**SpeakX Data Science Assignment**

**Name: Pullela Harish Chowdary Reg.No: 12002666**

**University: Lovely Professional University**

**Assignment:** Predicting Customer Churn in a Telecommunications Company

**Objective:** The primary objective of this project is to develop a predictive model that can identify

customers at risk of churning, enabling the company to take proactive measures to retain

them.

**Implementation starts here**

**Tasks:**

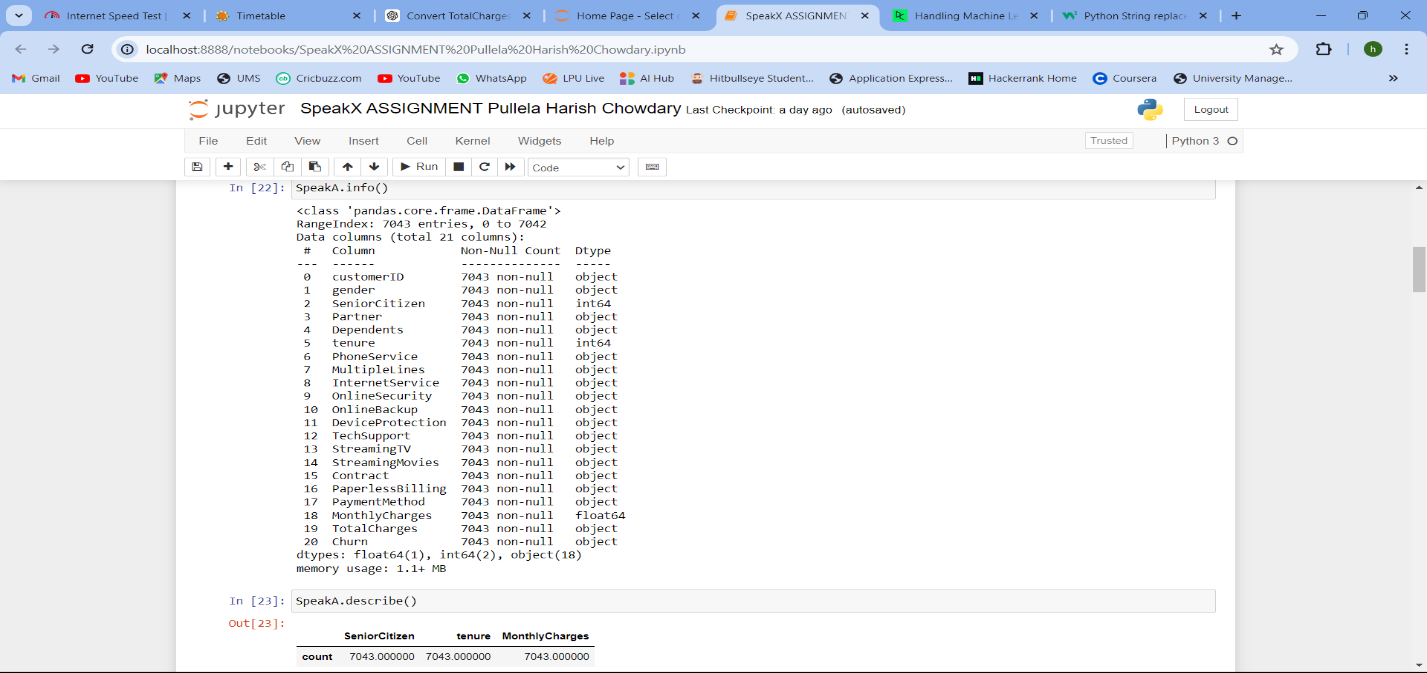
1. **Data Collection and Preprocessing:**

**Data Set:** [**https://www.kaggle.com/datasets/blastchar/telco-customer-churn**](https://www.kaggle.com/datasets/blastchar/telco-customer-churn)

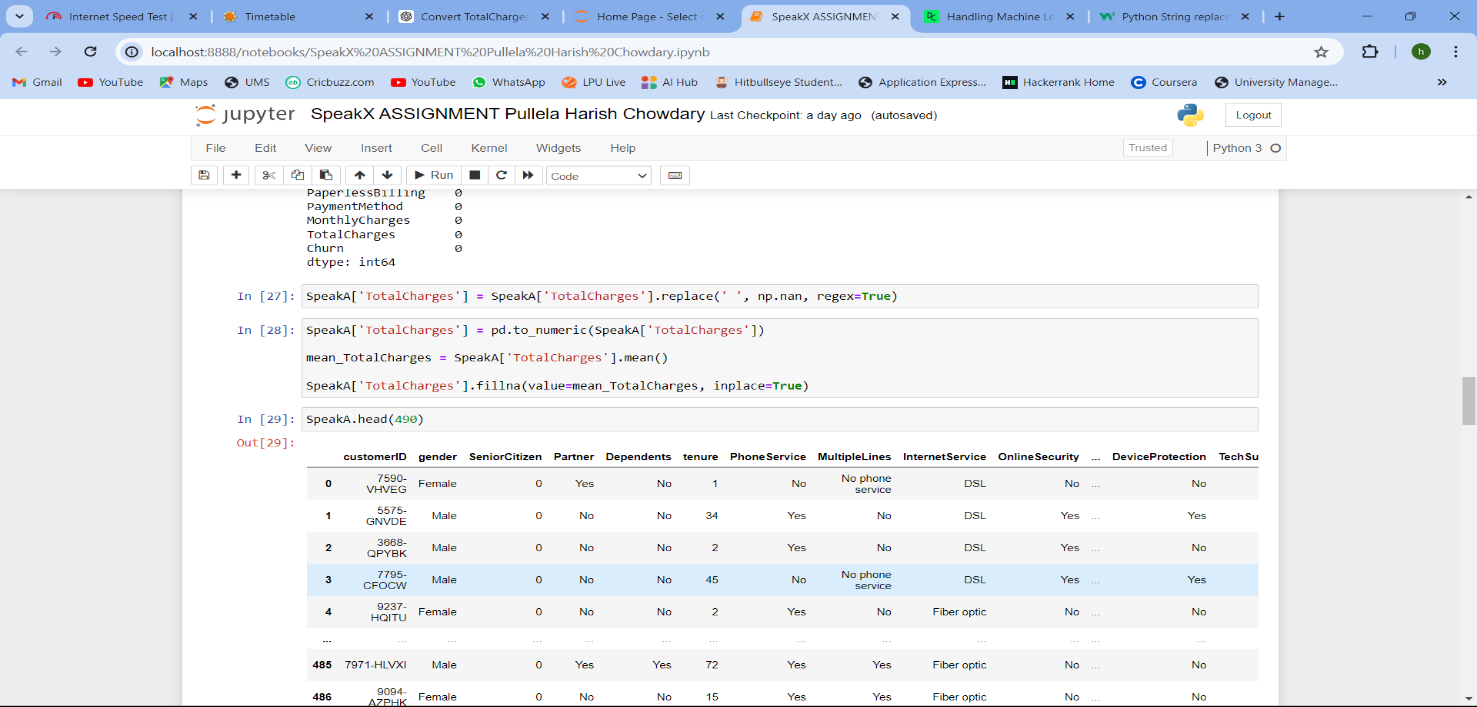
**Steps for importing and loading the dataset in Python:**

* Imported numpy, pandas, matplotlib, seaborn
* For loading the dataset, the data set should be in the current directory, and using pandas reading the CSV file is done
* I explored **shape –** shows how many columns and rows are there**, columns** – shows column headers**, info –** shows columns with their datatype**, describe –** shows columns of continuous variables and explores count, mean, std, min, max, etc**, is.na –** shows whether any null values in the data set are in a Boolean form. The below screenshots are representations for above mentioned implementation

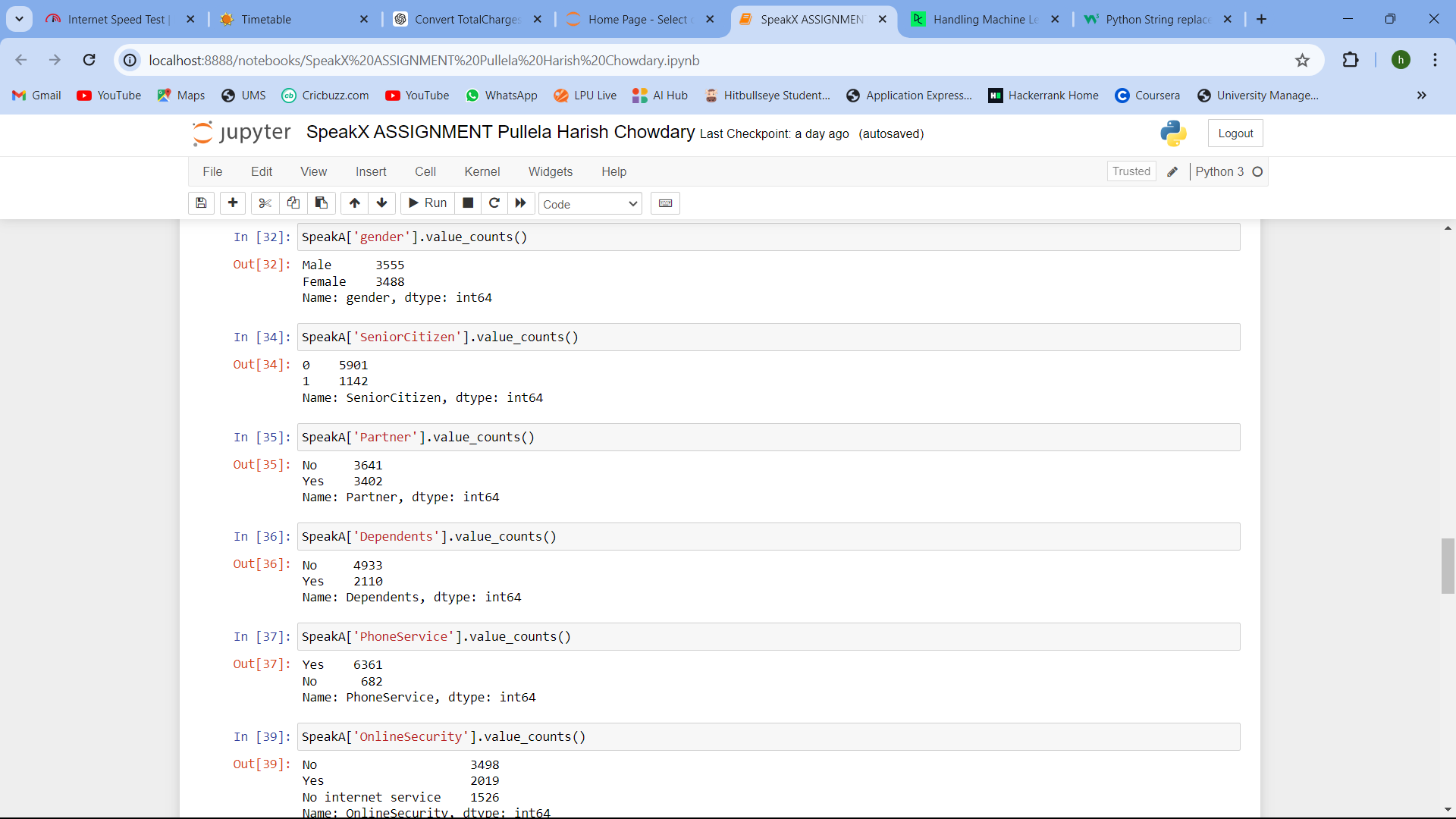


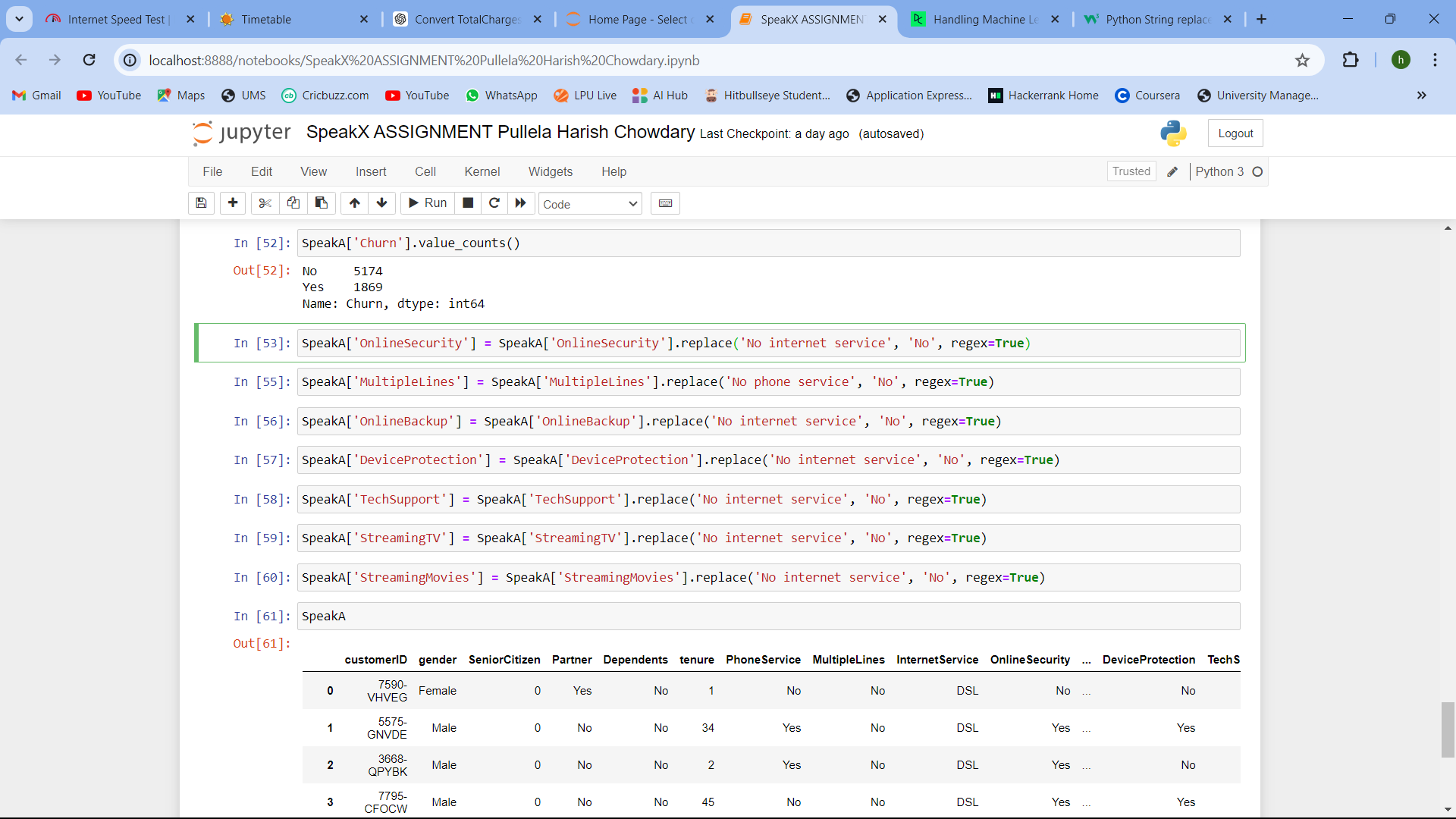


* I observed that column **TotalCharges** has some blank rows that I have replaced with nan. And then replace those nan values with the **mean of TotalCharges**



* Now I explored categorical variables in the dataset where with function **.value\_counts()** it will print the count of values in a particular column of categories that column has. I have observed columns like **‘OnlineSecurity’, ‘MultipleLines’, ‘OnlineBackup’, ‘DeviceProtection’, ‘TechSupport’, ‘StreamingTV’, and ‘StreamingMovies’** have some categories like **No Internet service and No phone service** along with Yes and No. So I replaced No Internet service and No Phone service with No because maybe they belong to the No category. It will have binary values for those columns

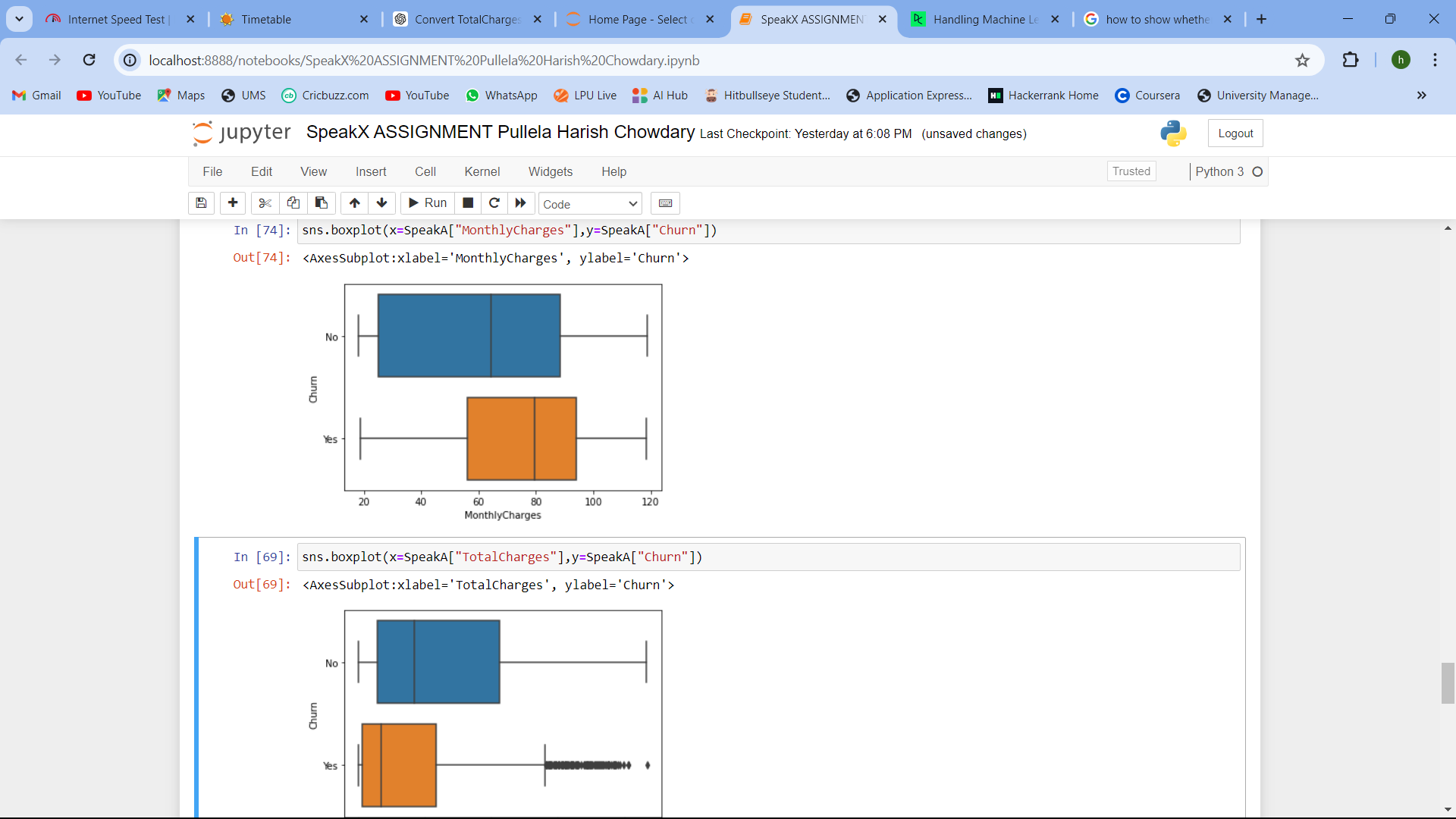




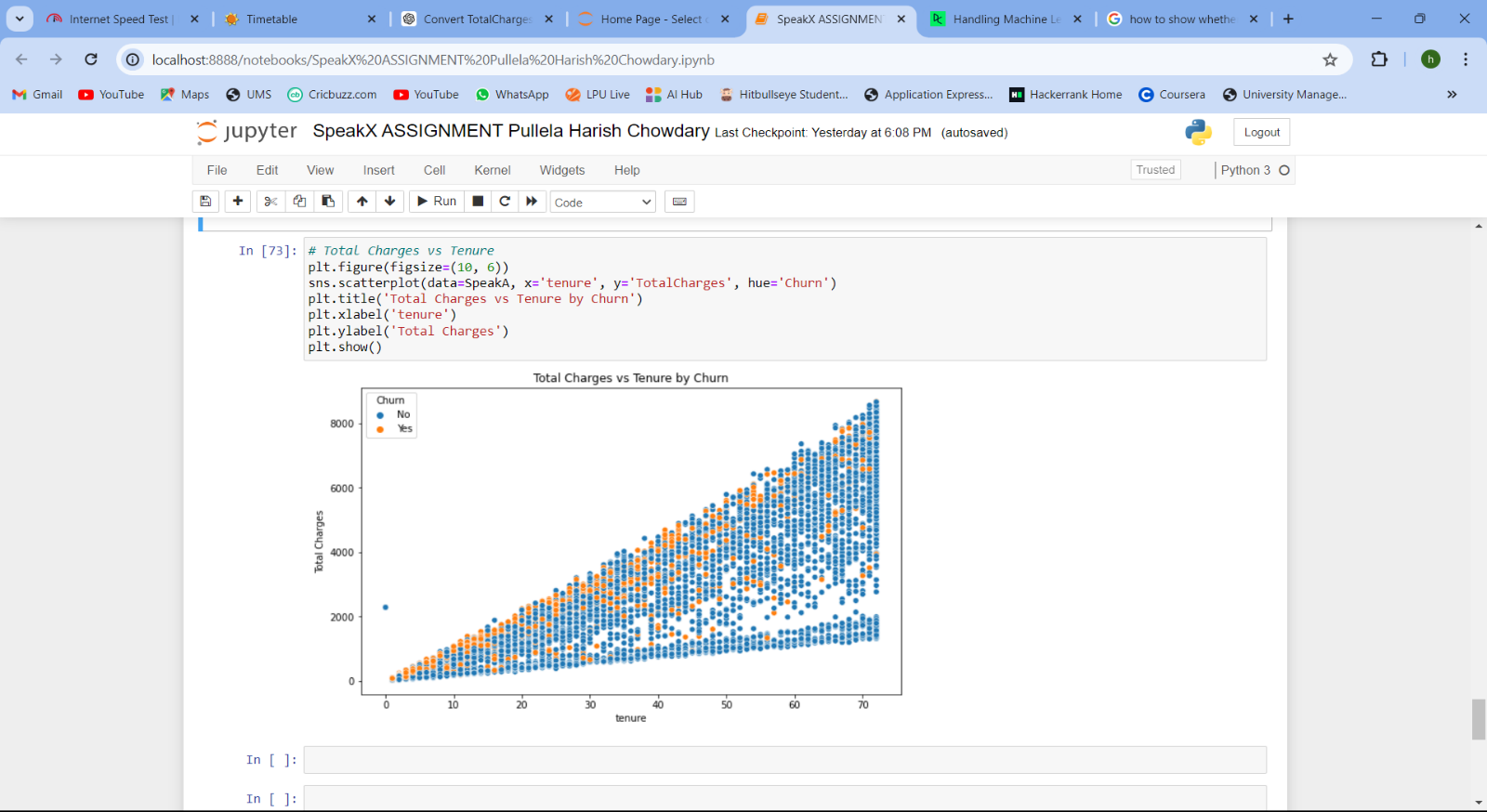
Above mentioned steps are for this task - **Preprocess the data to handle missing values, encode categorical variables, and prepare it for analysis. Encode categorical variable at prediction time I going to do**

1. **Exploratory Data Analysis (EDA):**

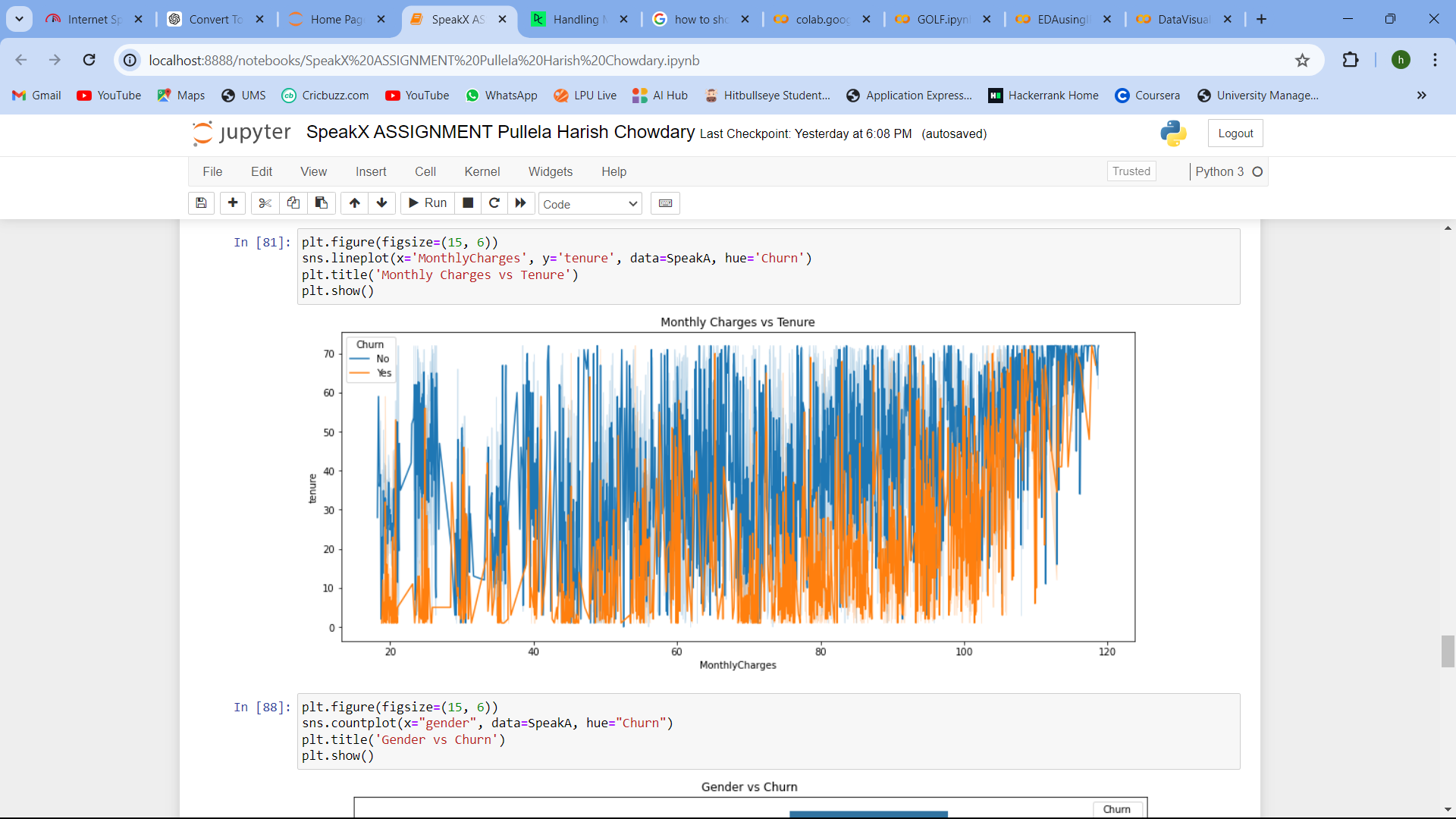
Perform EDA on the dataset to understand customer behavior and factors influencing churn. Visualize key findings using appropriate graphs and charts.

1. **Utilizing Box plot for showing outliers of Churn by tenure, TotalCharges, MontlyCharges**

**Conclusion:** I have observed that Yes of Churn means Churn users have outliers in tenure and TotalCharges

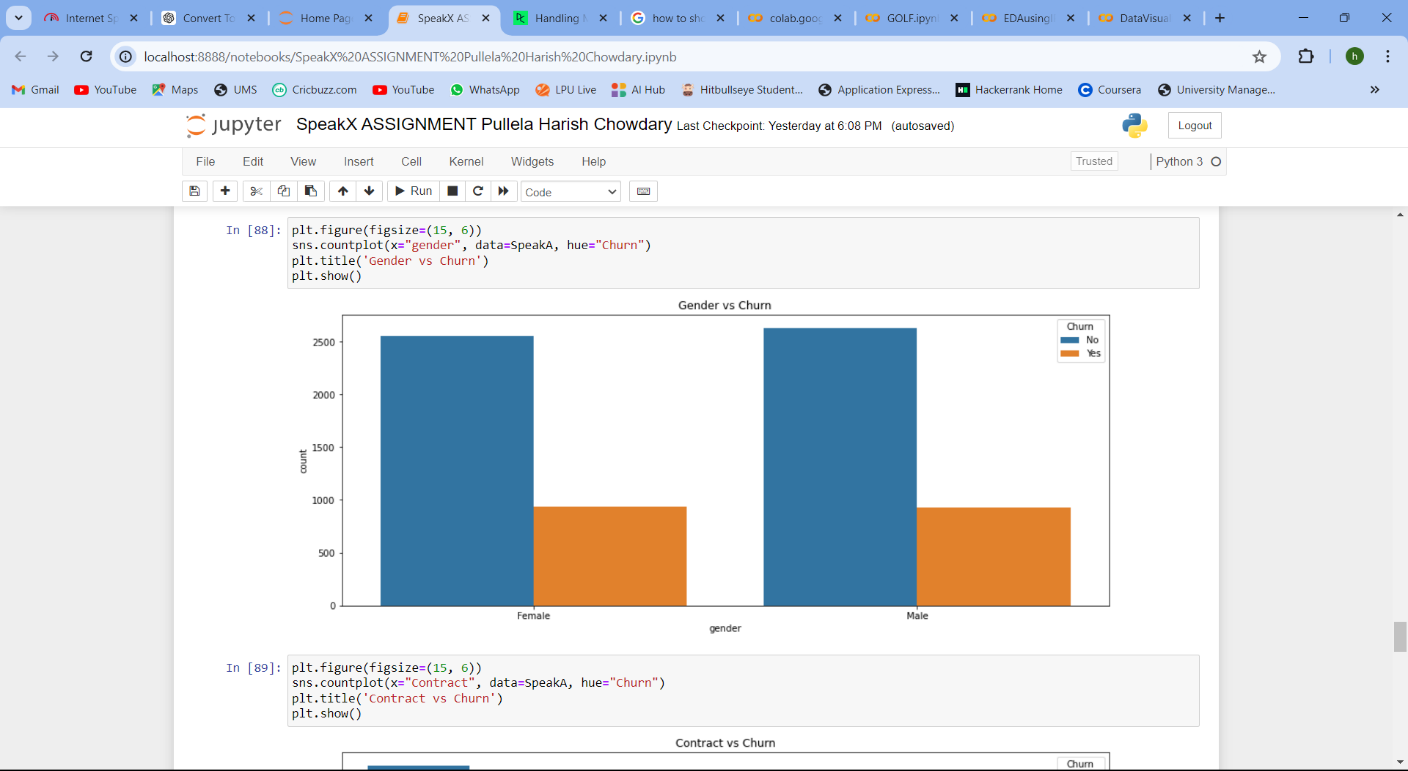
1. **Utilizing a scatter plot to show relationship between tenure and TotalCharges based on Churn**

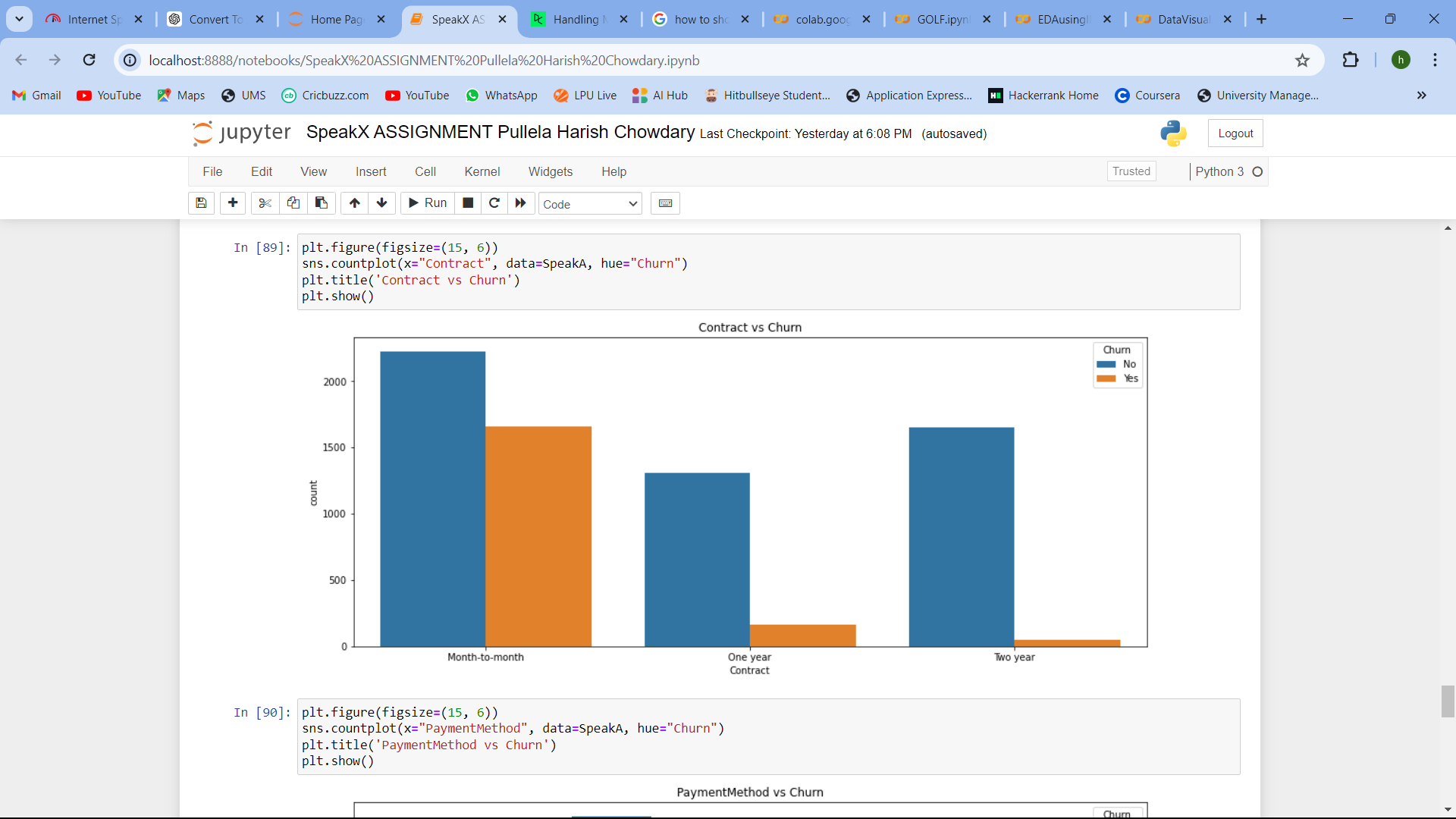
**Conclusion:** There is a positive linear relation between tenure and TotalCharges

1. **Utilizing a Line chart to show Monthly Charges vs Tenure by Churn**

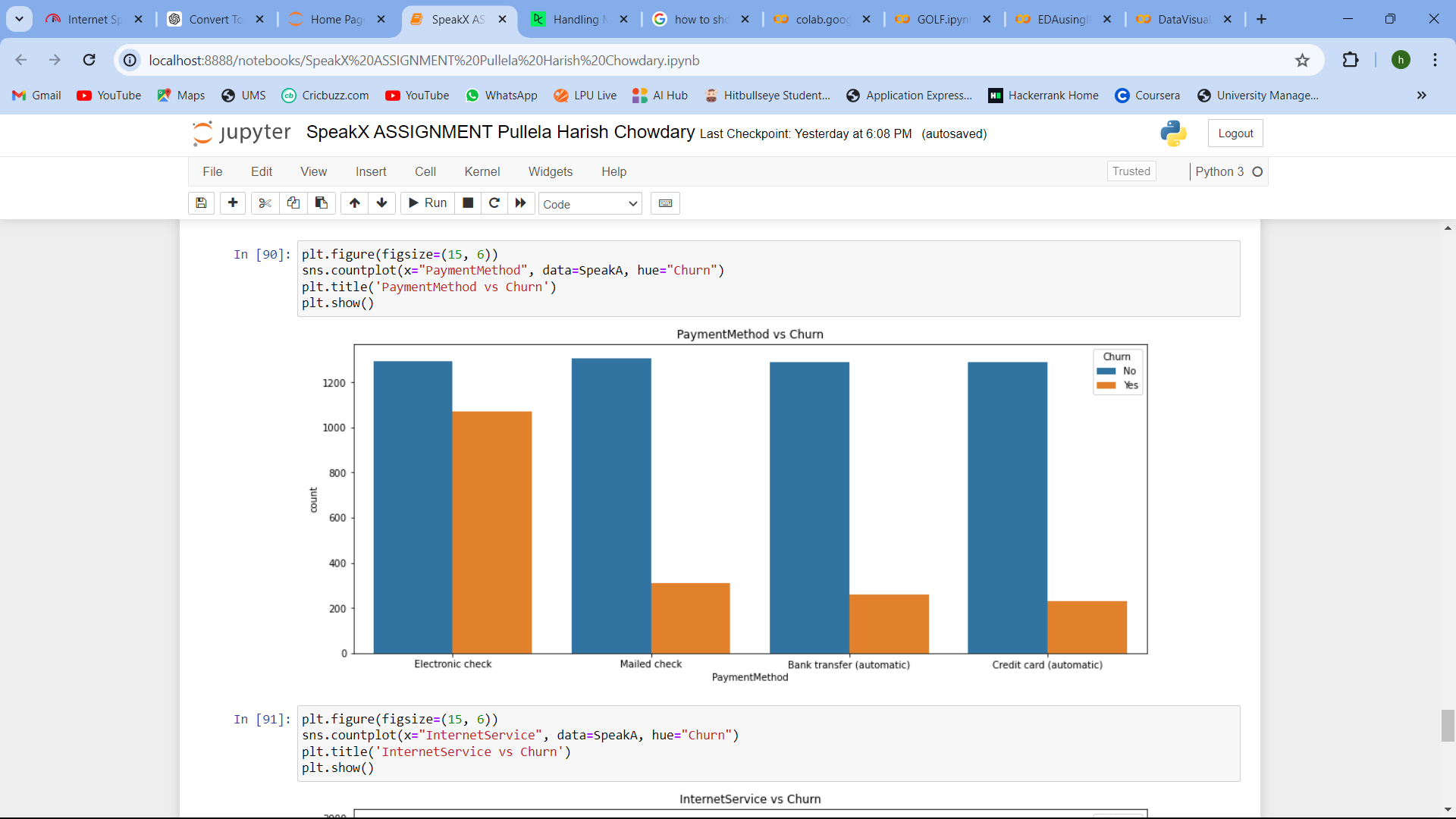
**Conclusion:** (No) of Churn has more tenure and monthly charges

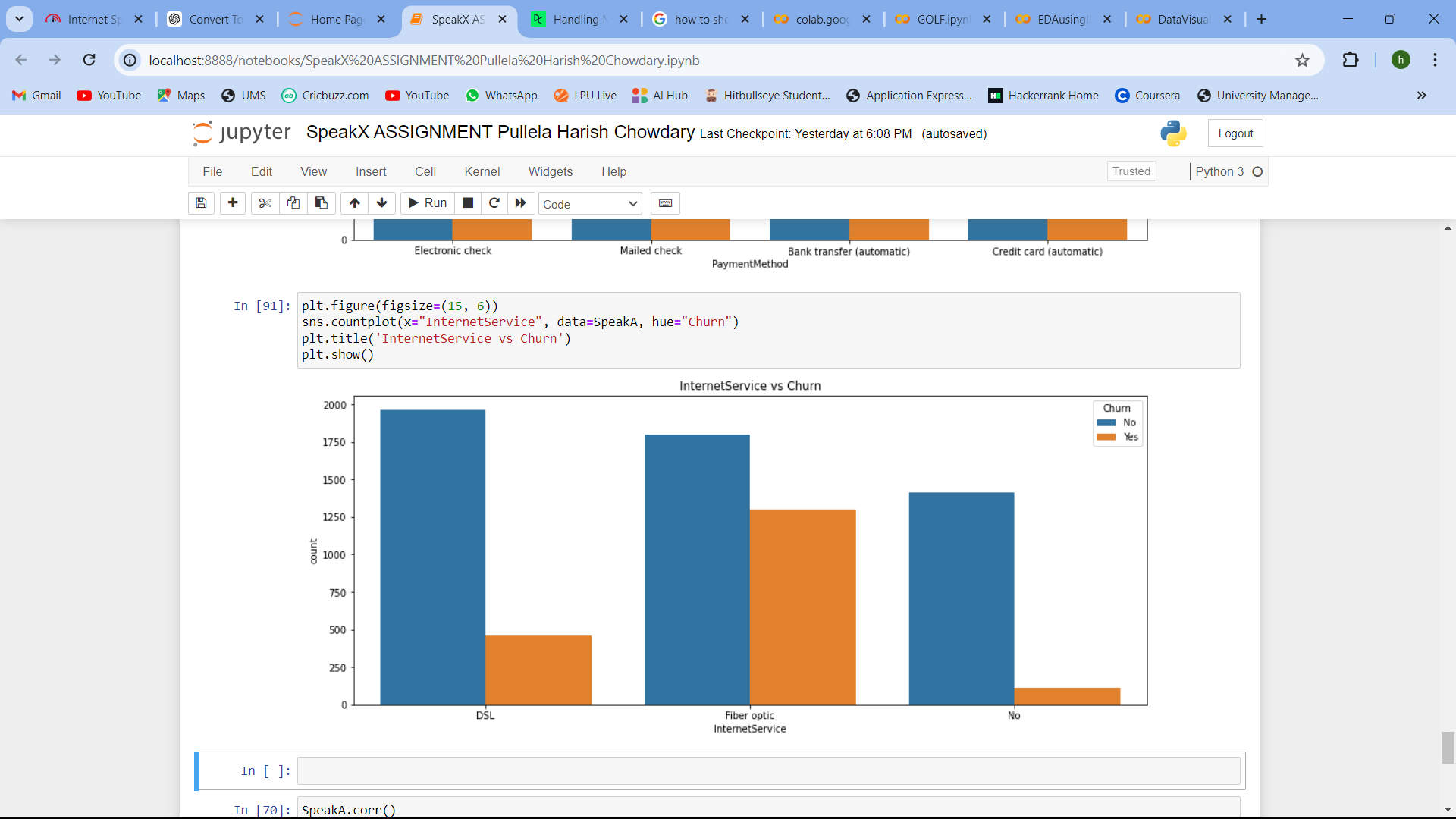
1. **Utilizing a count plot or showing a bar plot where the count of values for gender, internet service, contract, and payment method based on the Churn column**



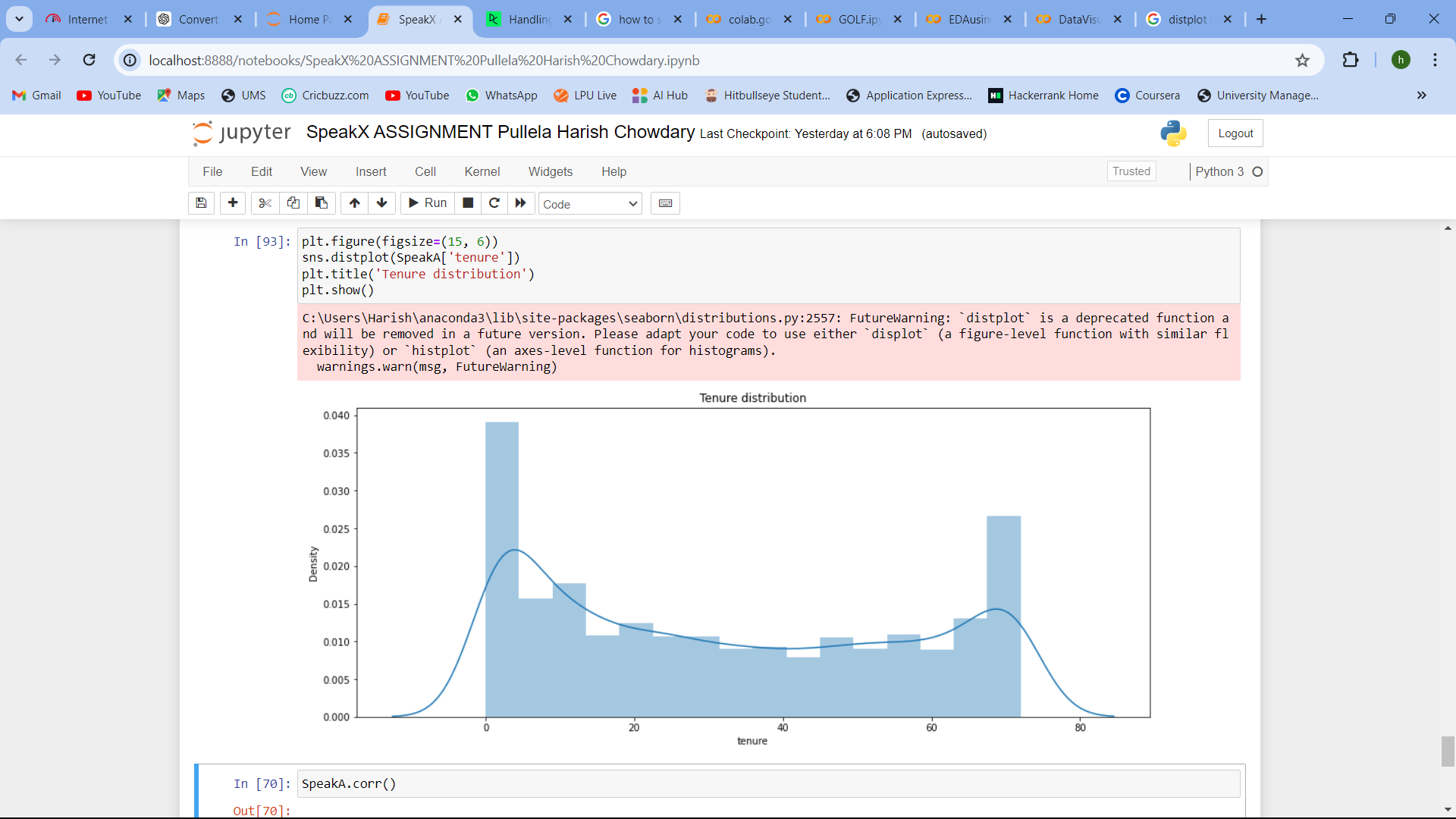
**Conclusion:** Almost similar distribution between female and male count and Based on Churn as No category has more value distribution between females and males. No Churn users have more distribution

**Conclusion:** Month-to-month contract users are more than one year contract and two-year. Based on Churn too month-to-month users have more users



**Conclusion:** Most users have opted electronic check payment option. (No) category churn users have equally opted for all payment method options.

**Conclusion:** No category of Churn users has DSL internet service. When combined with the yes and no categories of churn users then DSL and Fiber optic internet services have equal distributions.

1. **Utilizing a distplot graph to show the distribution of the tenure column**

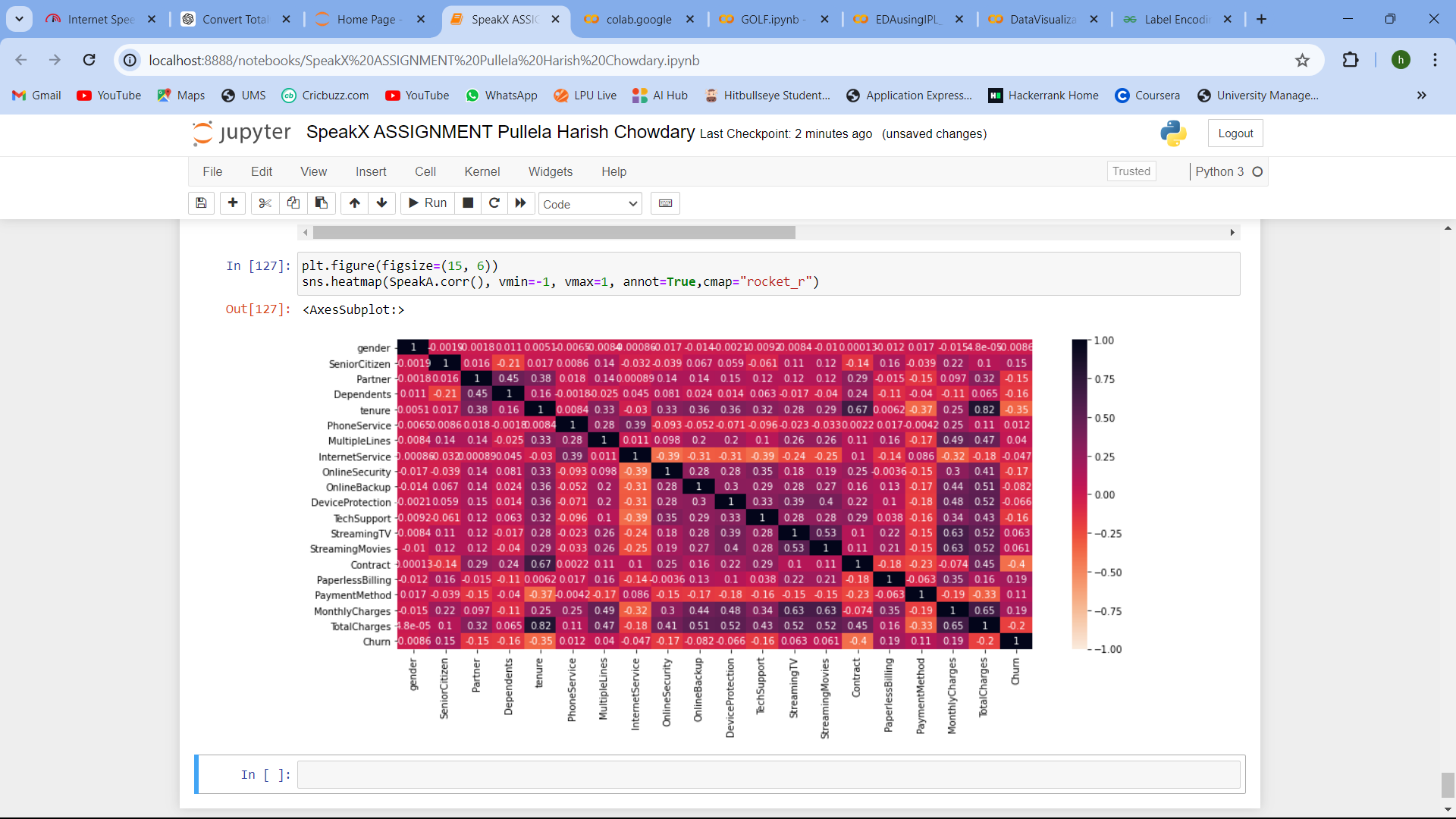
**Conclusion:** 0-20 density has more value of tenure like 0-20 has more tenure distribution

1. **Utilizing a tree map graph to show the correlation between attributes or variables**

**Before encoding of categorical variable means only between continuous variables:**



**After encoding of categorical variables:**

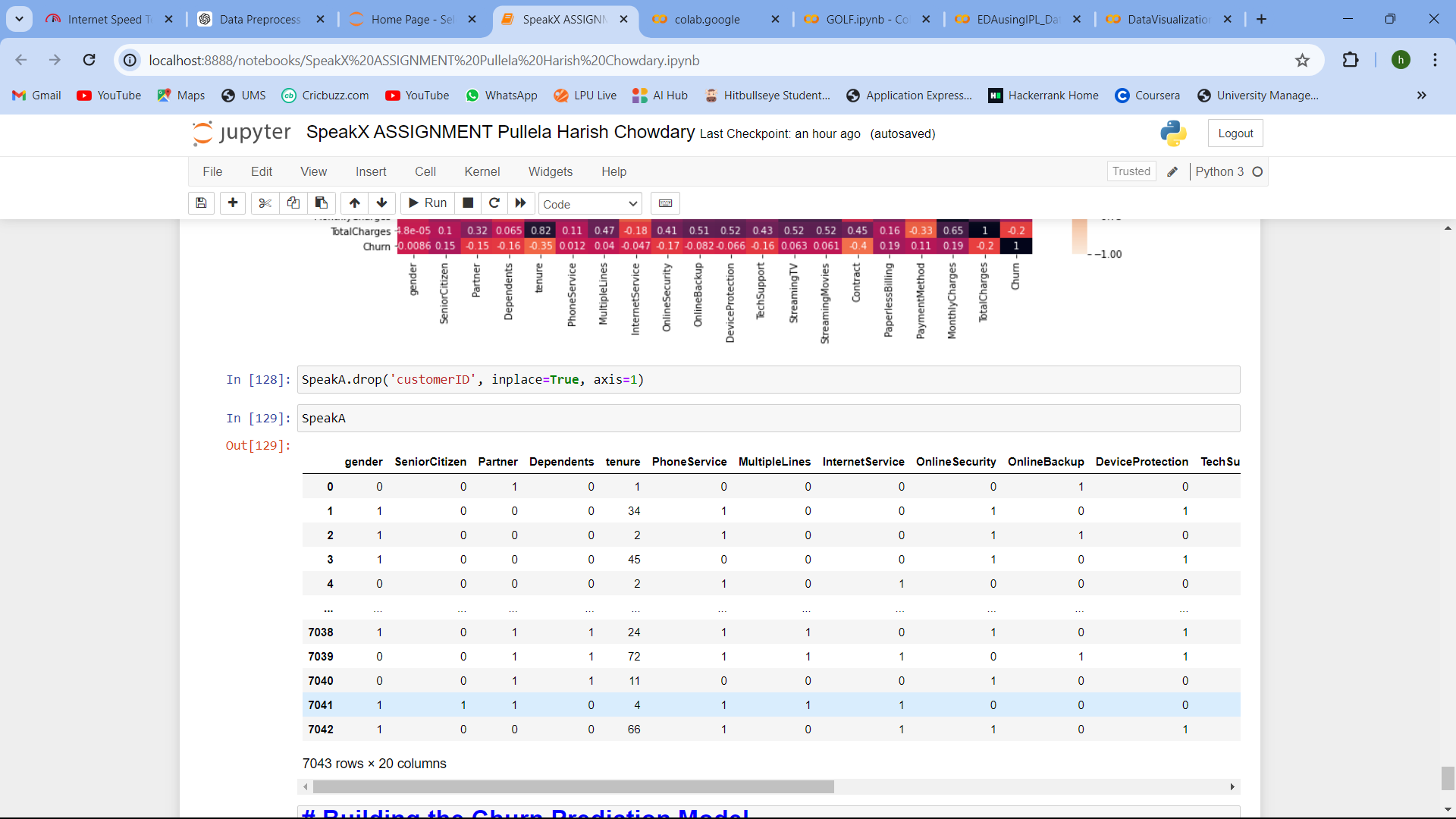


1. **Feature Engineering:**

Create relevant features that can help in predicting churn.

* Firstly, label encoding is part of feature engineering as features involve preprocessing and cleaning too. So before building a machine learning model label encoding for categorical variables is necessary here in feature engineering label encoding has been done.



* For selecting relevant features from the dataset given we don’t require a customer ID column and remaining all are useful. So, I dropped the customer ID column

1. **Building the Churn Prediction Model:**

Choose and implement a machine learning algorithm for churn prediction. You can consider algorithms like logistic regression, random forests, gradient boosting, or any other suitable models. Train and fine-tune the models using the dataset.

**Step1: Data collection and data importing**

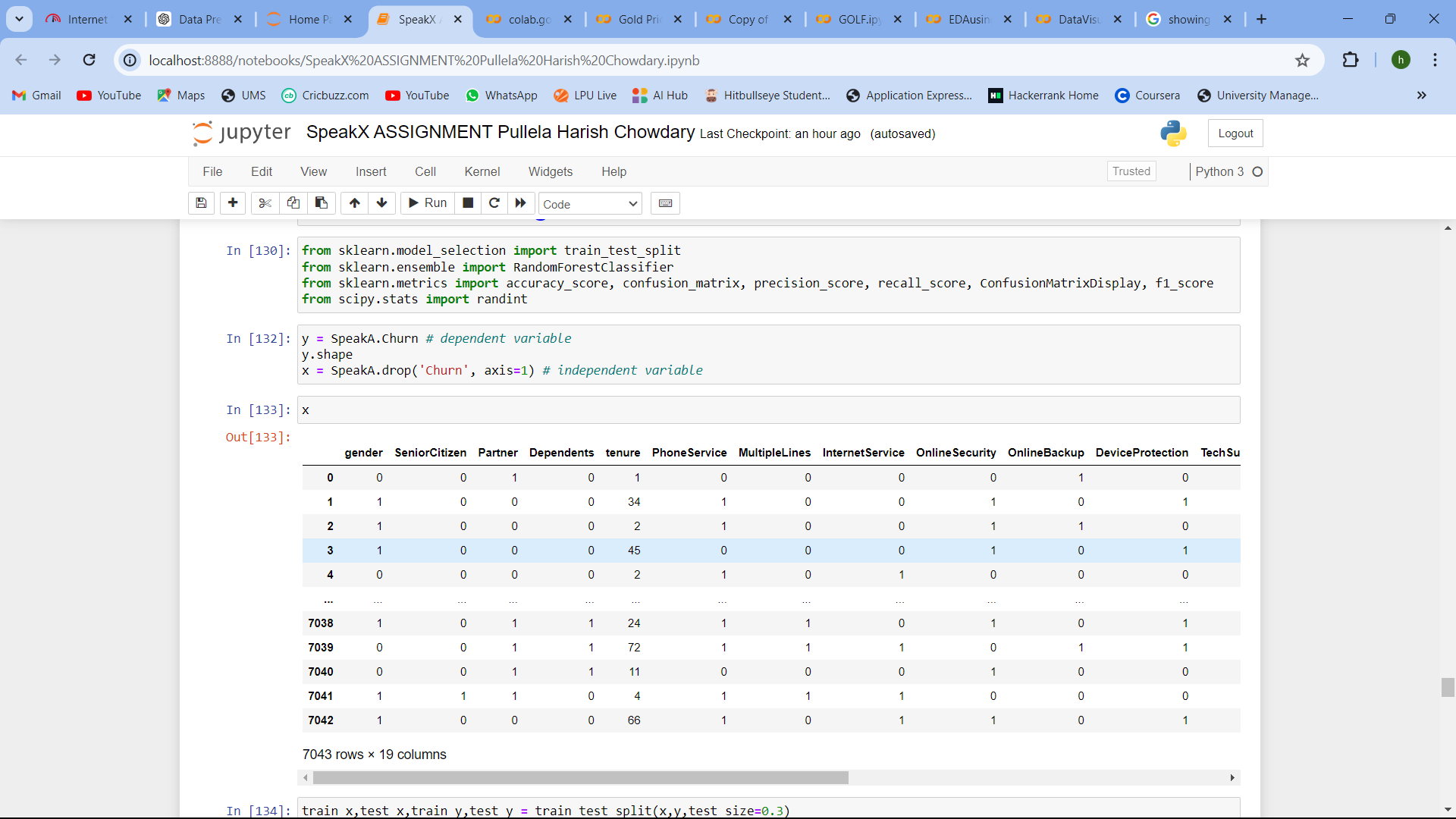
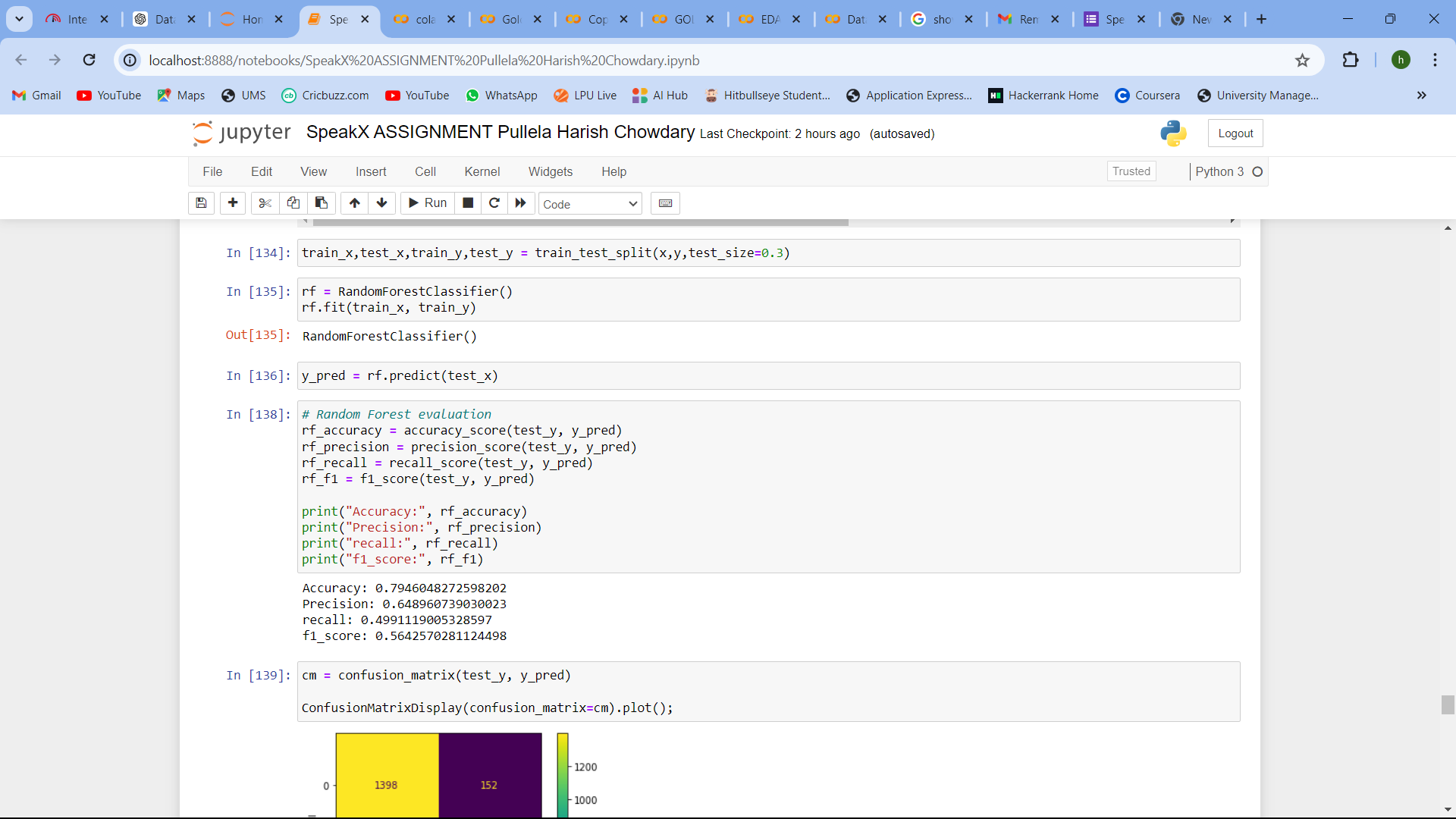
* Already before we imported

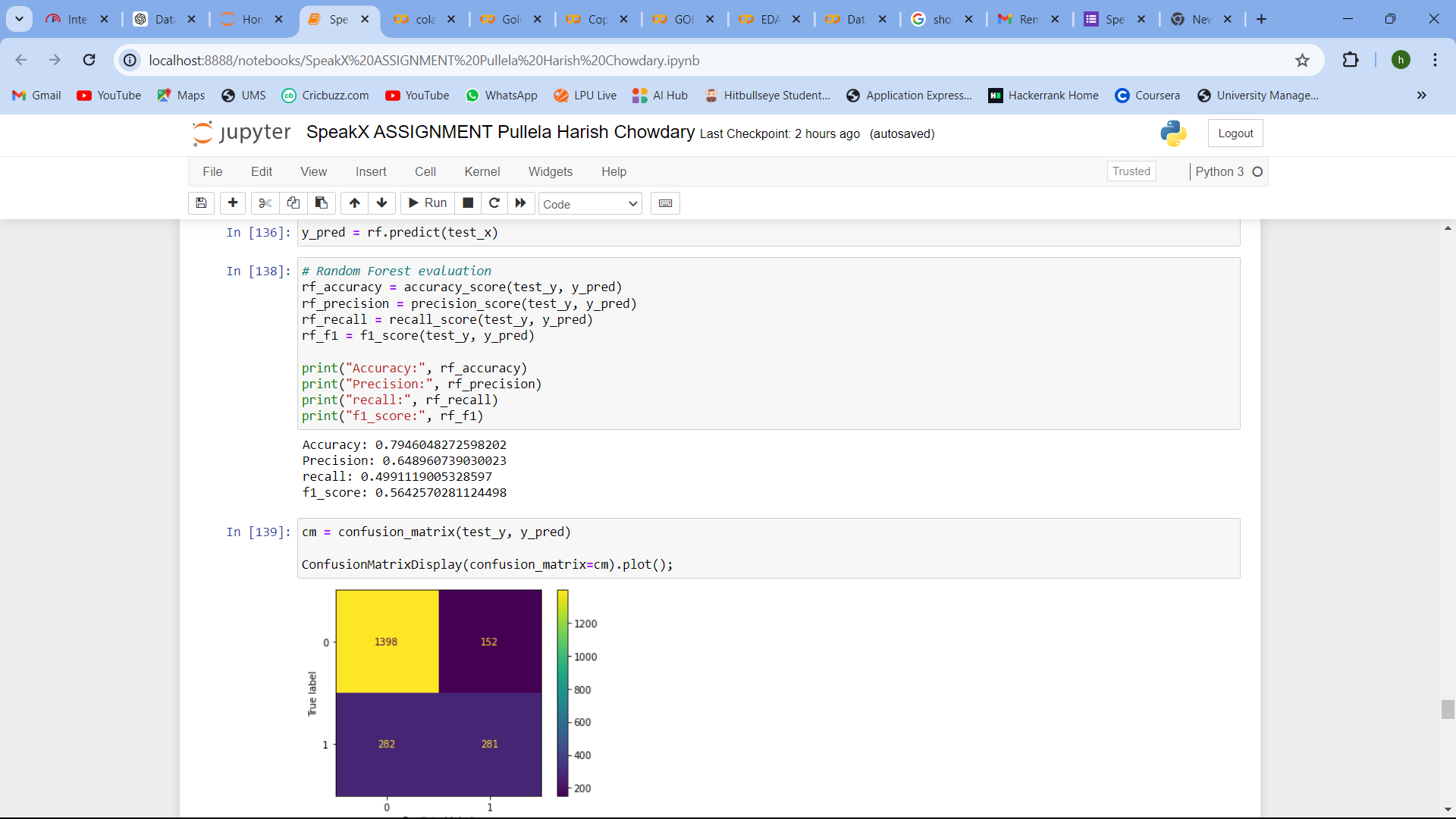
**Step 2: Data Pre-processing and Cleaning**

**Step 3: Feature Engineering**

* Available on previous pages

**Step 4: Machine learning implementation with Model Evaluation**

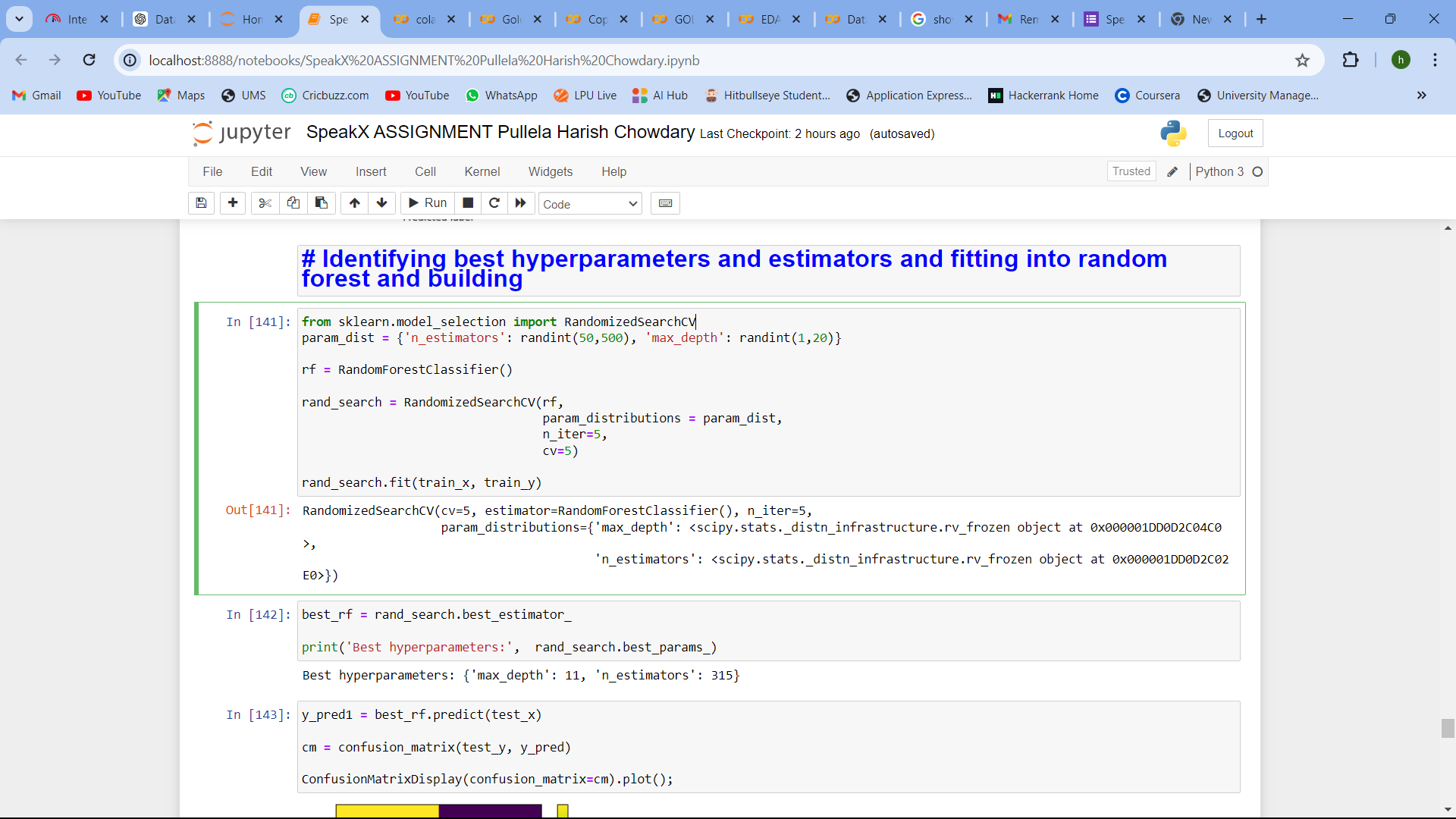
1. **Random Forest Algorithm:**
2. **Importing necessary libraries and giving dependent and independent variable**
3. **Splitting the dataset into train and test datasets & Fit using a random classifier:**
4. **Predicting using y\_pred and Model Evaluation:**

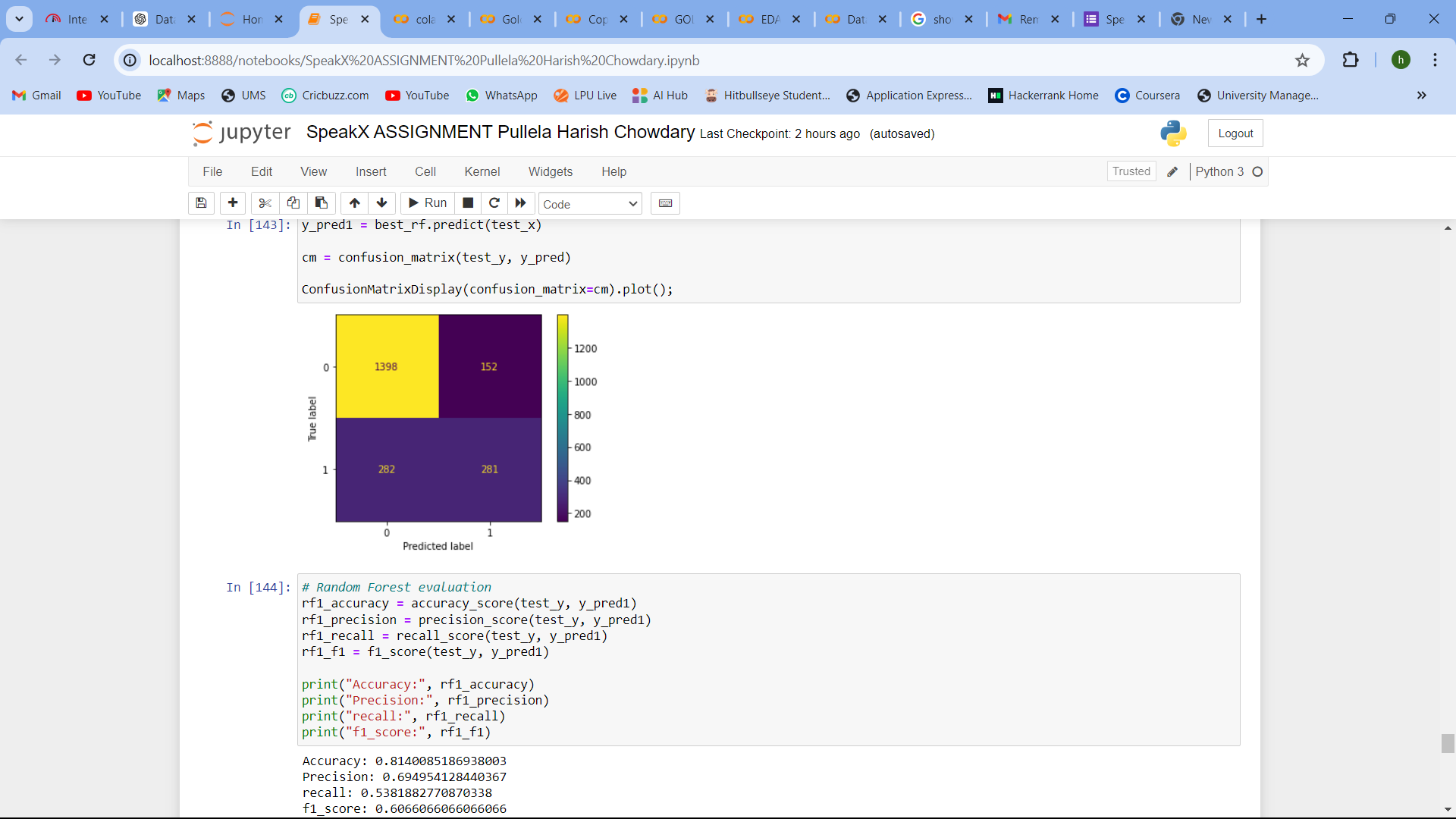


Conclusion: Random Forest gave an accuracy of 79% with a precision of 64%, recall of 49%, and F1 score of 56%

Method 2: Identifying best hyperparameters and estimators and fitting into random forest and building using

RandomizedSearchCV

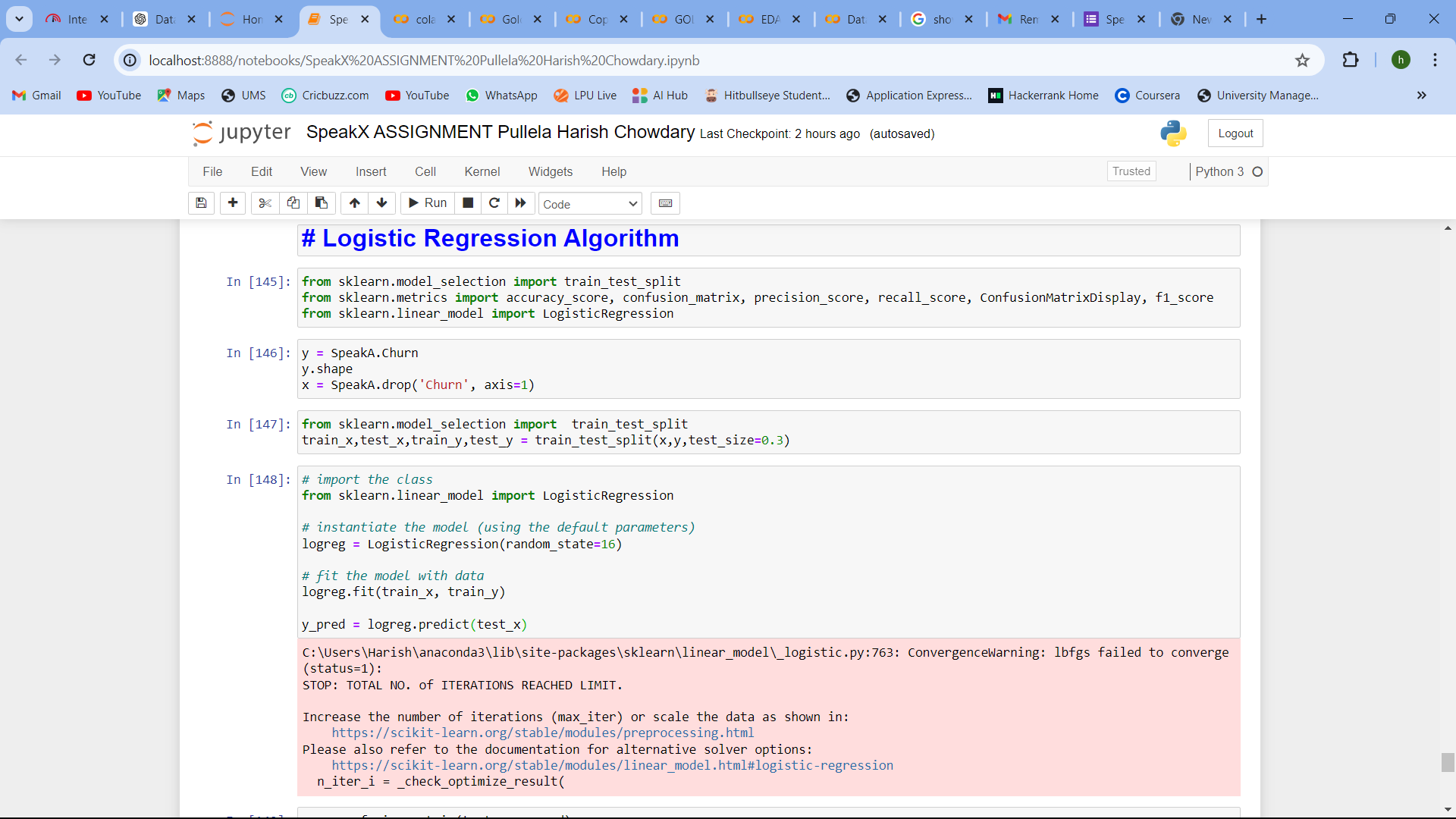


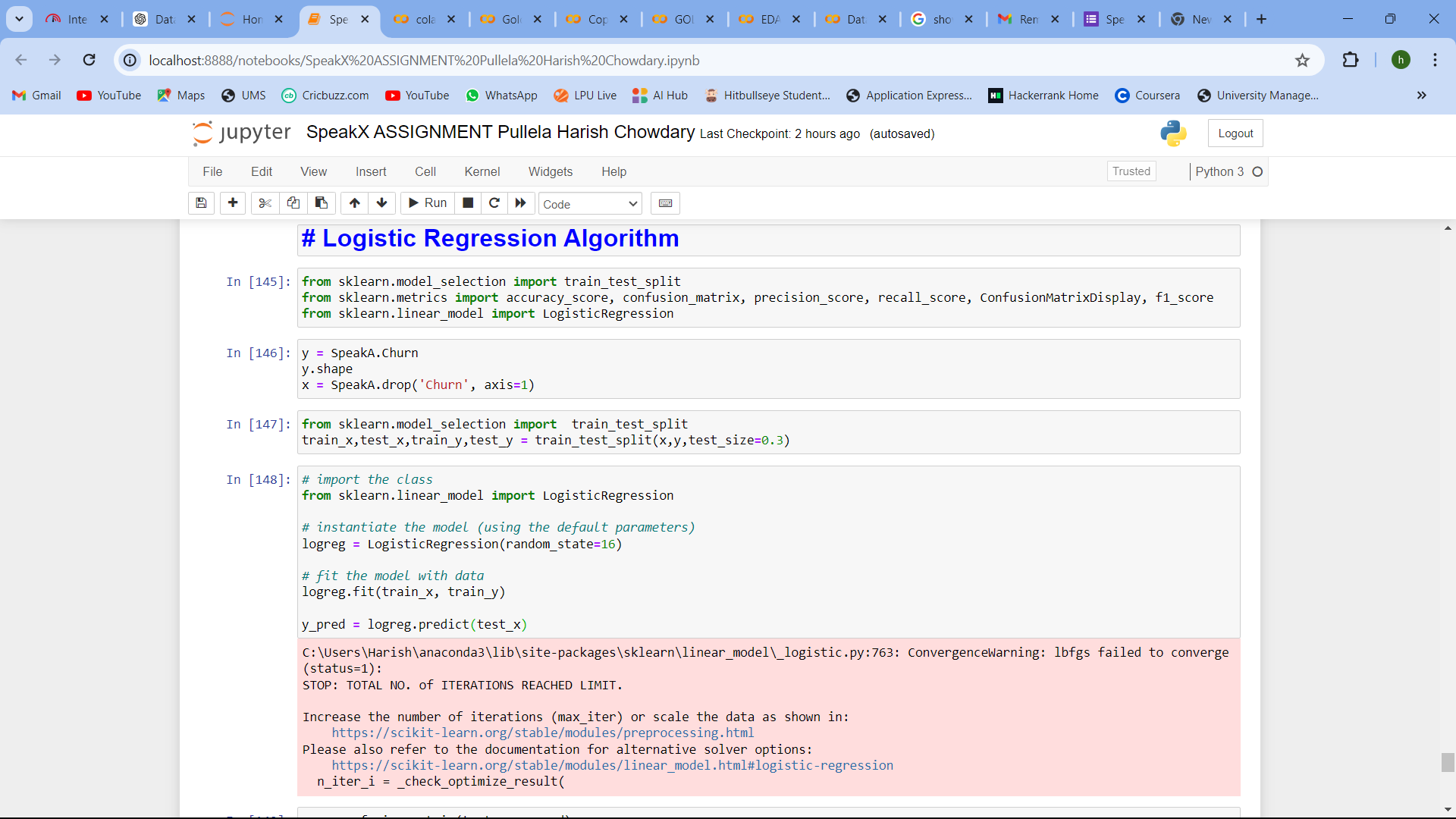


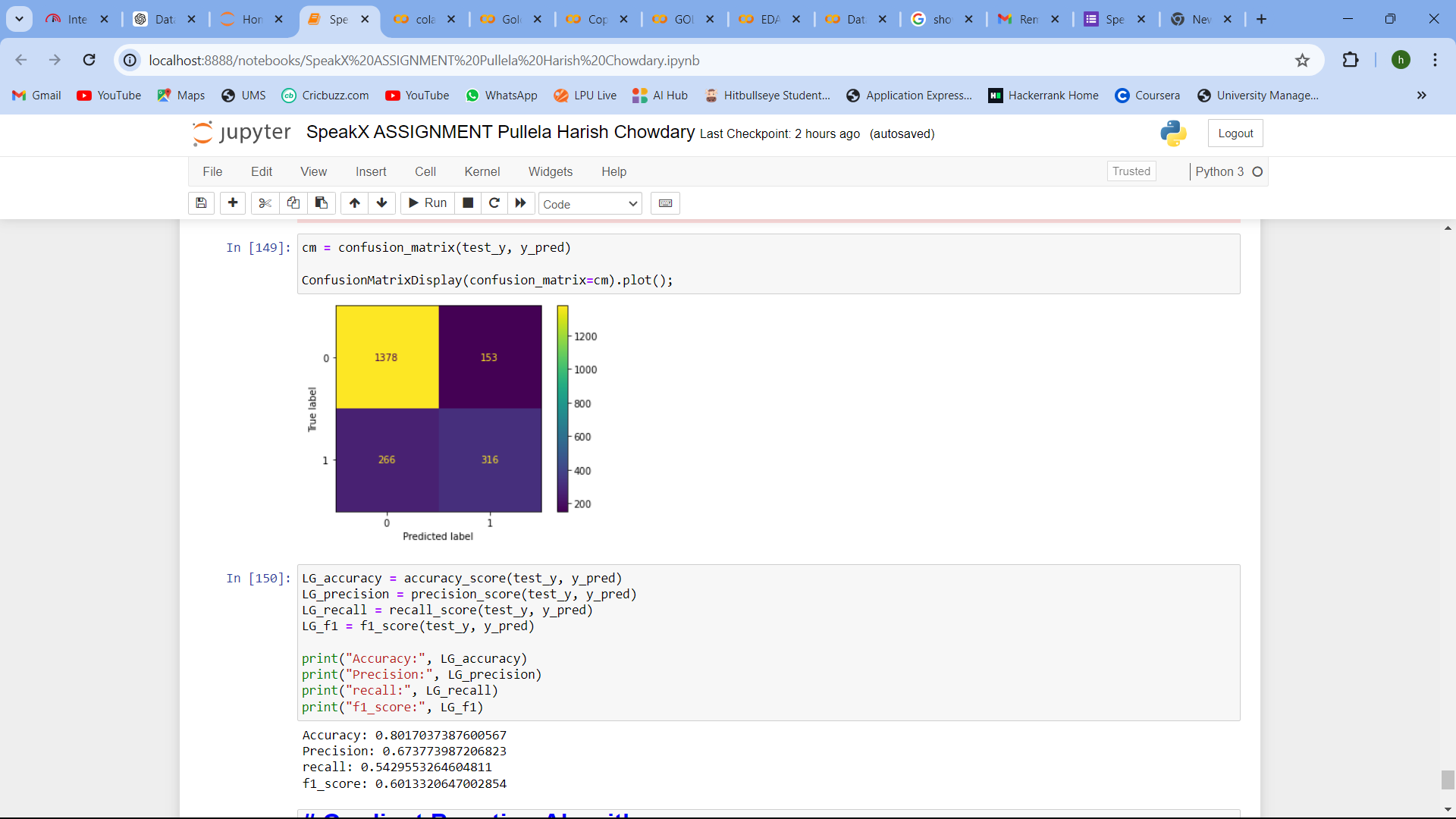
Output: Best hyperparameters: {'max\_depth': 11, 'n\_estimators': 315}

Conclusion: Random Forest with best hyperparameter and estimator gave an accuracy of 81% with a precision of 69%, recall of 53%, and F1 score of 60%

1. **Logistic Regression Algorithm:**
2. **Importing necessary libraries and giving dependent and independent variable**

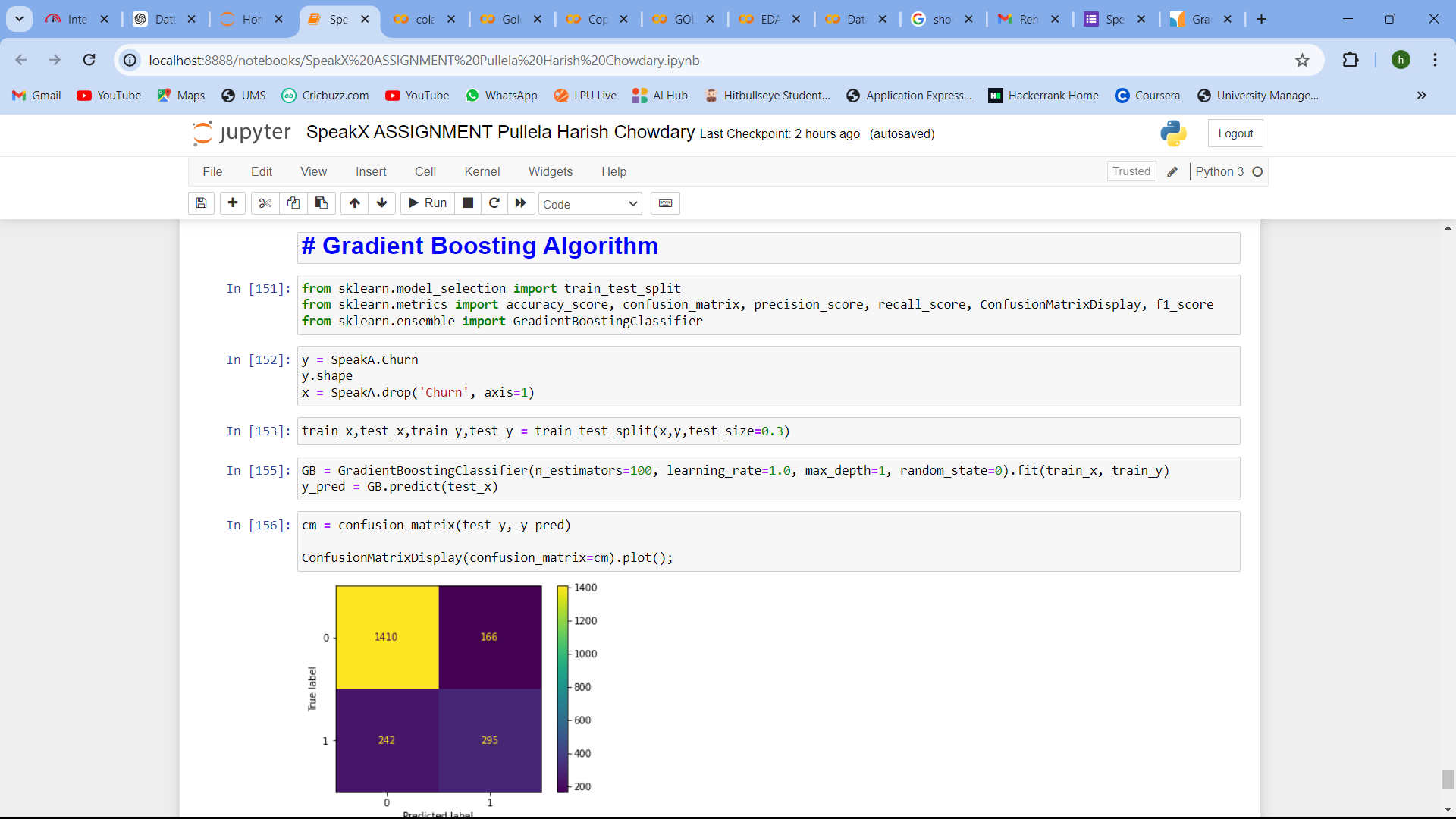


1. **Splitting the dataset into train and test datasets & Fit using a random classifier:**
2. **Predicting using y\_pred and Model Evaluation:**

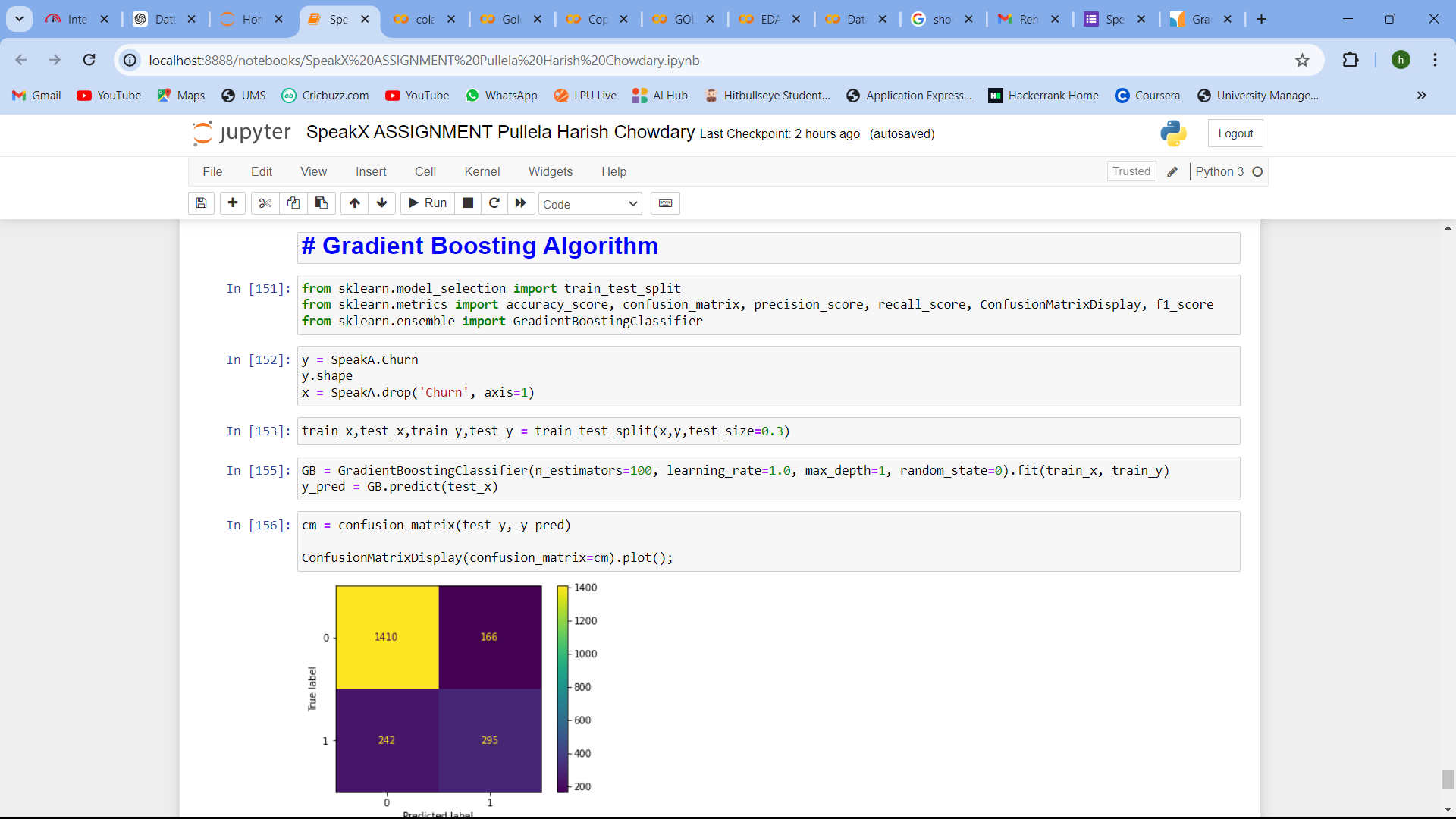


Conclusion: Logistic Regression gave an accuracy of 80% with a precision of 67%, recall of 54%, and F1 score of 60%

1. **Gradient Boosting Algorithm:**
2. **Importing necessary libraries and giving dependent and independent variable**



1. **Splitting the dataset into train and test datasets & Fit using a random classifier:**



1. **Predicting using y\_pred and Model Evaluation:**



Conclusion: Gradient Boosting gave an accuracy of 80% with a precision of 63%, recall of 54%, and F1 score of 59%

Challenges faced during the project:

1. Identifying blank rows and filling those values
2. As there are more categorical variables encoding separately checking count values is made as difficult as individually checking for every categorical variable.