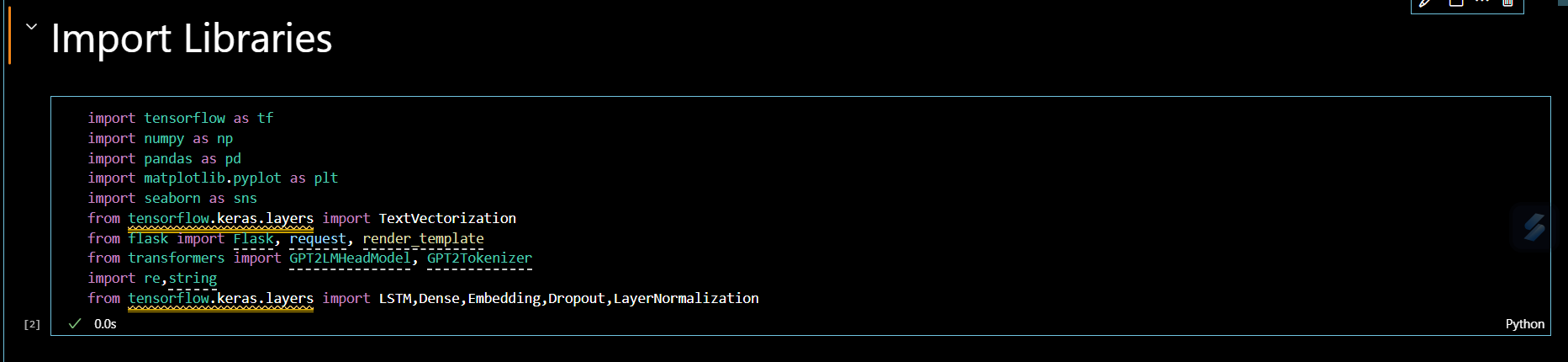
**CHATBOT**

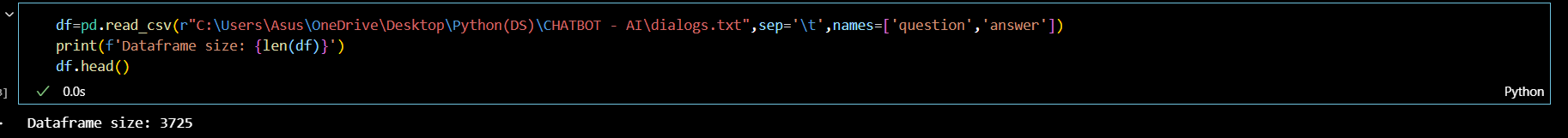
**Step 1: Importing Libraries**

(Download/Import required libraries)



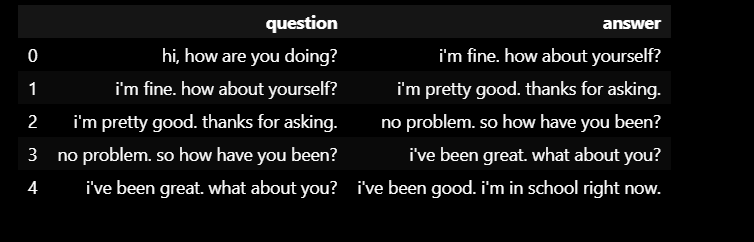
**Step 2: Data Loading**

(Read CSV “dialogs.txt”)



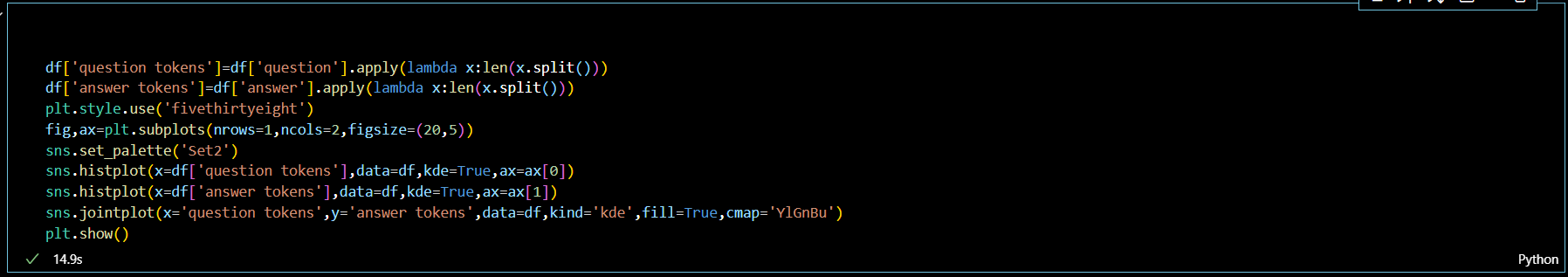
“name” is a parameter used in Panda’s library to specify the column names *[question], [answer]*;

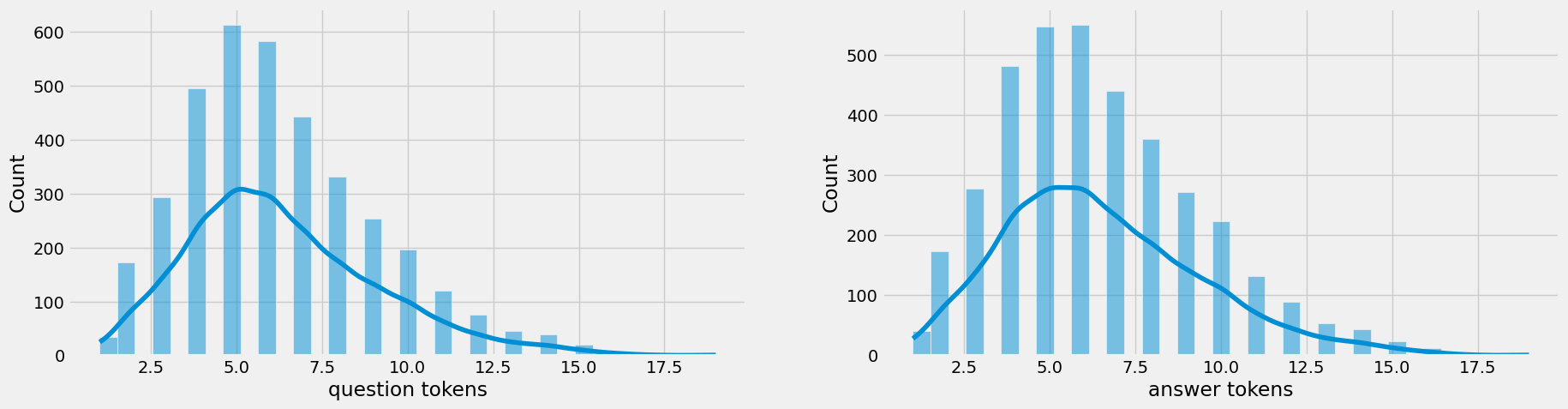
**Output:**

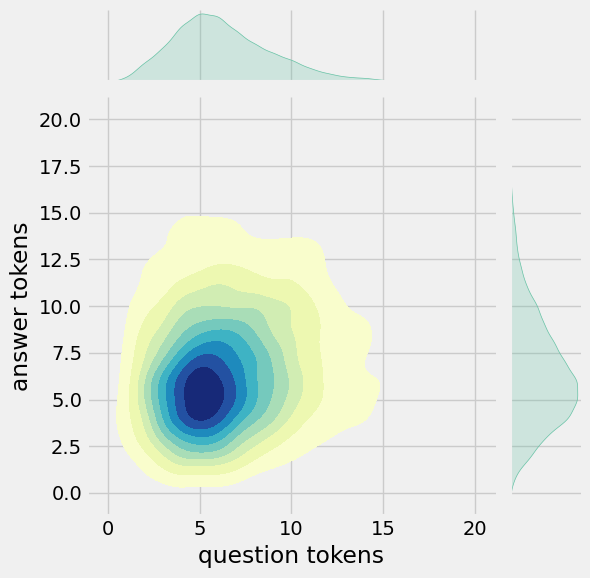


**Step 3: Data Preprocessing**

Data preprocessing is for preparing data for analysis or machine learning tasks.







Data Preprocessing includes these steps :

* *Tokenization*
* *Data Visualization*
* *Styling and Plotting*
* *Displaying the Plots*
* **Text Cleaning**

Text cleaning is a preprocessing step that aims to standardize and normalize the text to make it more suitable for Natural Language Processing (NLP) tasks...

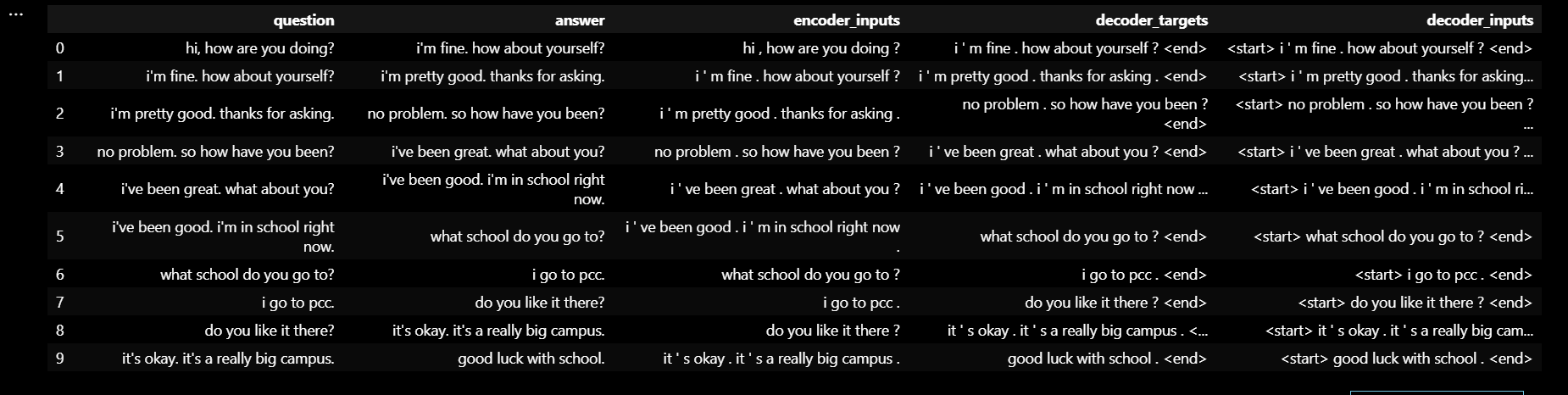
The purpose of text cleaning is to make the text more consistent, removing any irregular characters that might interfere with text processing, such as tokenization or data analysis.



In this code, we replace hyphens with spaces and convert the text to lowercase.

Adding spaces around periods(.), digits (1), commas (,), question marks(?), exclamation marks(!), dollar signs, forward slashes(/), colons(:), semicolons Etc..,

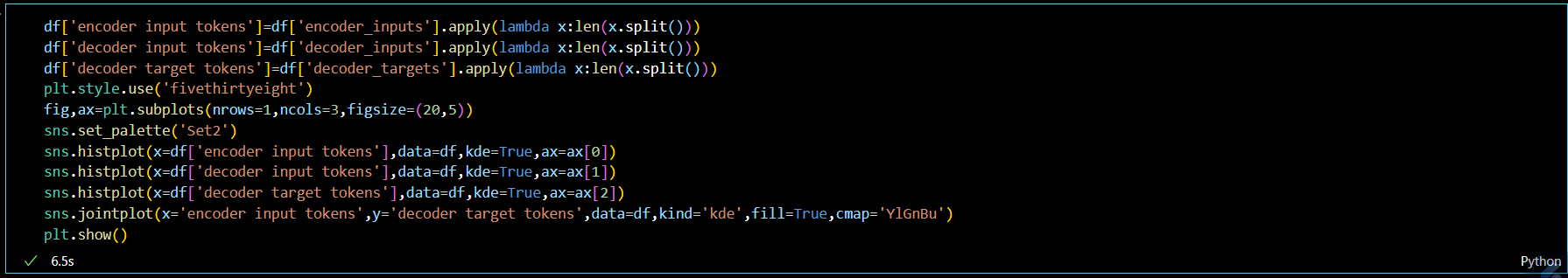
**Output:**



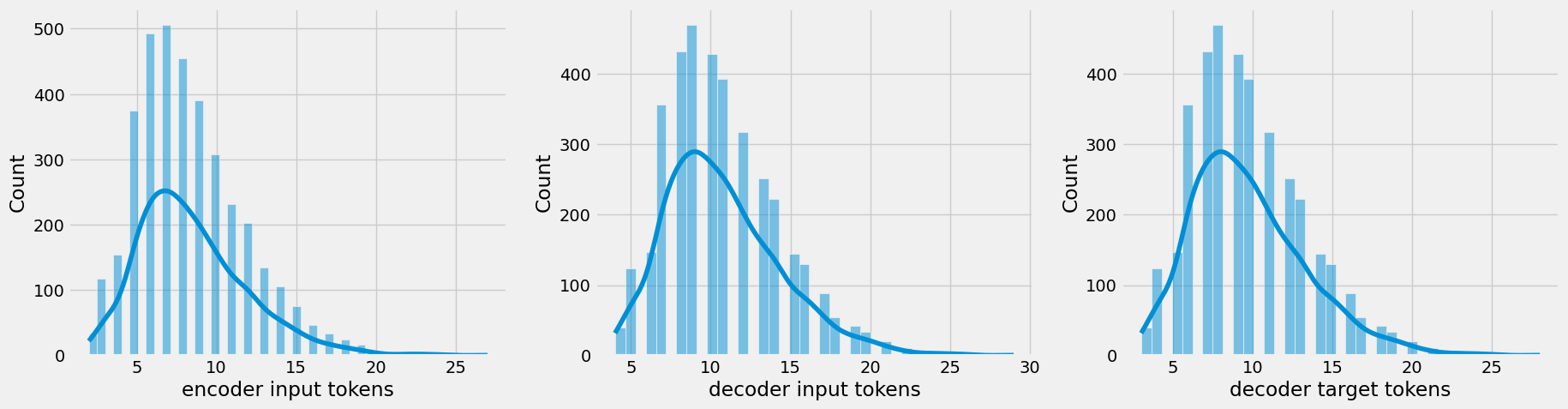
* **Data Preparation**:

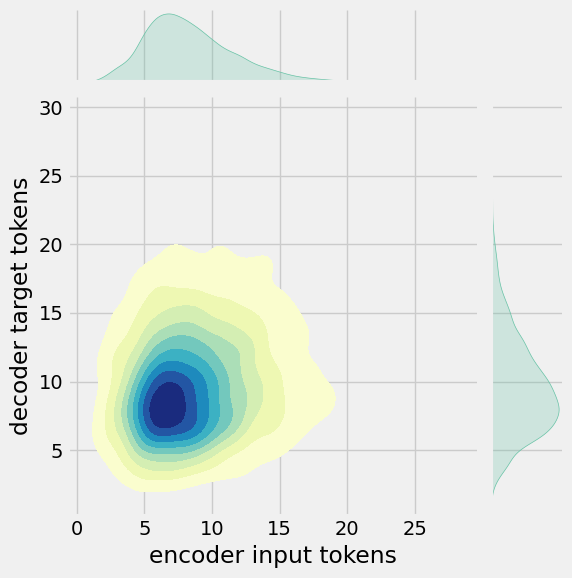
In this code,

* *df['encoder input tokens'], df['decoder input tokens'*], and *df['decoder target tokens'*] columns are created in a pandas DataFrame df.
* *df['encoder\_inputs'].apply(lambda x: len(x.split()))* calculates the number of tokens in the *'encoder\_inputs'* column for each row in the DataFrame.
* Here the style for the plots is set to *'fivethirtyeight'* using *plt.style.use('fivethirtyeight').*
* Finally, *plt.show()* displays all the plottings.



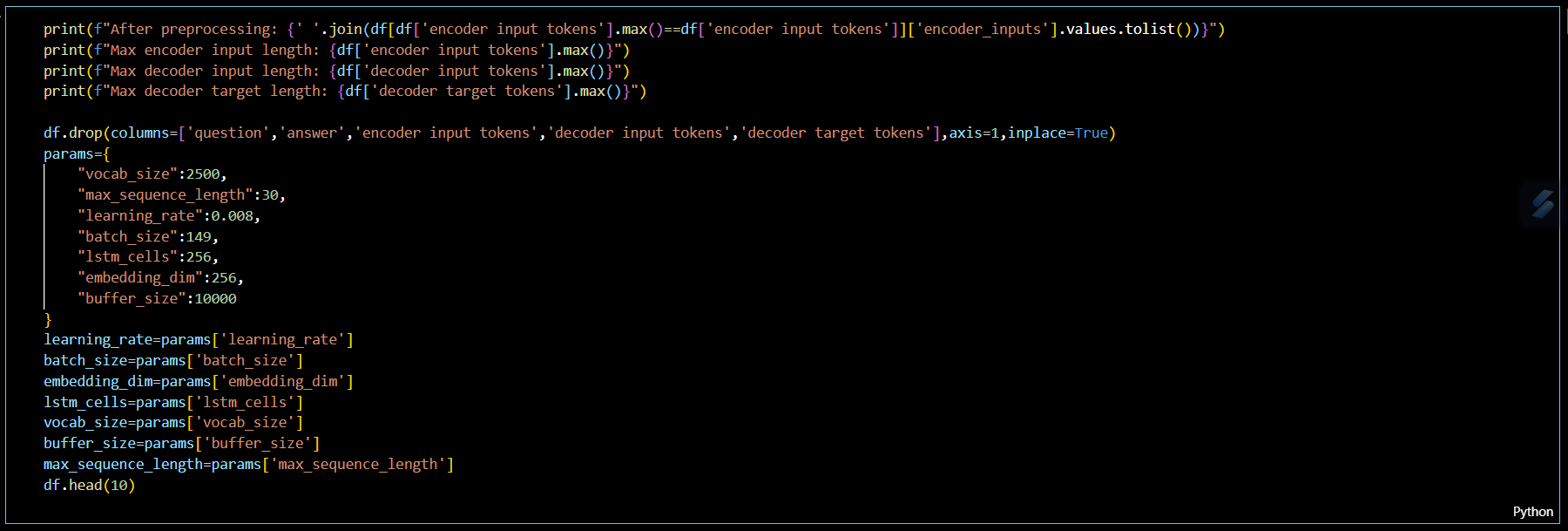
**Output:**





Further in this code, we also add some parameters for the tasks…

* The **print** statement prints a sentence formed by joining the *'encoder\_inputs'* values from the rows where the *'encoder input tokens'* column has the maximum value in the DataFrame.



* The columns 'question', 'answer', 'encoder input tokens', 'decoder input tokens', and 'decoder target tokens' are dropped using *‘df.drop’* method
* The parameters from the ‘*params’* dictionary are assigned to variables for future use. Some of the definitions of parameters from param are …
* **vocab\_size**: Represents the size of the vocabulary.
* **batch\_size**: The batch size for training the data model etc..,
* ‘*df.head(10)*’ displays the first 10 rows of the DataFram

Output:

