PYTHON DATA EXPLORATION: CRITICAL THINKING 1 OPT 1

Sean Senko

MIS500 – Foundations of Data Analytics

Colorado State University – Global Campus

Dr. Dwight Davis

April 18, 2019

**CODE BLOCK**

**#BEGIN**

*#############################################################  
# Program Begin HERE  
# Some data Exploration using Python. Assuming that all the needed packages  
# are already install for your IDE to find them.  
#############################################################  
# Program name - Data Exploration  
# input - NONE  
# output - Some Exploration statistics  
###############################################################***import** pandas **as** pd  
  
*# Create data\_frame of array values*df = pd.DataFrame({  
    **'name'**:[**'matt'**,**'lisa'**,**'richard'**,**'john'**,**'Julia'**,**'jane'**,**'marlon'**],  
    **'age'**:[23,78,22,19,45,33,20],  
    **'gender'**:[**'M'**,**'F'**,**'M'**,**'M'**,**'M'**,**'F'**,**'M'**],  
    **'state'**:[**'DC'**,**'CO'**,**'DE'**,**'VA'**,**'MD'**,**'DE'**,**'NY'**],  
    **'years\_of\_service'**:[10,0,2,0,2,1,5],  
    **'iq'**:[300,100,110,200,300,10,40]  
})  
*########################################################  
# BEGIN extract a 25% sample of data  
########################################################*rows = df.sample(frac =.25)  
*# validate first to check if sample is 0.25 times data or not***if** (0.25\*(len(df))== len(rows)):  
    **print**(len(df), len(rows))  
  
*# Display Sample Percentage***print ('sample of 25%'**,rows)  
  
*# END extract a 25% sample of data  
############################################################  
# BEGIN Split categorical variables by gender, Sum, Mean, count,  
# and describe on the data  
############################################################  
  
# Categorical Variables splitting*groupby\_gender = df.groupby(**'gender'**)  
**for** gender, value **in** groupby\_gender[**'iq'**]:  
    **print**((gender, value.mean()))  
  
*# Find the Summation of all ages in the data*SumofAge=df[**'age'**].sum()  
**print ('Sum of Ages'**, SumofAge)  
MeanAge = df[**'age'**].mean()  
**print ('Average Ages'**, MeanAge)  
*# Find the mean of all columns***print** (**'Means of each column'**, df.mean(axis=0))  
*# Describe the Data***print (**df[**'iq'**].describe())

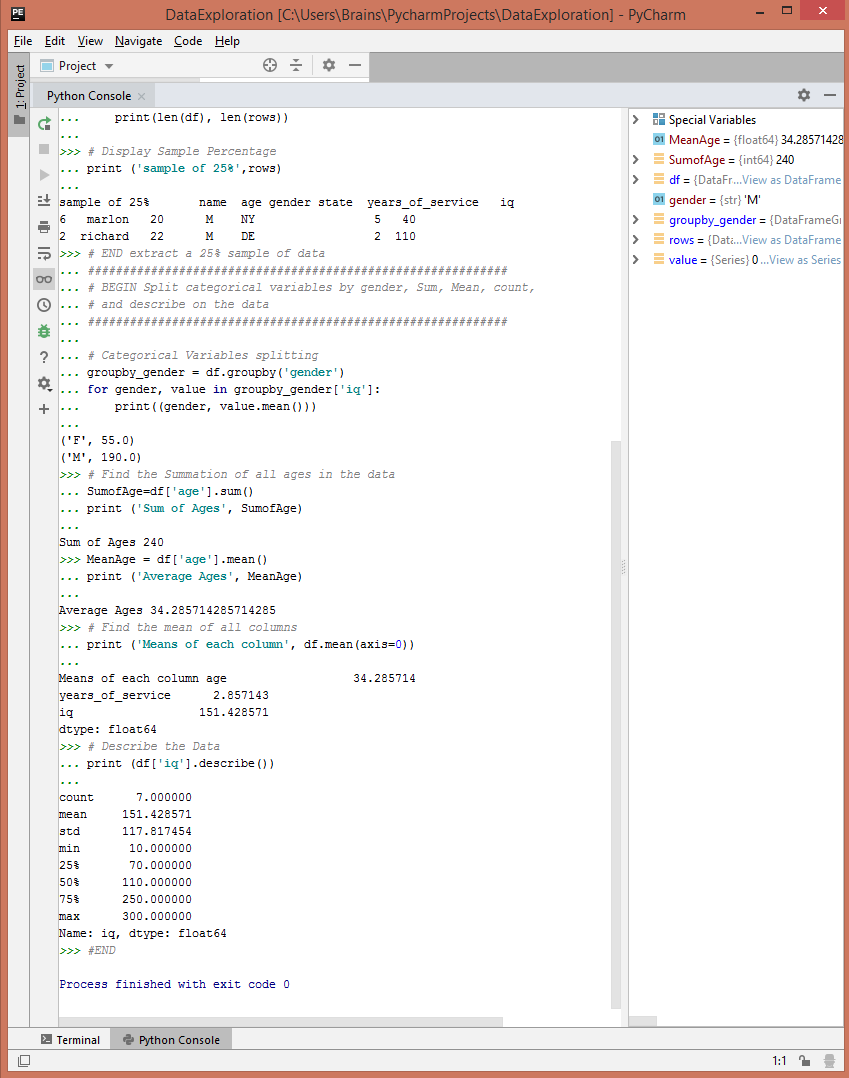
**#END**

Notes:

The screenshots are on the next two pages. It is worth noting that a triple quote does not actually produce a multiple line comment in Python; it actually allows multiple lines to be included in a print statement. As such, I edited one section of the code to reflect this, replacing the triple quotes with “#” with multiple lines in the code. Since I plan on uploading this to Github eventually, I want to make sure my technique is correct. Also, the only real error with the original code was a lack of parentheses around several print statements, but Python caught that quickly. This was not a hard assignment overall.

****

**Screenshot of the first half of the program**

****

**Screenshot of the remainder of the program**