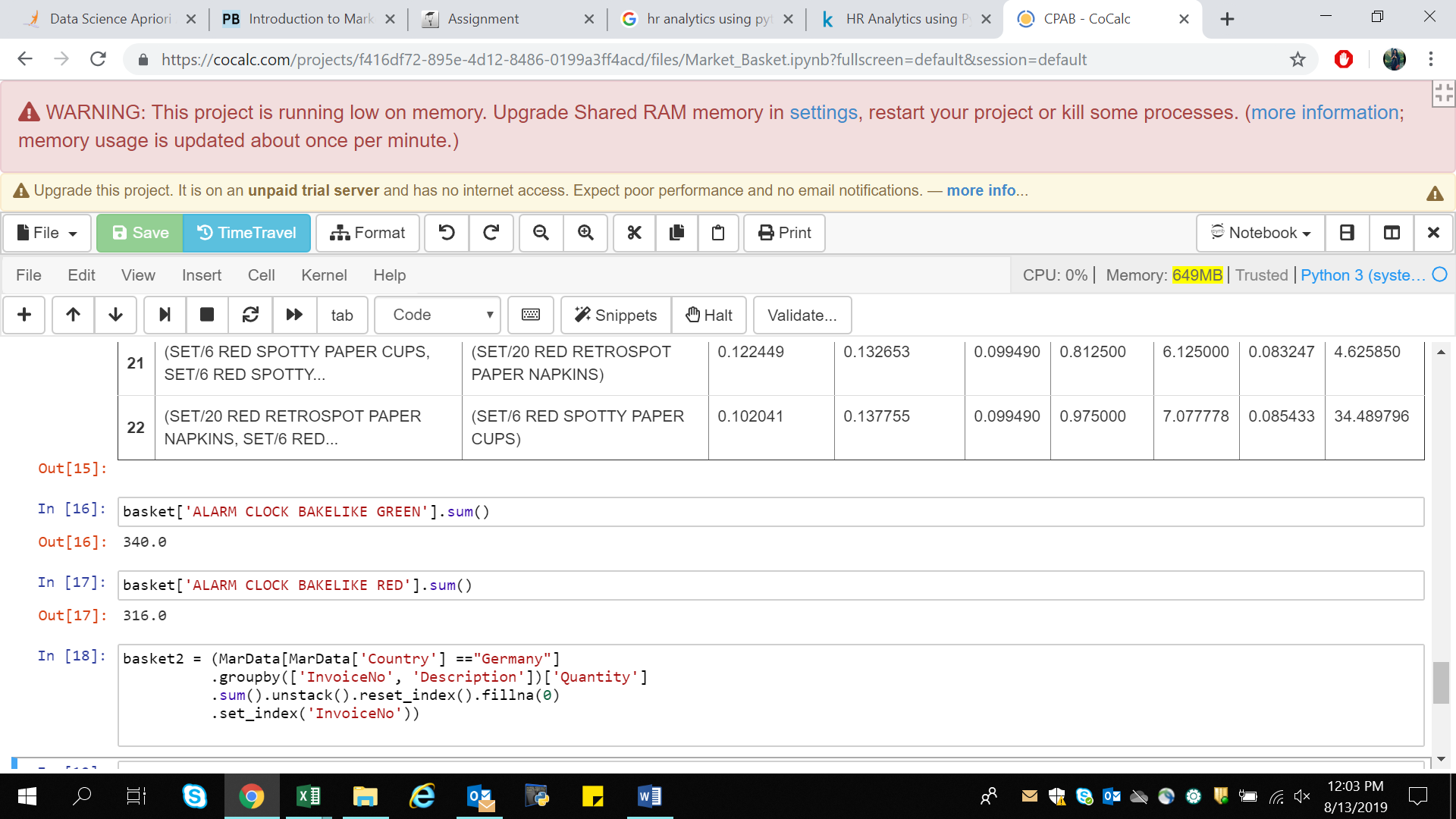
What do the rules say about?

We will find quite a few rules with a high lift value which means that it occurs more frequently than would be expected given the number of transaction and product combinations. We can also see several where the confidence is high as well. This part of the analysis is where the domain knowledge will come in handy.



So Here, it seems that the green and red alarm clocks are purchased together and the red paper cups, napkins and plates are purchased together in a manner that is higher than the overall probability would suggest.

At this point, we may want to look at how much opportunity there is to use the popularity of one product to drive sales of another.



For instance, we can see that we sell 340 Green Alarm clocks but only 316 Red Alarm Clocks so maybe we can drive more Red Alarm Clock sales through recommendations?

1. **Consumer Behavior Analysis-**

It is crucial to understand customer behavior and categorize customers based on their demography and buying behavior. This is a critical aspect of customer segmentation that allows marketers to better tailor their marketing efforts to various audience subsets in terms of promotional, marketing and product development strategies.

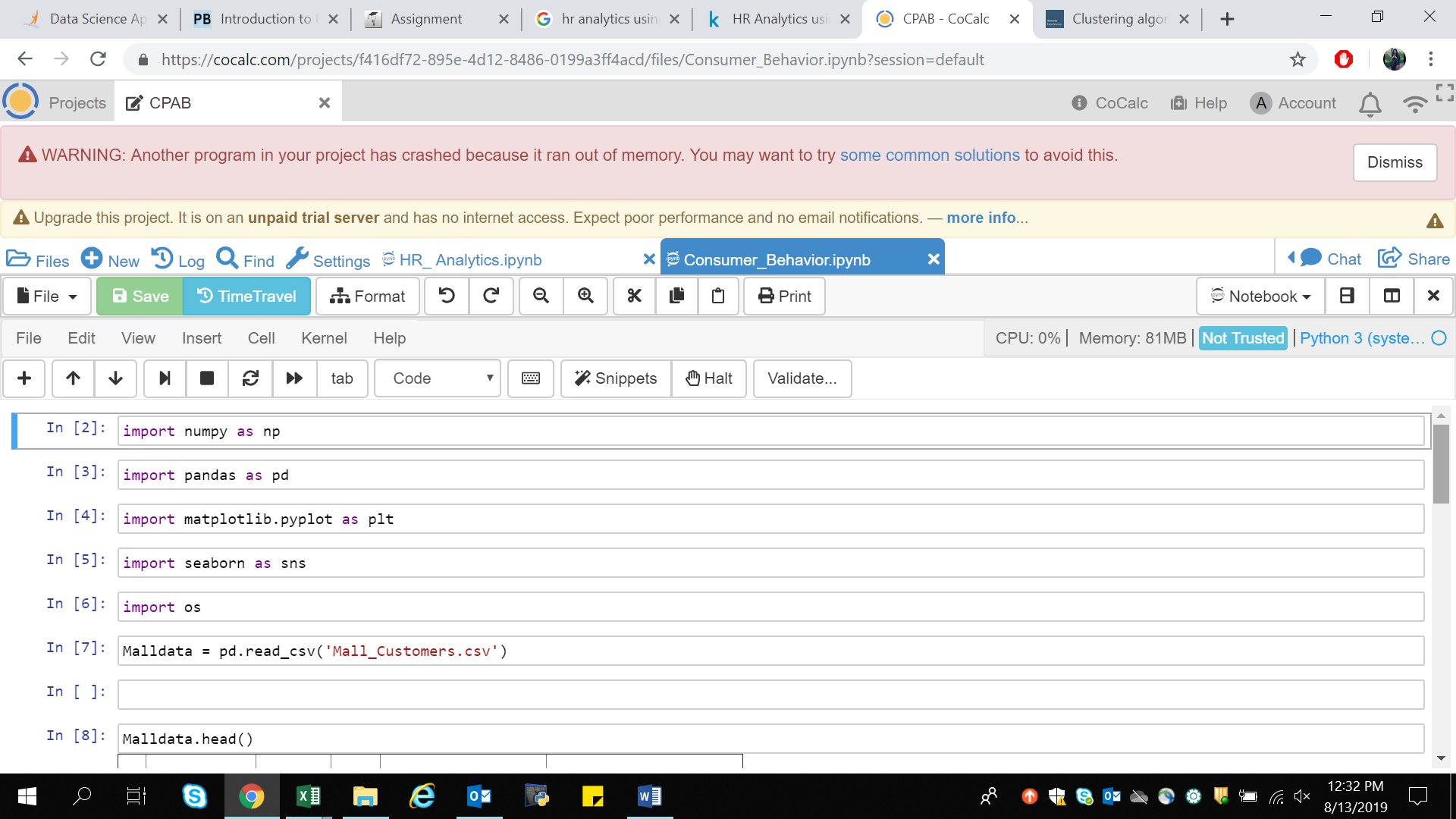
So with the help of unsupervised learning (k-means clustering) will be performing consumer behavior analysis.

The objective of any clustering algorithm is to ensure that the distance between data points in a cluster is very low compared to the distance between 2 clusters. In other words, members of a group are very similar, and members of different groups are extremely dissimilar.

About the Dataset-

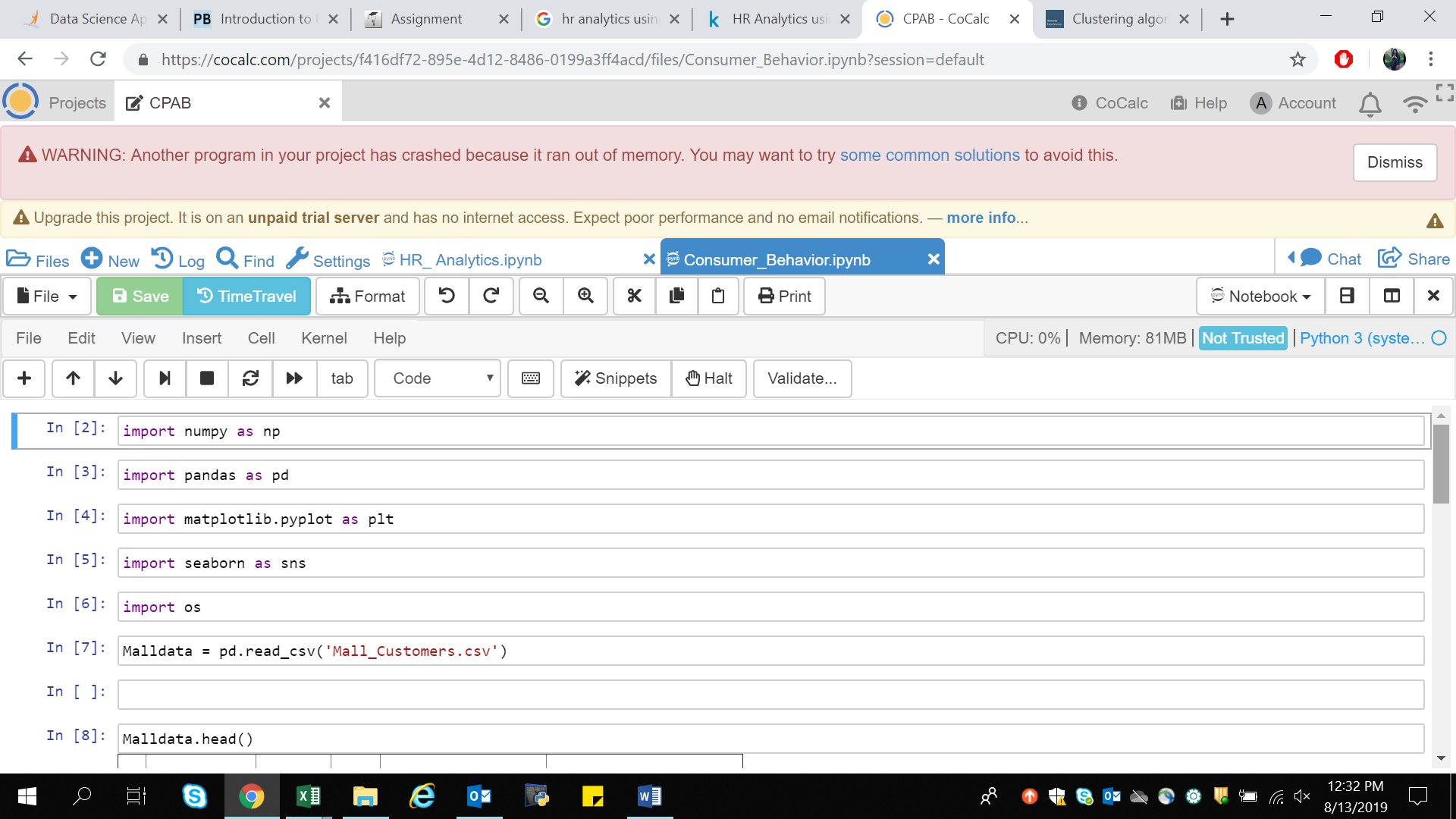
The dataset consists of Annual income (in $000) of 303 customers and their total spend (in $000) on an e-commerce site for a period of one year.

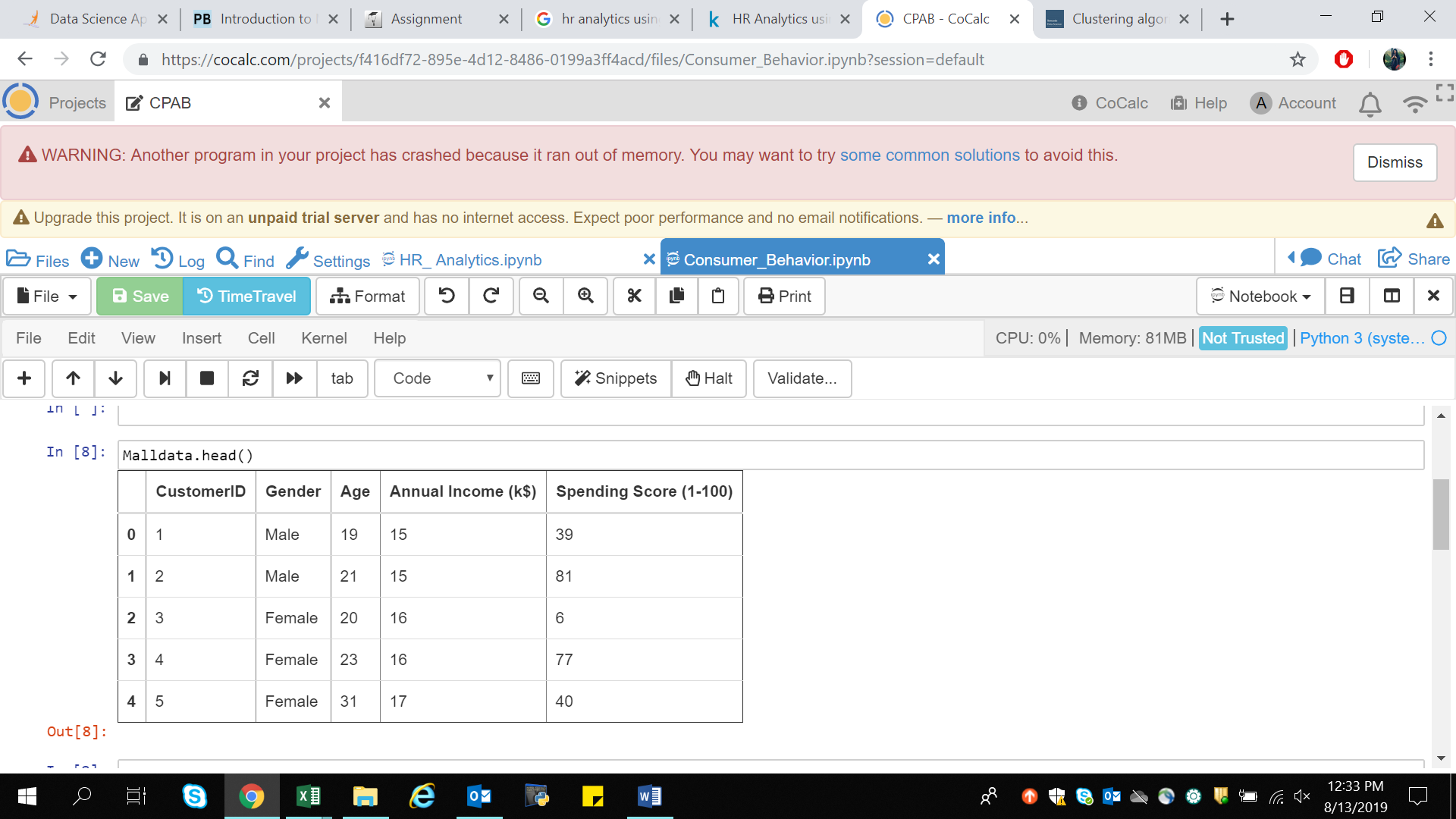
Library Used-



1. pandas- for data manipulation and data analysis.
2. numpy- for high-performance multidimensional array object, and tools for working with these arrays
3. matplotlib.pyplot- for visualization.
4. seaborn- for visualization.
5. os- for directory.
6. kmeans- for clustering.

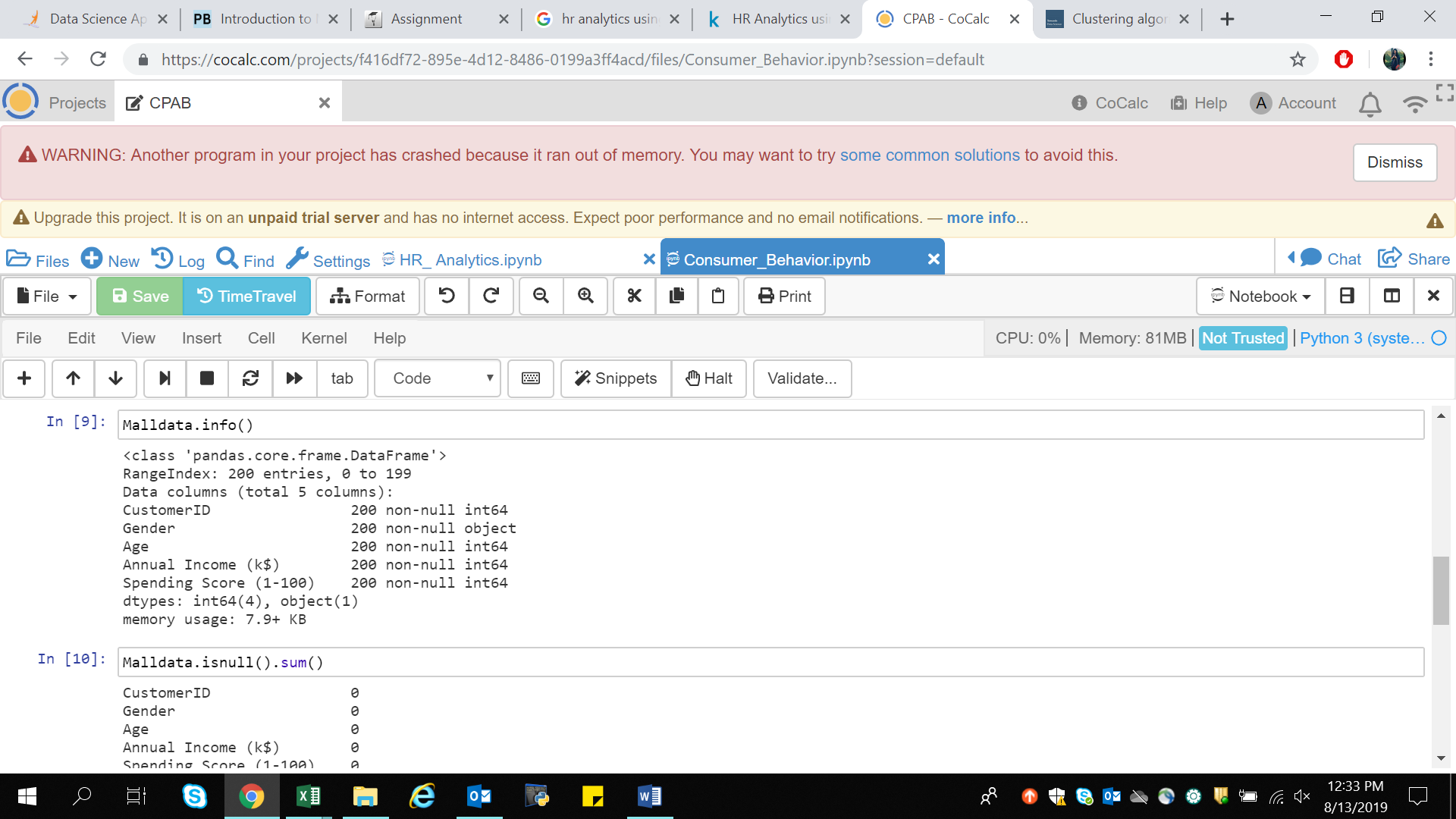
Data is imported using pandas library under the name Malldata



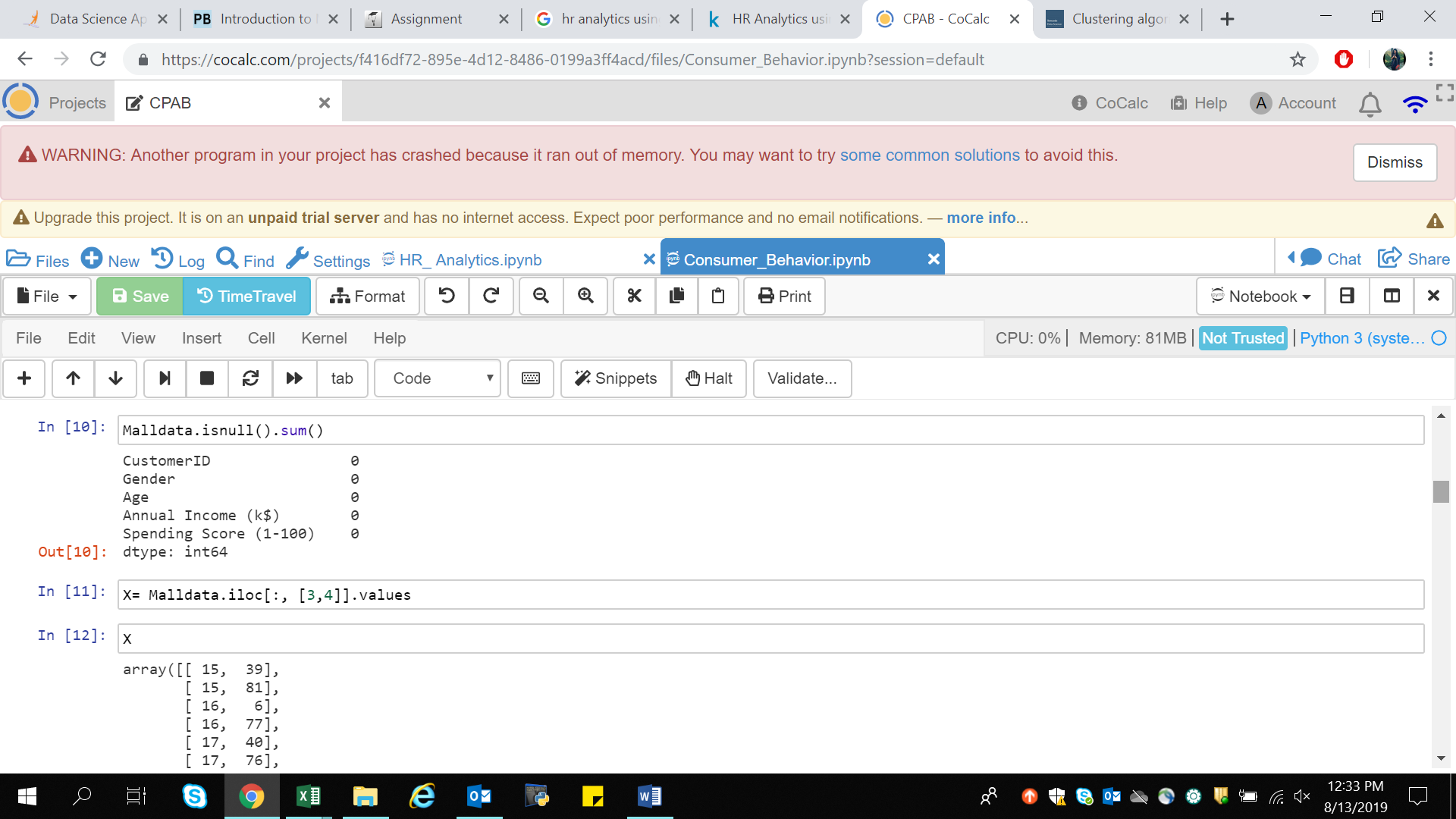


The above picture shows the data contained by the dataset.

Dataset contains 200 rows and 5 columns.



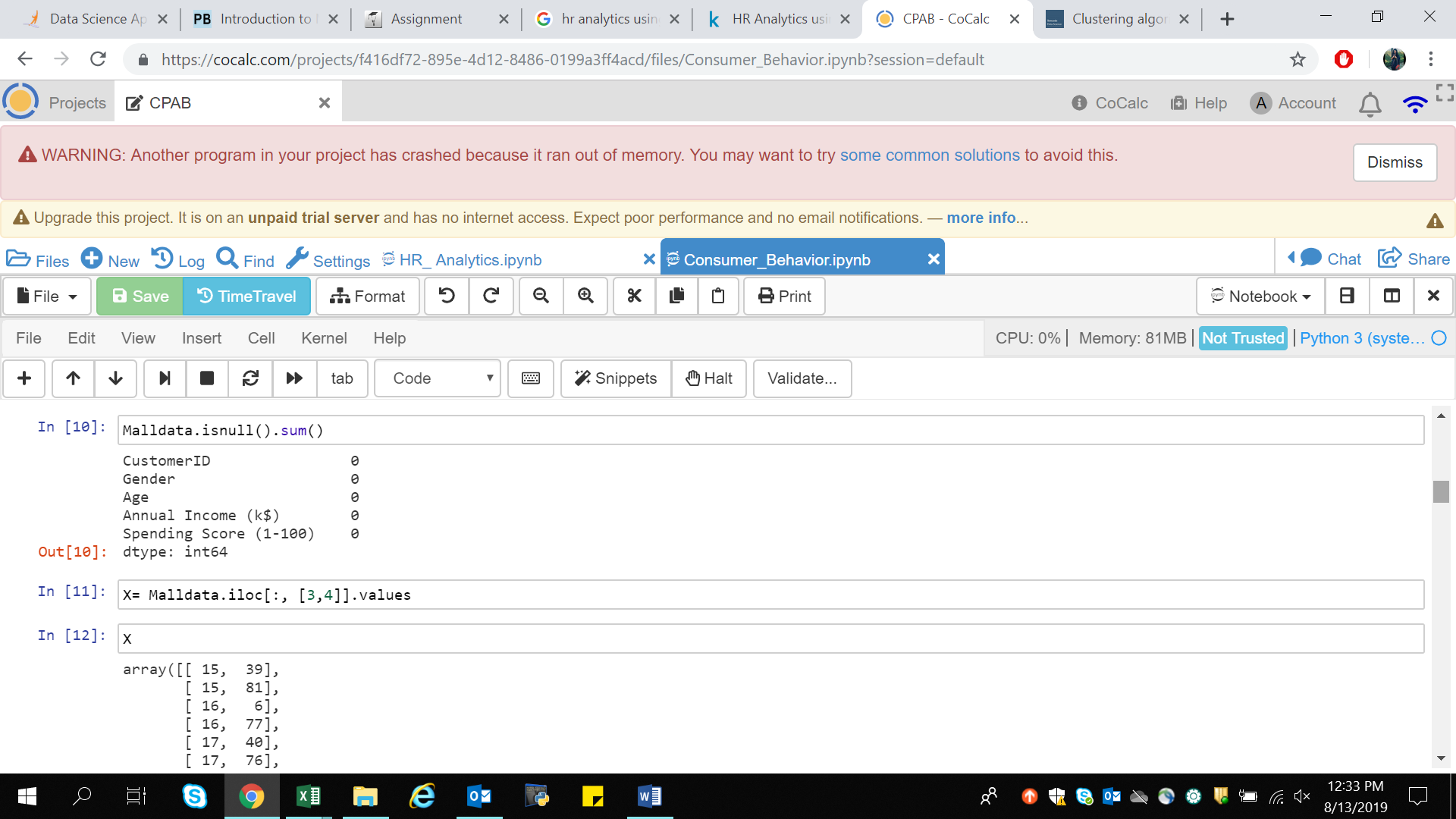
To check data contain any white spaces or Null values can be viewed using dataobject.info() or dataobject.isnull() functions .



This shows the data contains no Null values. Therefore data cleaning is not required.

For building the model and moving to our next stage of analysis we need to select feature for the model. Here we are considering only 2 features (Annual income and Spending Score) with no Label available.

For the same store the data into some variable using dataobject.iloc[ ].values command



Now for Building the Mode I will be using K-Means Algorithm to decide the optimum cluster number , K-Means++ using Elbow method.

Elbow Method- to figure out K for K-Means++, I will use ELBOW Method.

[K-means](https://en.wikipedia.org/wiki/K-means_clustering) is a simple [unsupervised machine learning](https://en.wikipedia.org/wiki/Unsupervised_learning) algorithm that groups a dataset into a user-specified number (k) of clusters. The algorithm is somewhat naive--it clusters the data into k clusters, even if k is not the right number of clusters to use. Therefore, when using k-means clustering, users need some way to determine whether they are using the right number of clusters.

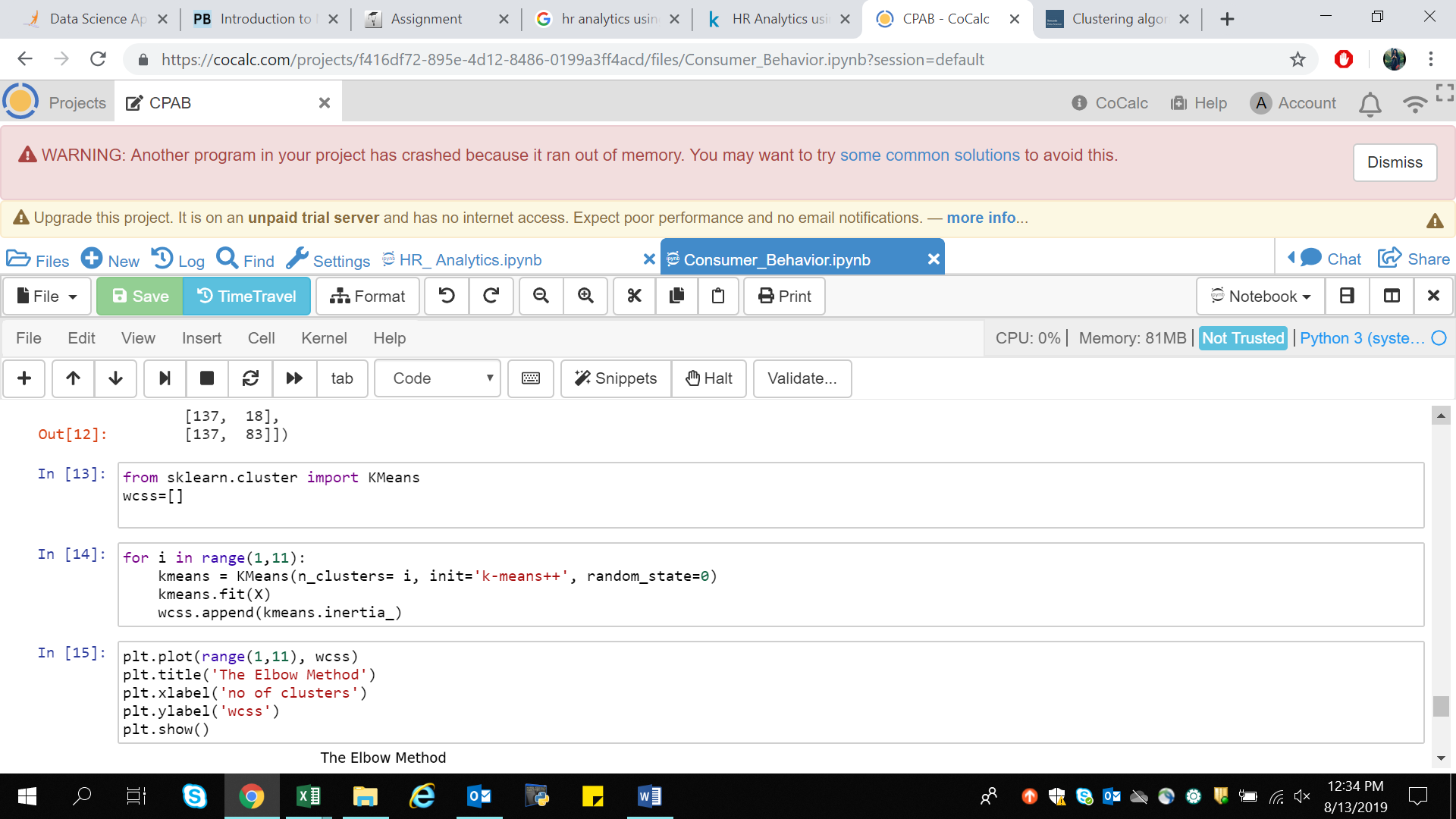
One method to validate the number of clusters is the elbow method. The idea of the elbow method is to run k-means clustering on the dataset for a range of values of k (say, k from 1 to 10 in the examples above), and for each value of k calculate the sum of squared errors (SSE).

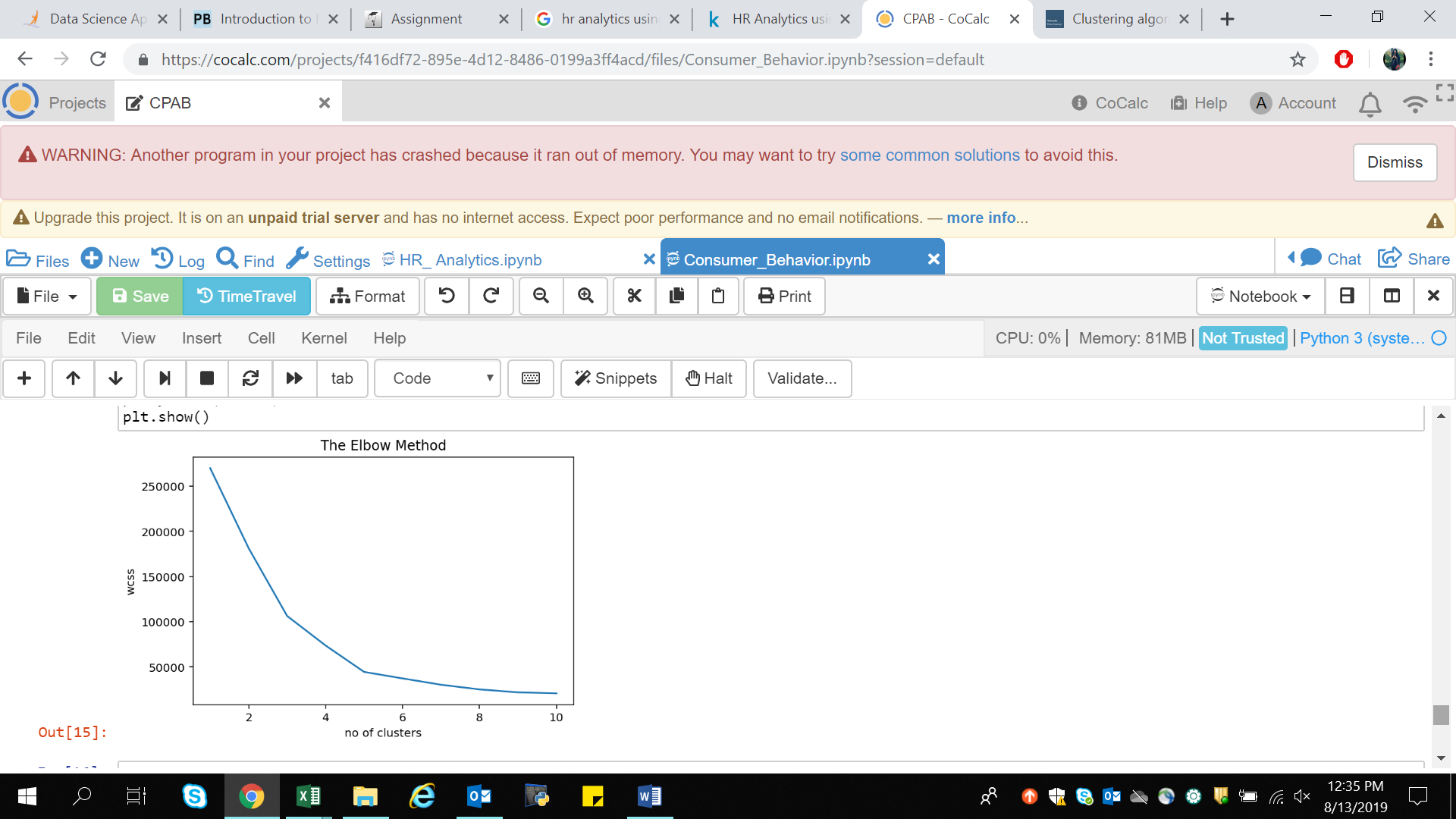
Then, plot a line chart of the SSE for each value of k. If the line chart looks like an arm, then the "elbow" on the arm is the value of k that is the best.

WCSS is within-cluster-sum-of-squares to measures sum of distances of observations from their cluster centroids.

Implementing k-means to get maximum number of clusters. inertia\_ is the formula used to segregate the data points into clusters

Plotting the elbow method for visualization of optimal values of k form by using plot() functions.





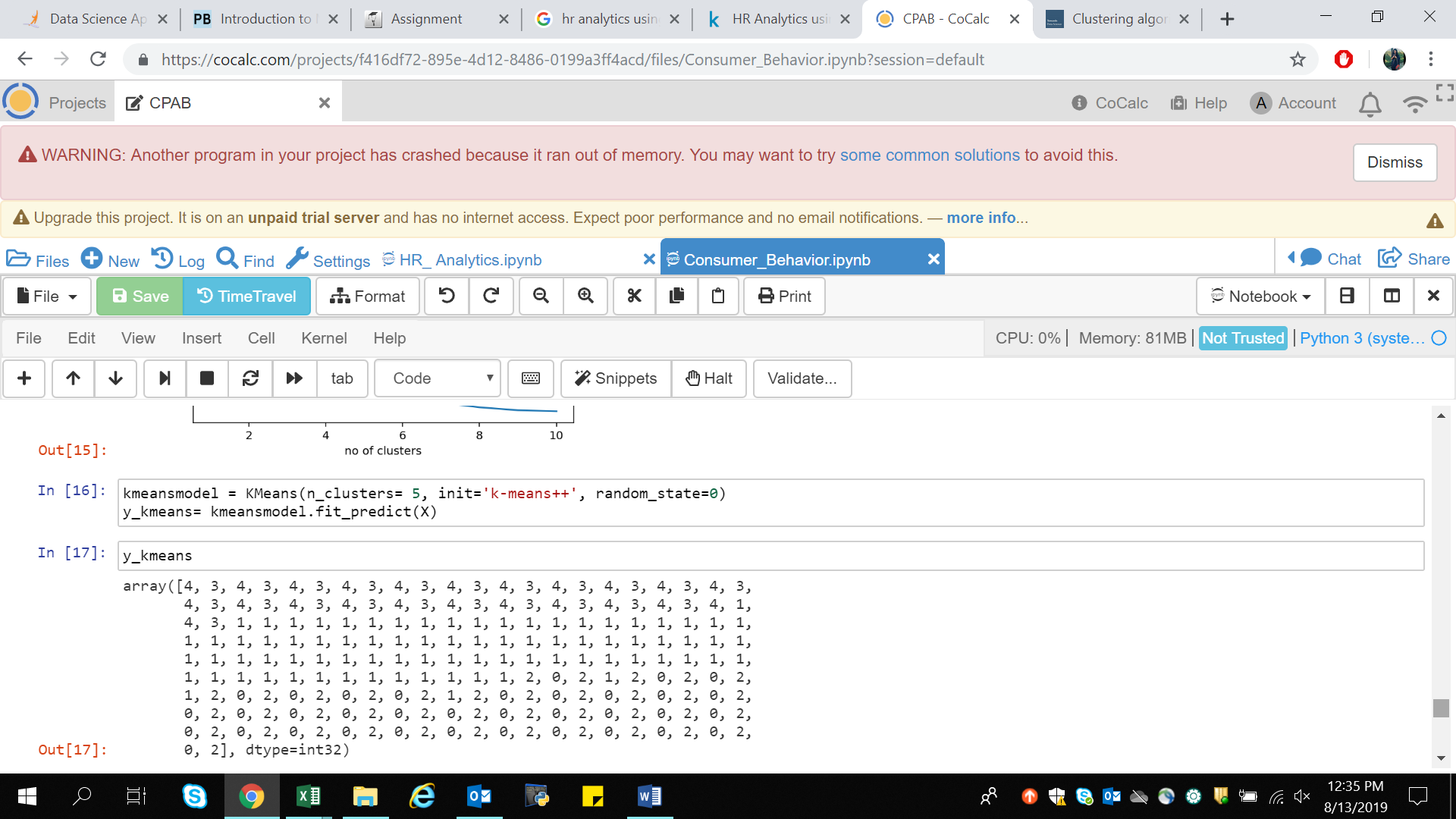
If we see in the above plot last elbow comes at k=5 but if we chose higher range it is little difficult to visualize the ELBOW that is why we are preferring between range (1,11)

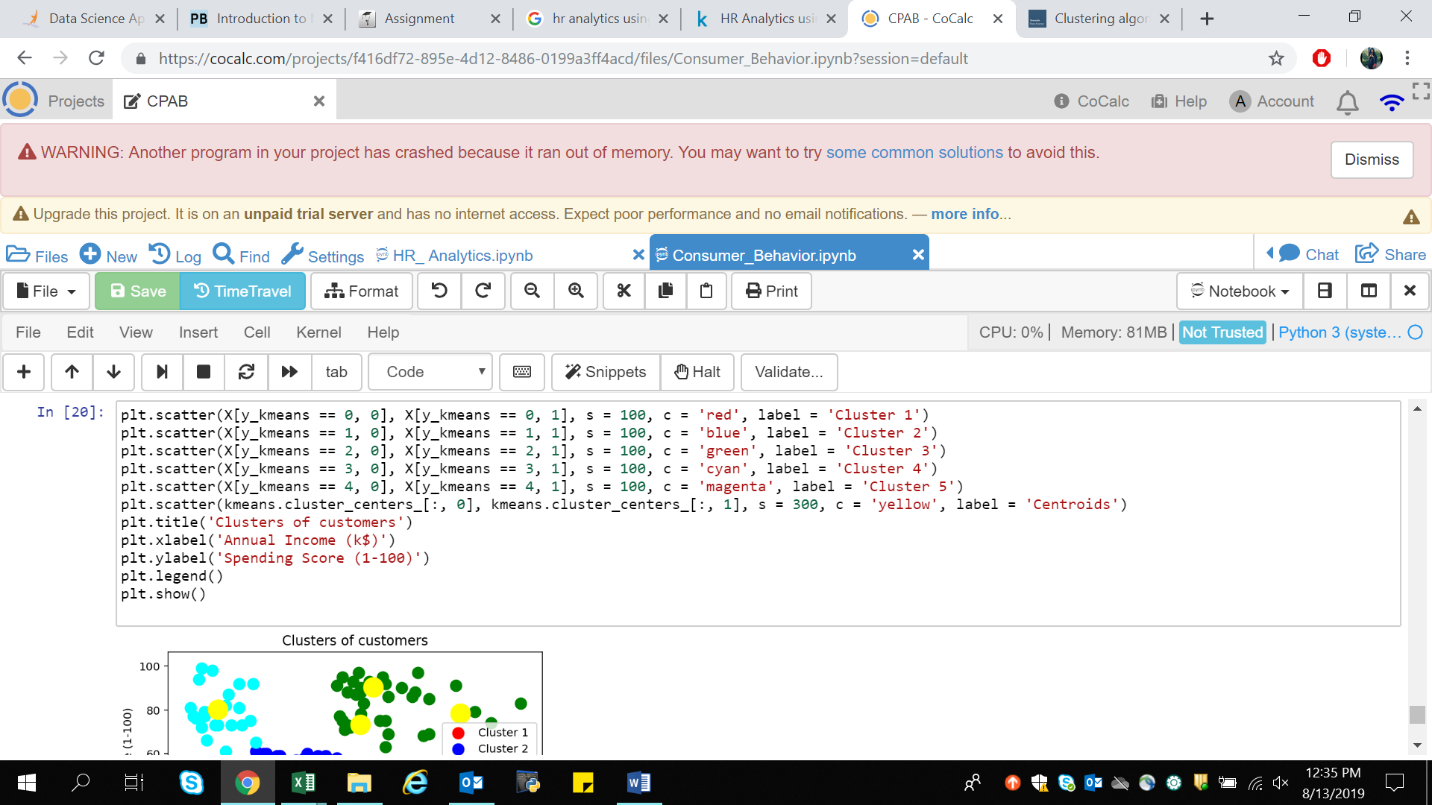
With that I got the value of k, k=5.

For unsupervised learning we use "fit\_predict()" wherein for supervised learning we use "fit\_tranform()".

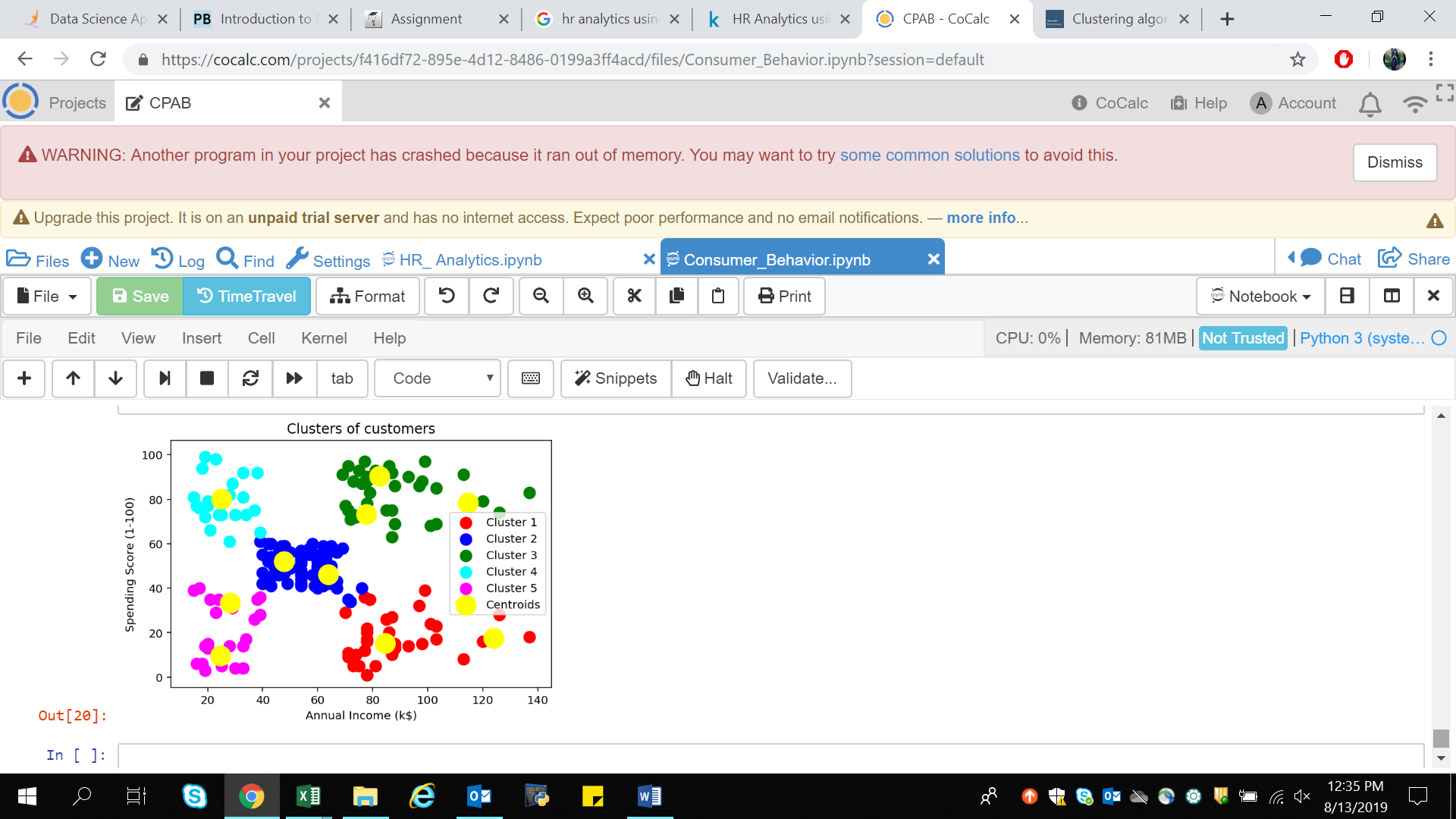
As we are working with unsupervised learning will be using fit.predict().

y\_kmeans is the final model . Now how and where we will deploy this model in production is depends on what tool we are using. This use case is very common and it is used in BFS industry(credit card) and retail for customer segmentation.

****For visualizing all the 5 clusters.



The Below graph represent the model interpretation.

1. Cluster 1 (Red Color) - Earning high but spending less
2. Cluster 2 (Blue Color) - Average in terms of earning and spending
3. cluster 3 (Green Color) - Earning high and also spending high [TARGET SET]
4. Cluster 4 (cyan Color) - Earning less but spending more
5. Cluster 5 (magenta Color) - Earning less , spending less.

So what we can analyze from the below clustering is we can put Cluster 3 into some alerting system as they are the target audience and will be helping in increasing the sales of business/retail/malls wherein others we can set like once in a week or once in a month.

As customer is king every customer is valued but with this we got the prior customer to be targeted.

* A typical strategy would focus certain promotional efforts for the high value customers of Cluster 6 & Cluster 3.
* Cluster 4 is a unique customer segment, where in spite of their relatively lower annual income, these customers tend to spend more on the site, indicating their loyalty. There could be some discounted pricing based promotional campaigns for this group so as to retain them.
* For Cluster 2 where both the income and annual spend are low, further analysis could be needed to find the reasons for the lower spend and price-sensitive strategies could be introduced to increase the spend from this segment.
* Customers in clusters 1 and 5 are not spending enough on the site in spite of a good annual income — further analysis of these segments could lead to insights on the satisfaction / dissatisfaction of these customers or lesser visibility of the e-commerce site to these customers. Strategies could be evolved accordingly.

1. **HR ANALYTICS –**