**import** **pandas** **as** **pd**

**import** **numpy** **as** **np**

**import** **matplotlib.pyplot** **as** **plt**

**PART 1**

*#Question 1*

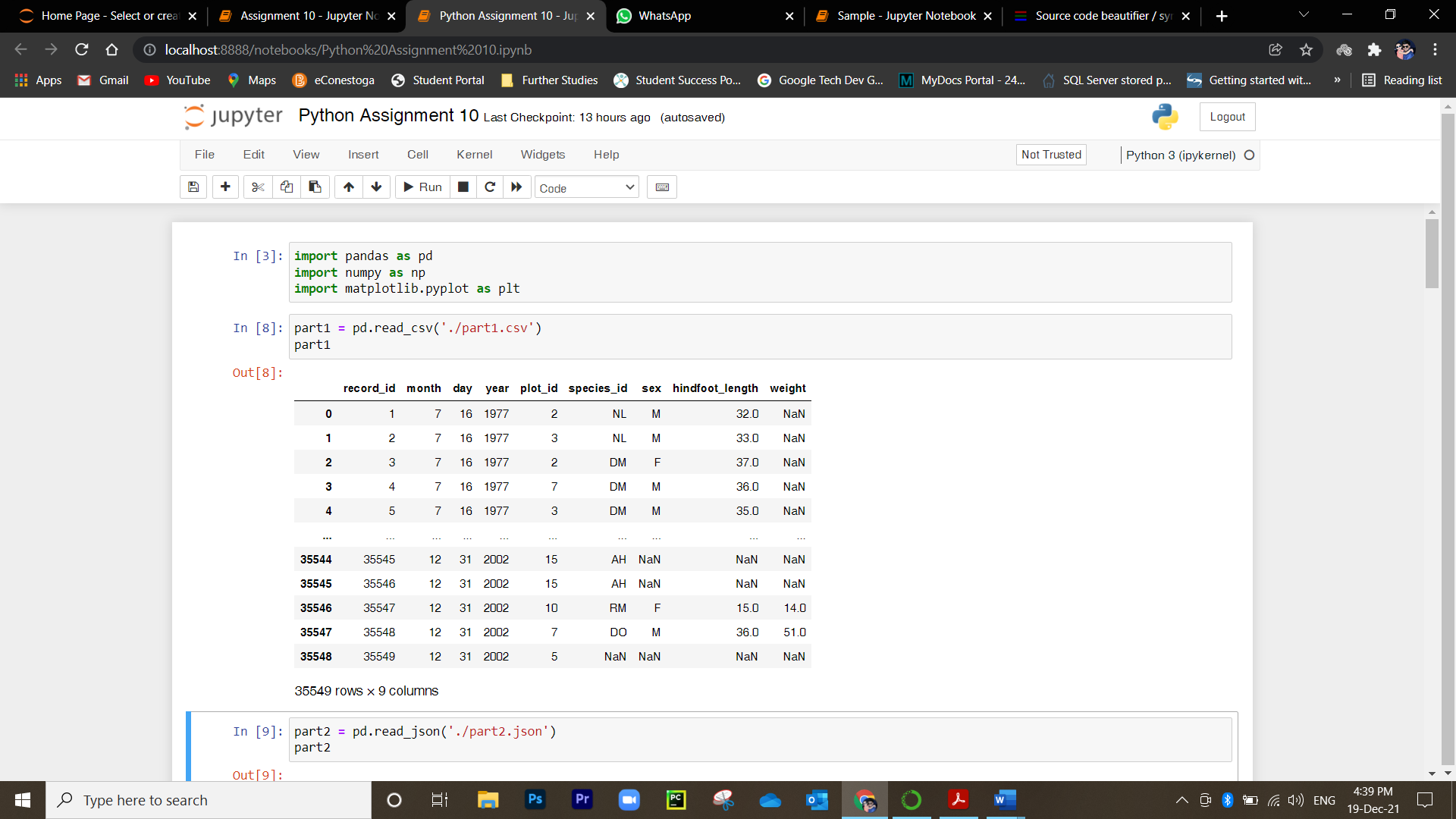
*Use the pandas library to read data files:* ***part1.csv*** *and* ***part2.json****.to create 2 data frames.*

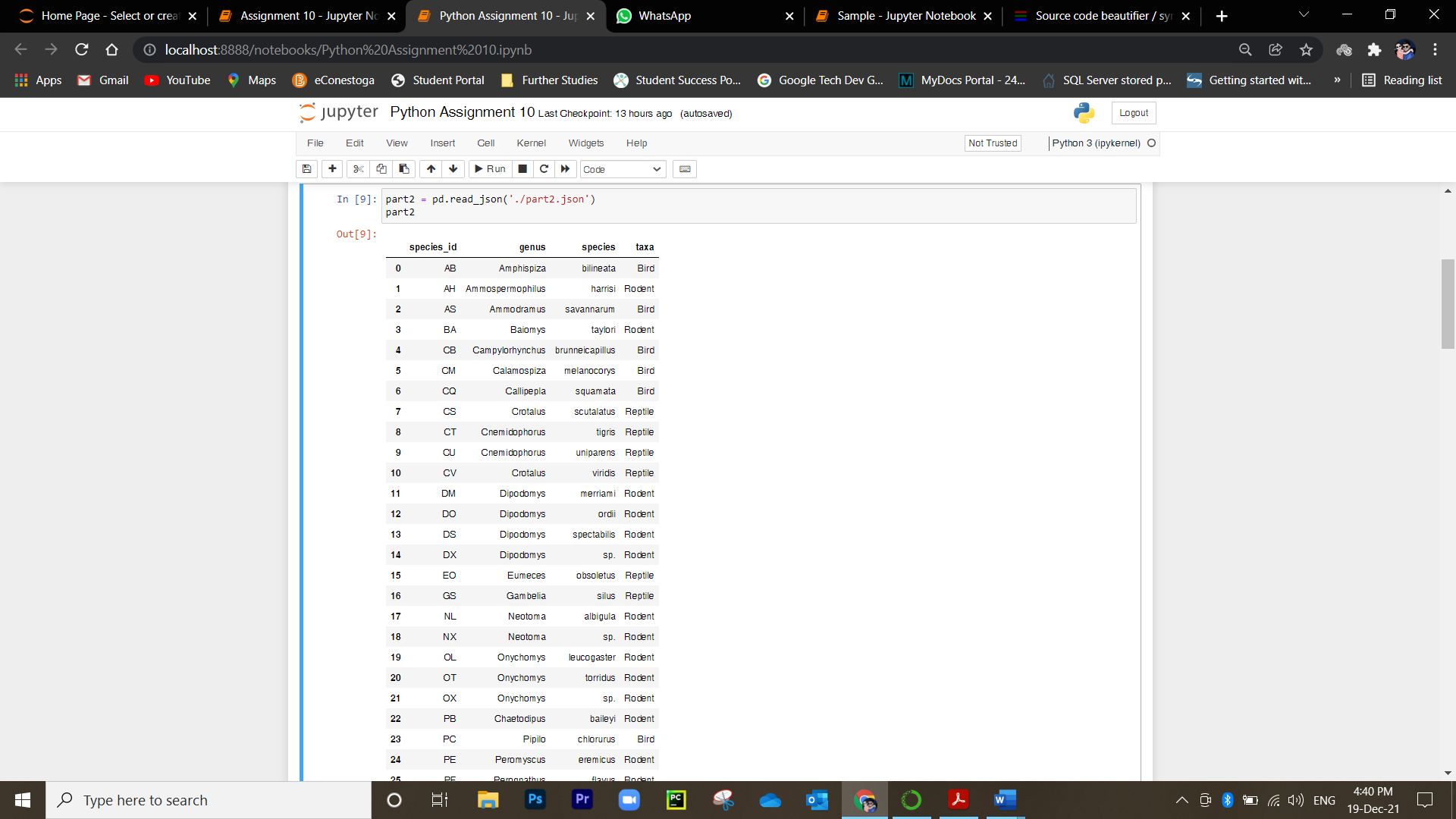
part1 = pd.read\_csv('./part1.csv')

part1

part2 = pd.read\_json('./part2.json')

part2





*#Question 2*

*Display the first* ***5*** *rows and the last* ***3*** *rows of both data frames.*

RowColumns1 = pd.concat([part1.head(5), part1.tail(3)])

RowColumns1

Or # part1.head(5)

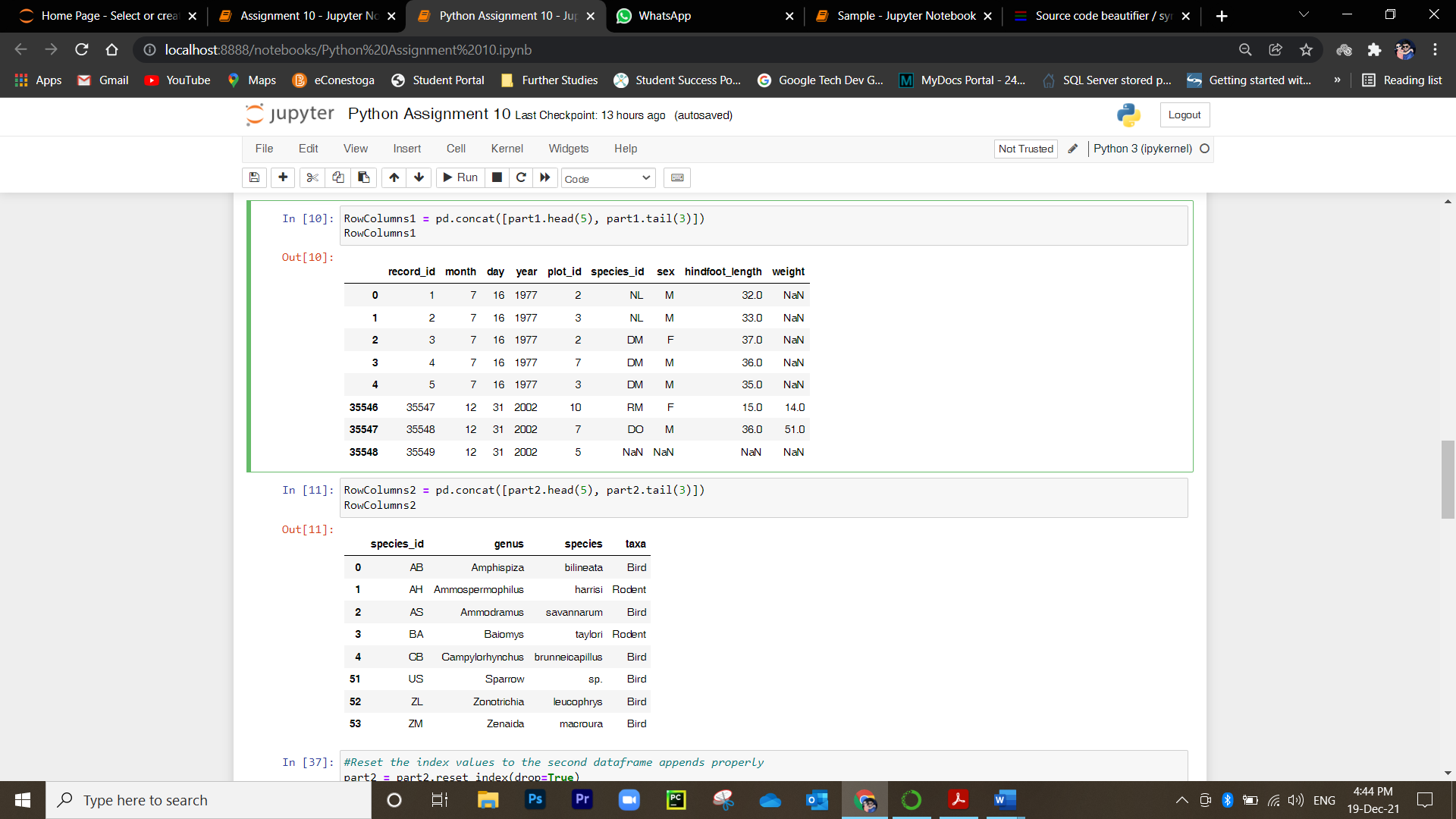
# part1.tail(3)

RowColumns2 = pd.concat([part2.head(5), part2.tail(3)])

RowColumns2

Or # part2.head(5)

# part2.tail(3)



*#Question 3*

*Combine both data frames into a single one: Columns in the second data frame to the RIGHT of the first data frame.*

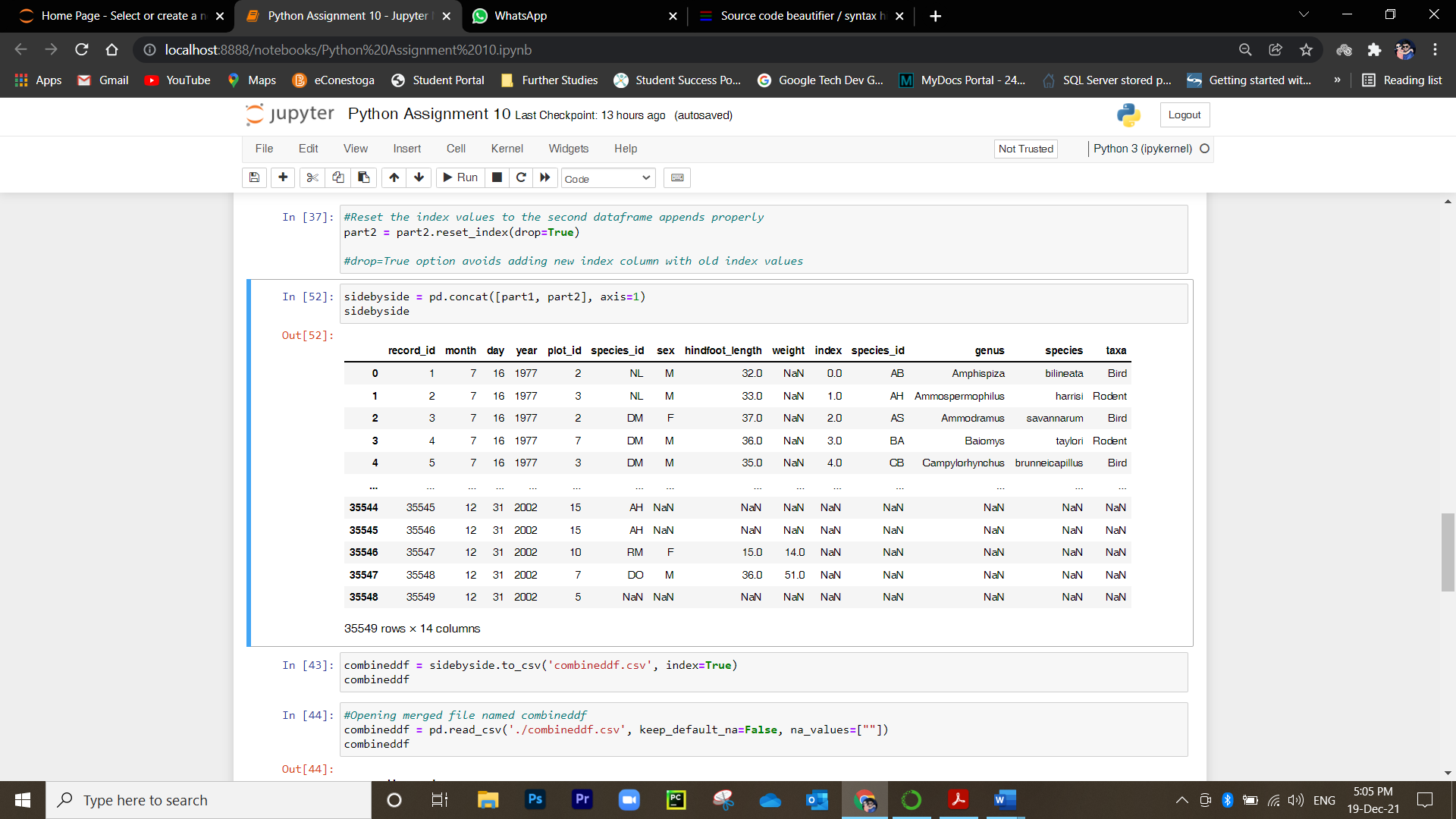
*#Reset the index values to the second dataframe appends properly*

part2 = part2.reset\_index(drop=**True**)

*#drop=True option avoids adding new index column with old index values*

sidebyside = pd.concat([part1, part2], axis=1)

sidebyside



*#Question 4*

*Export your results as a CSV and make sure it reads back into Python properly.*

combineddf = sidebyside.to\_csv('combineddf.csv', index=**True**)

combineddf

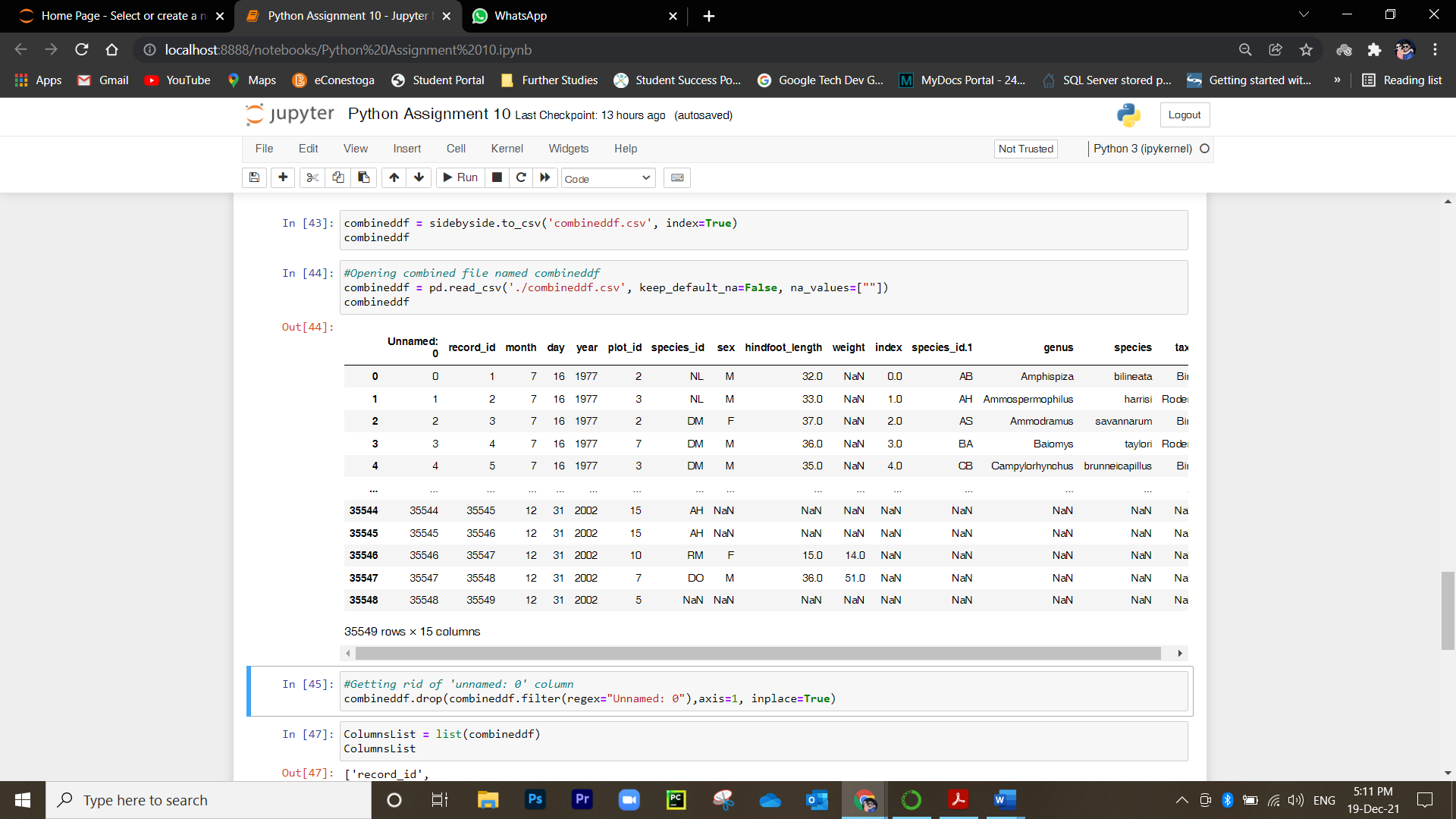
*#Opening combined file named combineddf*

combineddf = pd.read\_csv('./combineddf.csv', keep\_default\_na=**False**, na\_values=[""])

combineddf

*#Getting rid of 'unnamed: 0' column*

combineddf.drop(combineddf.filter(regex="Unnamed: 0"),axis=1, inplace=**True)**

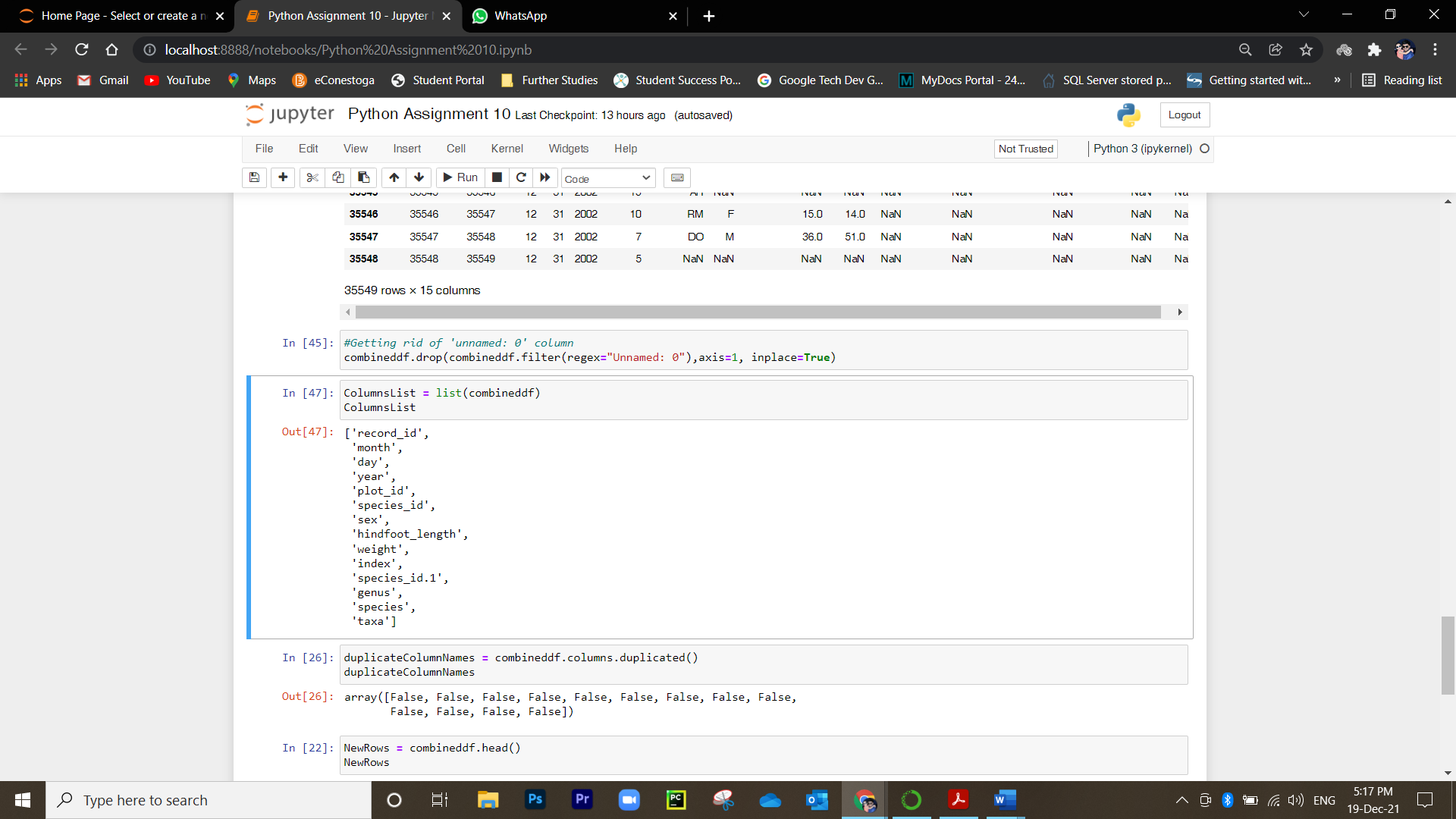


*#Question 5*

*Show the list of columns in your data frame.*

ColumnsList = list(combineddf)

ColumnsList



*#Question 6*

*Do you find any duplicated columns? Display the first 5 rows of your data, is your data accurate?*

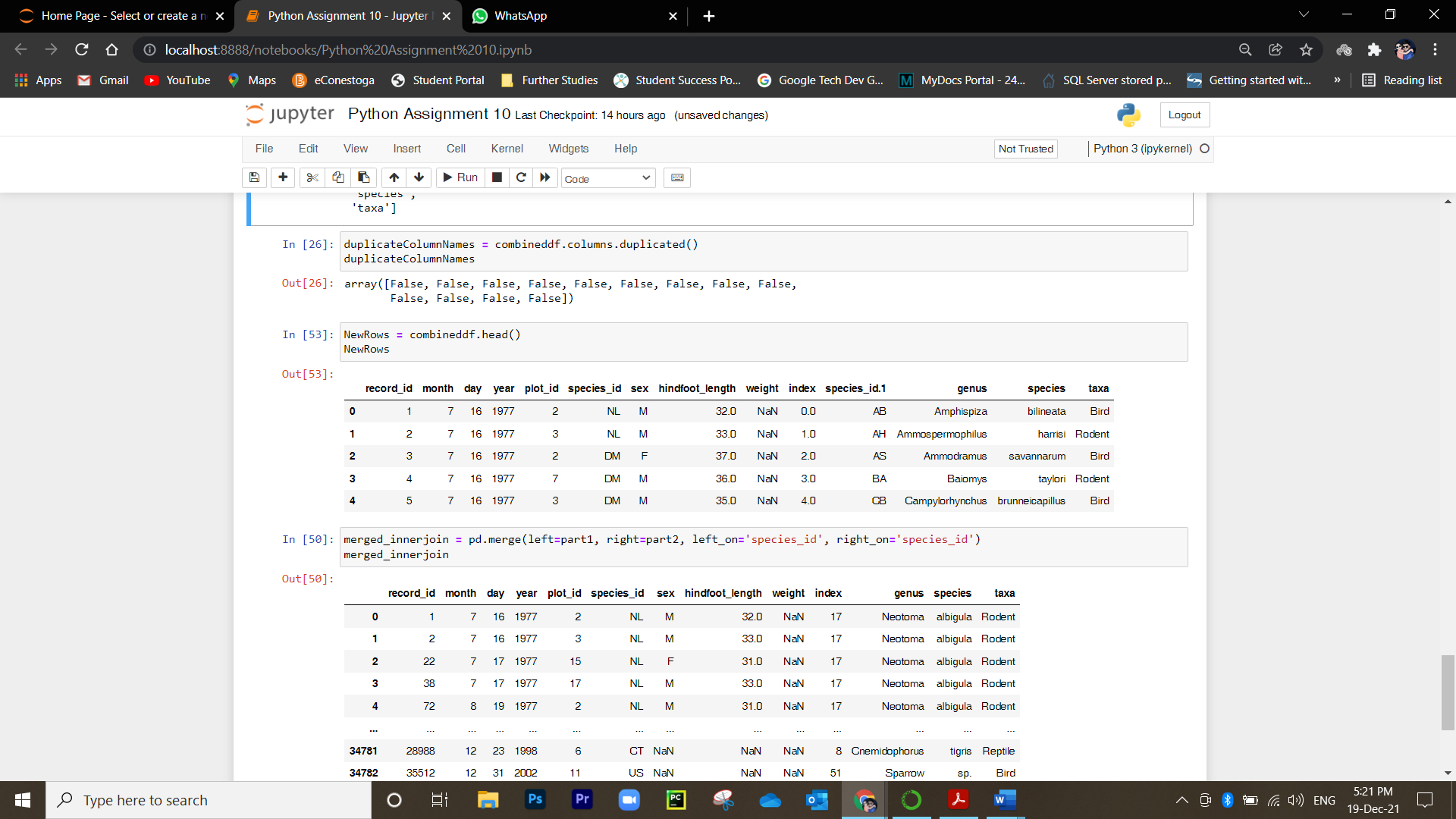
No, There is no duplicate column present in the combined dataframe. Moreover, the data seems unaccurate as it does not reflect the data as per the species\_id.

duplicateColumnNames = combineddf.columns.duplicated()

duplicateColumnNames

NewRows = combineddf.head()

NewRows

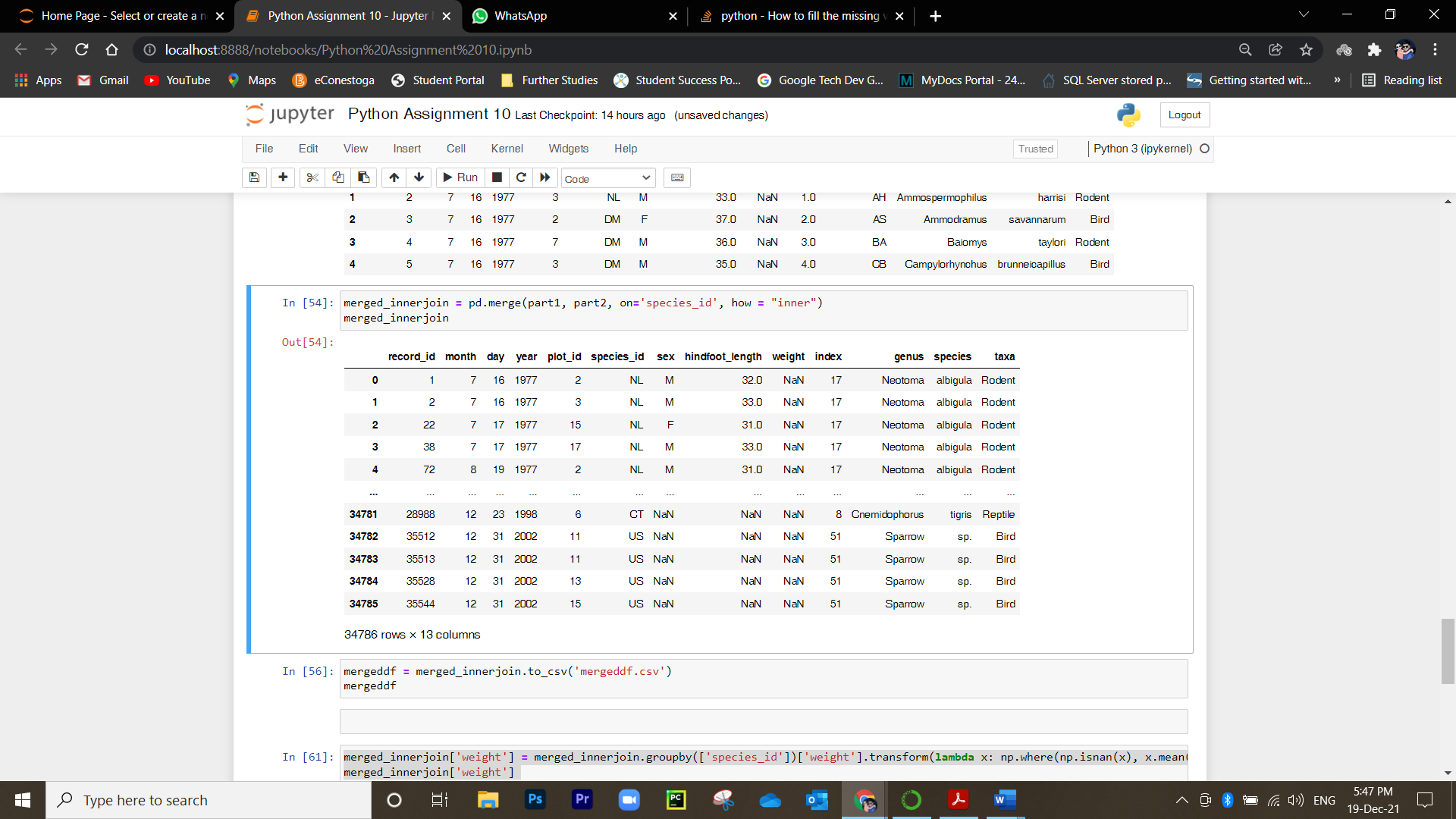


*#Question 7*

*Use* ***merge from the pandas package to*** *make an inner join.*

merged\_innerjoin = pd.merge(part1, part2, on='species\_id', how = "inner")

merged\_innerjoin



*#Question 8*

*Fill the missing value for the column weight by the mean*

merged\_innerjoin.fillna(merged\_innerjoin.mean(),inplace =True)

merged\_innerjoin.head()

Table

Description automatically generated

*9. Plot the distribution of taxa by plot*

merged\_innerjoin[['taxa','plot\_id']].groupby(["plot\_id"]).nunique().plot(kind = 'bar')

plt.legend(loc = 'best',bbox\_to\_anchor=(0.4, 1.2))

plt.show()

Graphical user interface, text, application

Description automatically generated

*10. Plot the distribution of taxa by sex by plot*

merged\_innerjoin['sex'].isna().sum()

*merged\_innerjoin['sex'].value\_counts()*

*merged\_innerjoin['sex'] = merged\_innerjoin['sex'].fillna('M')*

group\_tax\_sex = merged\_innerjoin [['plot\_id','taxa','sex']].groupby(["plot\_id", "sex"]).nunique()

tax\_by\_sex = group\_tax\_sex .pivot\_table(values='taxa',columns = 'sex', index = 'plot\_id')

tax\_by\_sex.plot(kind = 'bar')

plt.legend(loc = 'best',bbox\_to\_anchor=(0.2, 1.2))

plt.show()

Graphical user interface, text, application

Description automatically generated

*11. Analyse your findings.*

:- In question 9 we can see that half of the plots are at 4 and others are at 3 and low.

After analysing the sex count in taxa by plot we found that

, we found that in all the plot the ratio of male is greater than the female.

There is a variation in the ration of male whereas female is constant for all the plots.