

**GANADIPATHY TULSI'S JAIN ENGINEERING COLLEGE**

**KANIYAMBADI - VELLORE – 632102**

**IBM - PROJECT**

**CHATBOT IN PYTHON**

***Submitted by,***

**TEAM MEMBER 1 – MADHUMITHA. B (au510821205014)**

**TEAM MEMBER 2 – SATHIYANARAYANAN.P (au510821205023)**

**TEAM MEMBER 3 – VEDHAPRIYA.P (au510821205027)**

**TEAM MEMBER 4 – SHAVITHIRA.S (au510821205025)**

**PROJECT MENTOR PROJECT EVALUATOR PROJECT SPOC**

MRS.K.DURGA PRIYANKA MRS.K.DURGA PRIYANKA MR.DURAIKUMAR

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.NO** |  | **PAGE NO** |
| **1** | **INTRODUCTION** | **3** |
|  | 1. **CHATBOT** |  |
|  | 1. **2.PURPOSE** |  |
| **2** | **PROBLEM DEFINITION** | **3** |
| **3** | **DESIGN THINKING** | **3** |
| **4** | **STEPS TO CREATE CHATBOT** | **8** |
|  | 1. **CREATING VIRTUAL ENVIRONMENT** |  |
|  | 1. **IMPORTING LIBRARIES AND DESCRIBING** |  |
|  | 1. **DATA VISUALIZATION** |  |
|  | 1. **TOKENIZATION** |  |
| **5** | **CREATING FLASK** | **20** |
|  | 1. **CREATING PYTHON CODE** |  |
|  | 1. **CREATING HTML CODE** |  |
|  | 1. **CREATING JAVASCRIPT CODE** |  |
|  | 1. **HOW TO RUN PYTHON CODE** |  |
| **6** | **OUTPUT FOR OUR CHATBOT** | **26** |
| **7** | **CONCLUSION** | **27** |

**INTRODUCTION:**

**Chatbot:**

A chatbot is a computer program that simulates human conversation with an end user. The chatbot use to help users by answering questions and guiding them when they're stuck. Though not all chatbots are equipped with artificial intelligence (AI), modern chatbots increasingly use conversation ai techniques like natural language processing (NLP) to understand the user’s questions and automate responses to them.

**PURPOSE:**

* providing exceptional customer service and offering .
* The chatbot is designed to enhance the user experience by addressing user queries and concerns promptly and providing personalized health insights related to diabetes risk.

**PROBLEM DEFINITION:**

The challenge is to create a chatbot in Python that provides exceptional customer service, answering user queries on a website or application. The objective is to deliver high-quality support to users, ensuring a positive user experience and customer satisfaction.

**DESIGN THINKING:**

Below is a high-level flowchart illustrating the key components and interactions within the system:

**1.User Interaction:**

Users interact with the chatbot through a web or app interface.

**2. Natural Language Processing :**

User queries and messages are processed by the NLP engine, which is responsible for understanding the user's intent and extracting relevant information.

**3. Customer Service Functionality:**

* If the user's query pertains to customer service, the chatbot provides assistance, answers questions, and directs users to appropriate resources or support channels.
* Responses may include FAQs, troubleshooting steps, or links to relevant help articles.

**4. Data Collection:**

The chatbot collects relevant user data, which may include age, gender, family history, lifestyle factors, and medical history, necessary for diabetes risk assessment.

**5. Machine Learning Model:**

The collected user data is processed by the diabetes prediction model, which utilizes machine learning algorithms to predict the likelihood of the user 0developing diabetes.

**6. Prediction Result:**

The prediction model provides a diabetes risk assessment, which is communicated to the user in a user-friendly manner.

The chatbot may provide personalized recommendations for preventive measures, such as lifestyle changes or medical consultations.

**7. User Feedback and Interaction:**

Users can interact with the chatbot to seek clarification, ask follow-up questions, or provide feedback on the predictions and recommendations.

**8. Continuous Improvement:**

* User interactions and feedback are continuously analyzed to improve the chatbot's performance and prediction accuracy.
* The machine learning model is periodically updated with new data to enhance its predictive capabilities.

**9 Data Privacy and Security:**

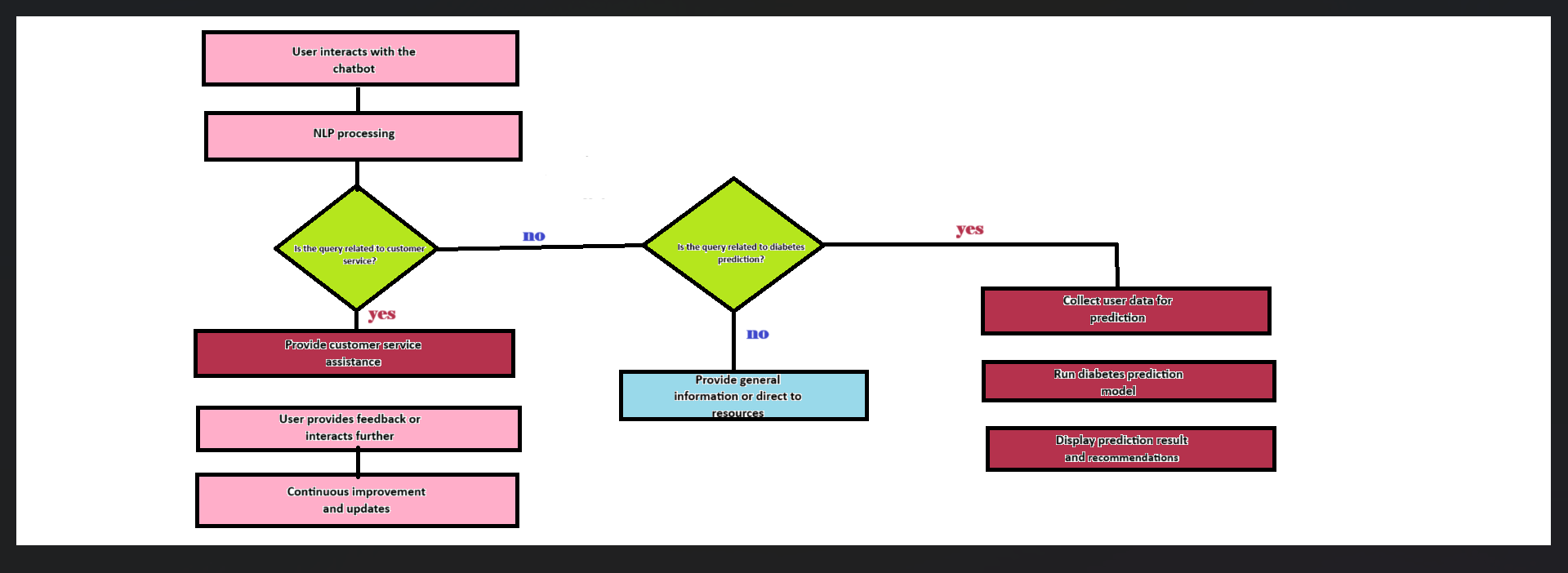
Throughout the process, strict data privacy and security measures are maintained to protect user information and ensure compliance with relevant regulations.

**11. Legal and Ethical Considerations:**

Legal and ethical guidelines are adhered to, and disclaimers are provided to users regarding the limitations of the chatbot's medical predictions.

**12. Maintenance and Updates:**

Regular maintenance and updates are performed to keep the chatbot and the prediction model up-to-date with the latest medical knowledge and user needs.

**FLOWCHART:**

**1.WHAT THE CHATBOT DOES:**

chatbot helps the users by answering questions and guiding them when they're stuck.

* The chatbot can be figure out what questions the chatbot can answer and how smart it is.
* the chatbot able to understand if users are happy or not.
* Learn from what users say to become better.

**2.HOW IT LOOKS AND FEELS:**

The chatbot is in such a way where user can easily talk to it, like on a website or in an app.

* The chatbot easy to use on websites or apps.
* Let users talk to the chatbot by typing or speaking.
* Help users before they ask by giving them useful info.

**3.HOW THE CHATBOT UNDERSTANDS AND TALKS:**

The chatbot understand and talk like a person, so users can chat with it in a natural way.

* Use fancy tools (like GPT-3) to help the chatbot understand and talk like a person.
* The chatbot know what users mean even if they don't say things perfectly**.**

**4.WHATR THE CHATBOT SAYS:**

Plan for the chatbot to give really helpful answers when users need assistance.

* The chatbot gives good and helpful answers.
* Change what the chatbot says based on who the user is.
* The chatbot learn and become better over time.

**5.HOW IT FITS IN:**

The chatbotfigure out how the it will work with websites or apps so users can easily chat with it there.

* Put the chatbot into websites and apps so it fits well.
* The users use can the chatbot on phones, computers, and other places.
* It works well on different devices.

**6.MAKING IT BETTER:**

Keep checking how well the chatbot is doing, ask users for their thoughts, and improve it over time.

* Test the chatbot a lot with real people to find problems.
* Try different versions to see which one works best.
* Keep updating the chatbot with new info and make it smarter.

**7.TRYING COOL IDEAS:**

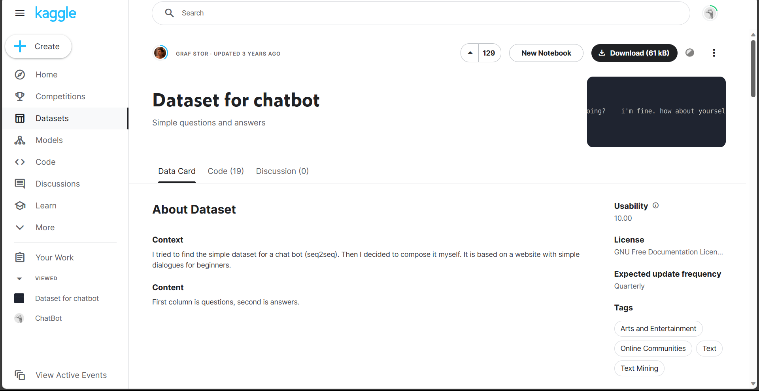
Experiment with smart ideas, like using advanced technology to make the chatbot's answers even better.

* Use computers to guess what users will do next.
* Make the chatbot learn by itself to get better without help.
* Personalize the chatbot for each user and try new things like looking at pictures or understanding data.

**STEPS TO CREATE CHATBOT:**

1. LOAD DATASET
2. PREPROCESS THE DATASET
3. TRAIN THE MODEL
4. IMPLEMENT TE LOGIC
5. DEVELOP THE CHATBOT
6. IMPLEMENT USING NLP
7. TEST THE CHATBOT

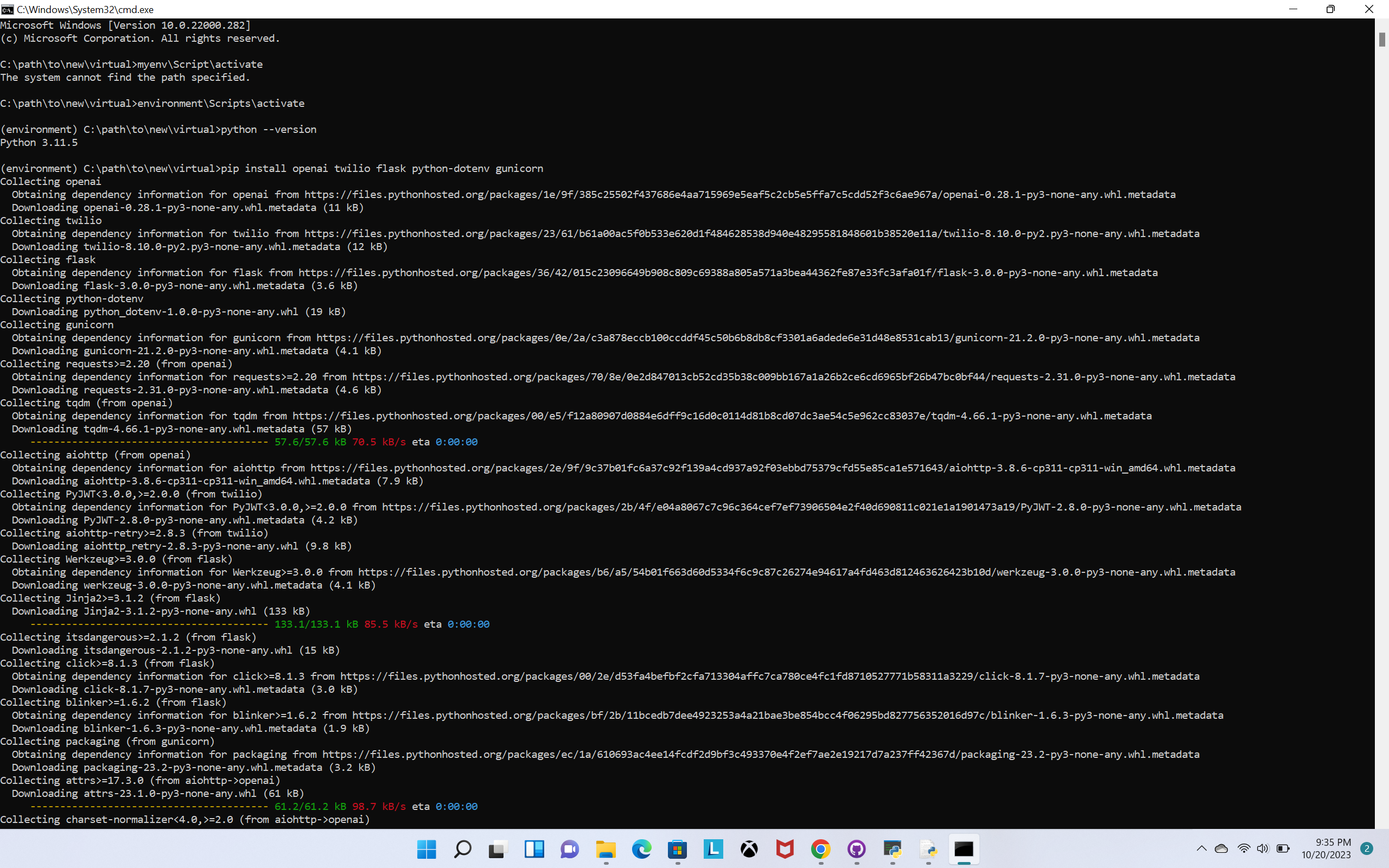
**DOWNLOAD THE DATASET FROM KAGGLE:**

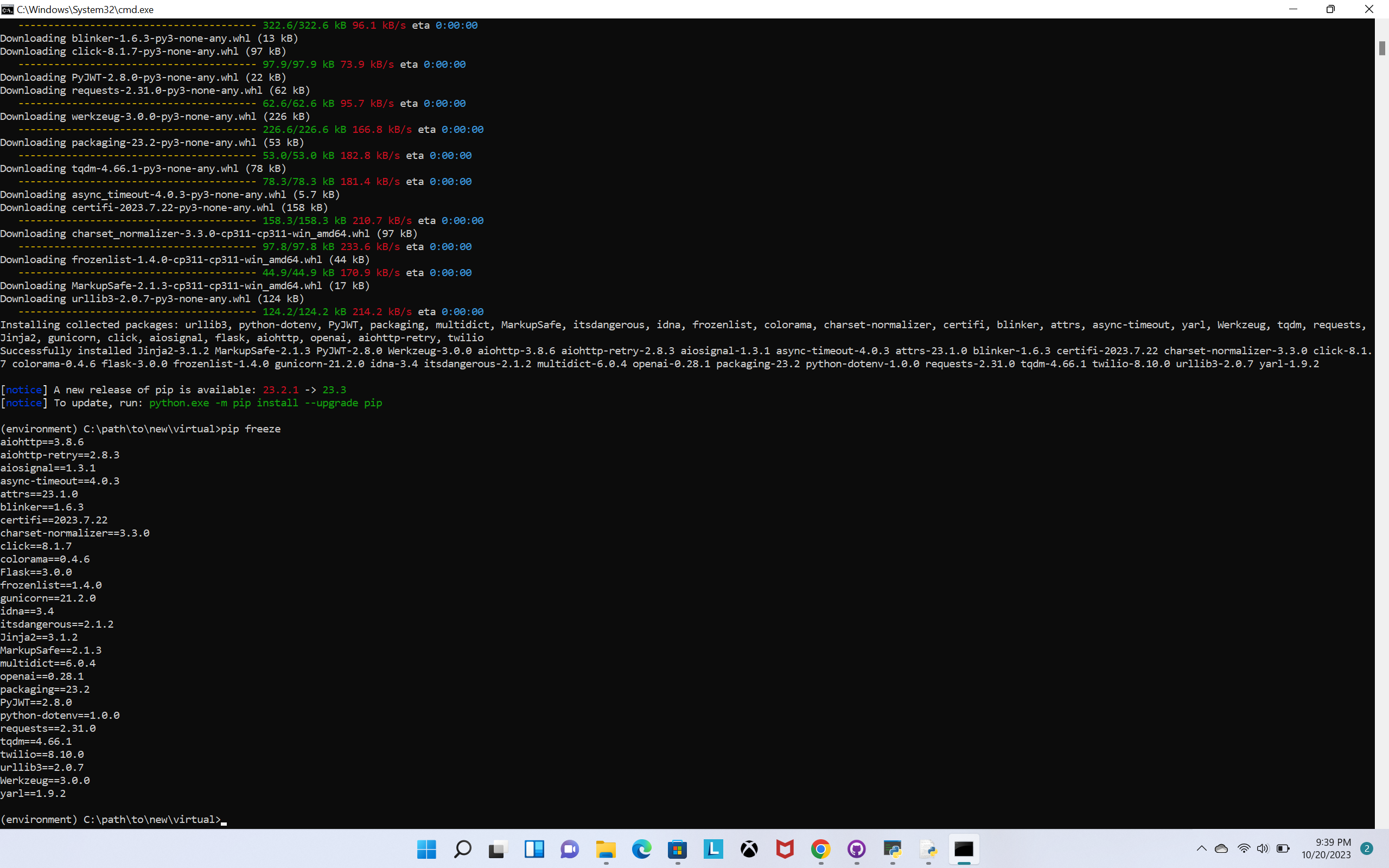


**DATA PREPROCESSING:**

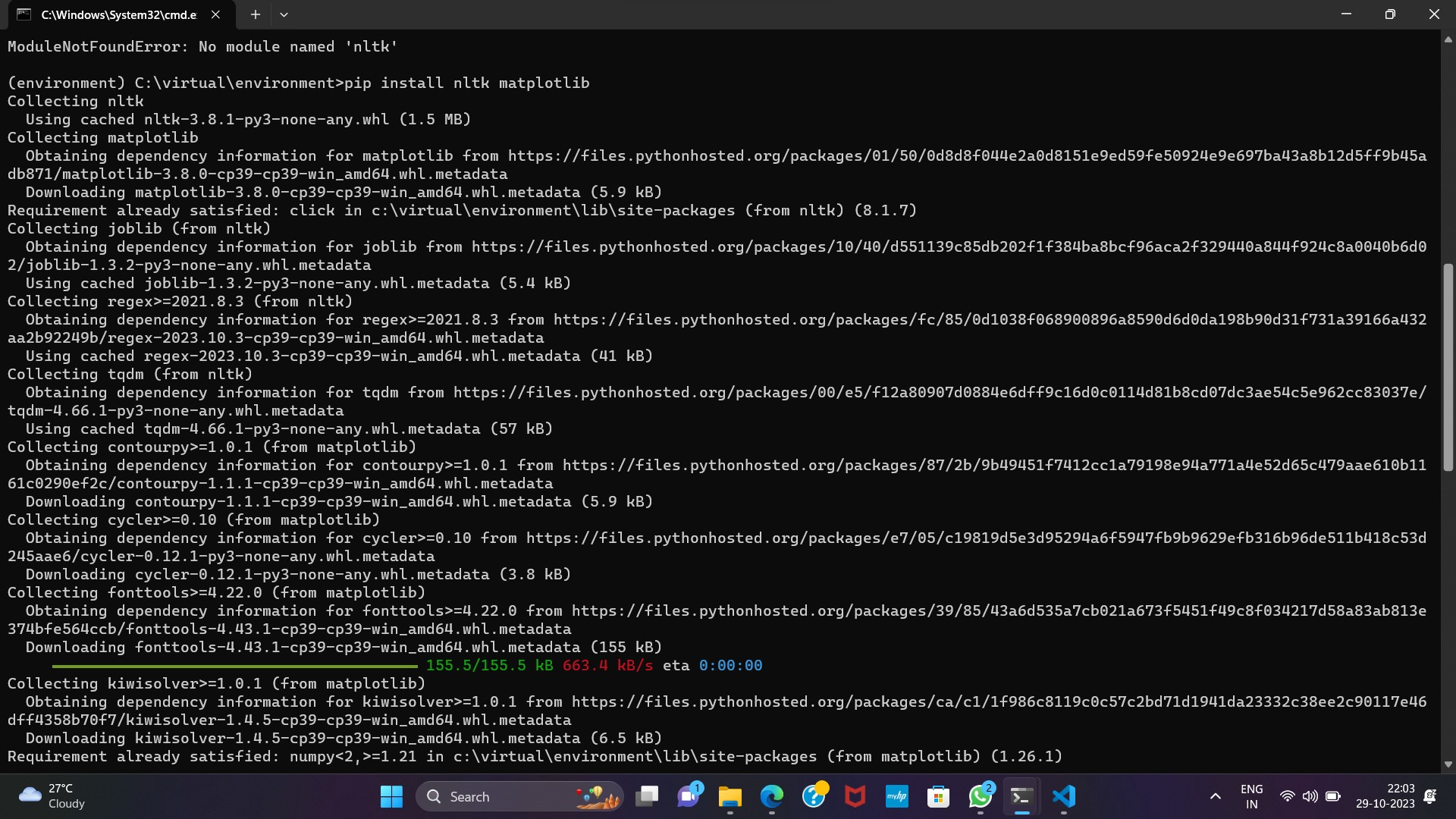
preprocessing is a crucial step in the data mining and data analyDatasis process that involves transforming raw data into a format that can be understood and analyzed by computers and machine learning algorithms.

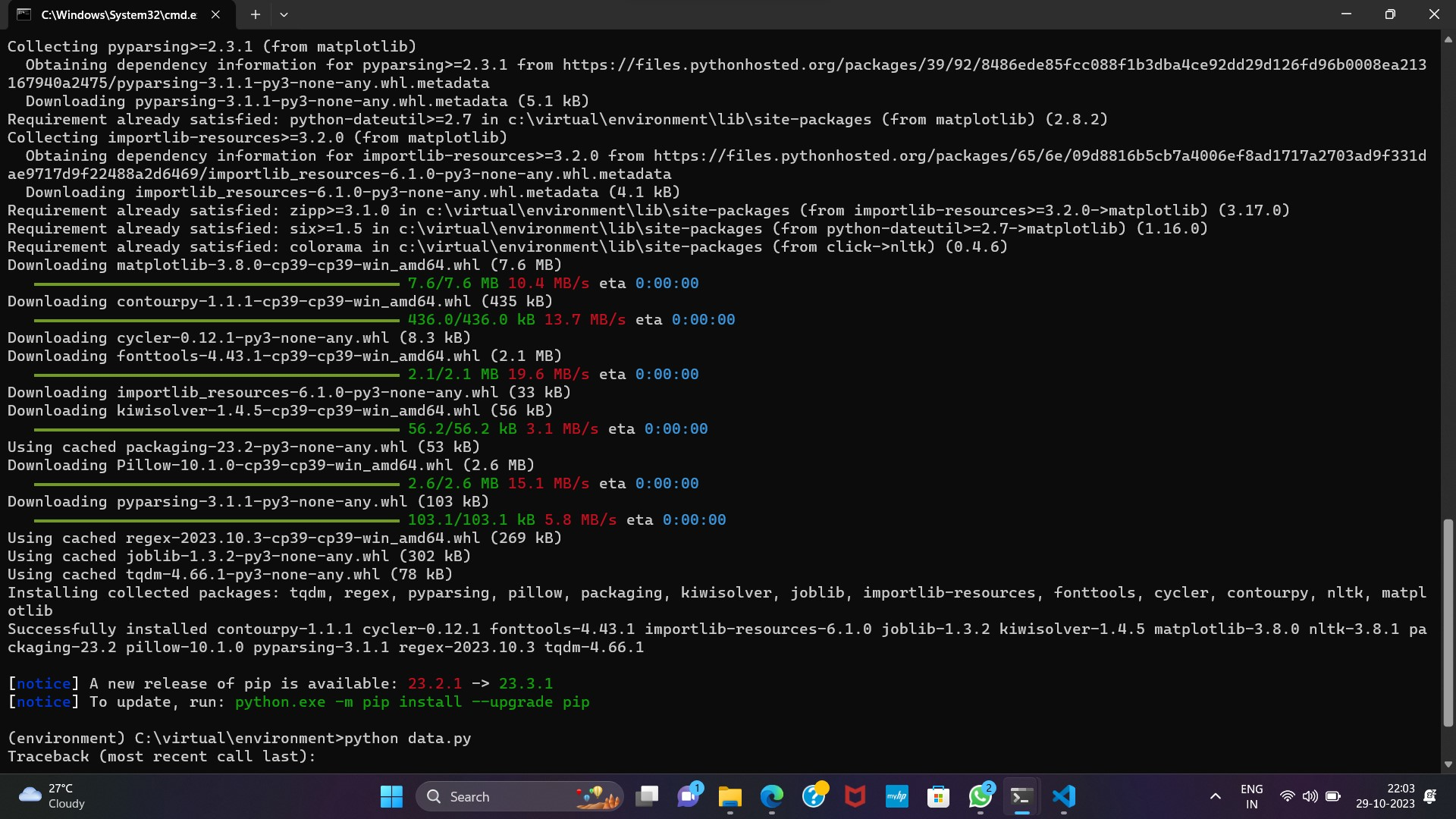
**CREATING VIRTUAL ENVIRONMENT:**

****

****

**INSTALLING LIBRARIES:**





**IMPORTING LIBRARIES:**

* Tensorflow
* Numpy
* Pandas
* Seaborn
* Matplotlib
* tensorflow.keras.layers
* TextVectorization
* re,string
* flask
* render\_template
* socketIO

**TENSORFLOW:**

TensorFlow is an open-source machine learning framework developed by Google that provides tools and libraries for building and deploying machine learning models, with support for tasks such as image recognition, natural language processing, and more. It offers flexibility, scalability, high-level APIs like Keras, and a visualization tool called TensorBoard.

**MATPLOTLIB:**

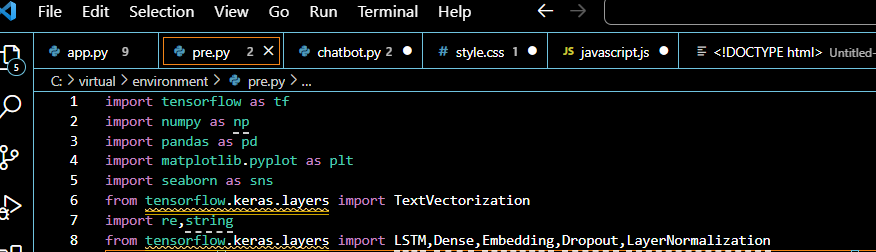
Matplotlib is a 2D plotting library for creating static, animated, and interactive visualizations in Python. It provides a wide variety of charts and plots for visualizing data.

**NUMMPY:**

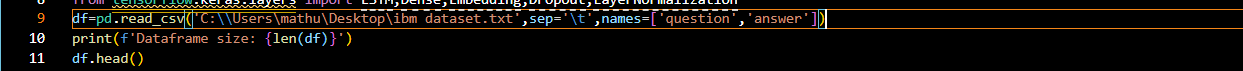
NumPy is a powerful numerical computing library for Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays efficiently.

**PANDAS**:

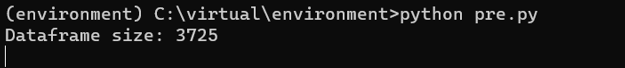
Pandas is a data manipulation and analysis library for Python. It offers data structures like DataFrames for efficient data handling and provides functions for reading, writing, and manipulating structured data, making it a crucial tool for data science and analysis tasks.

****

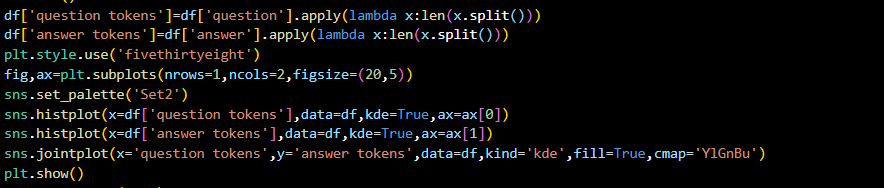
**READ THE DATASET:**

****

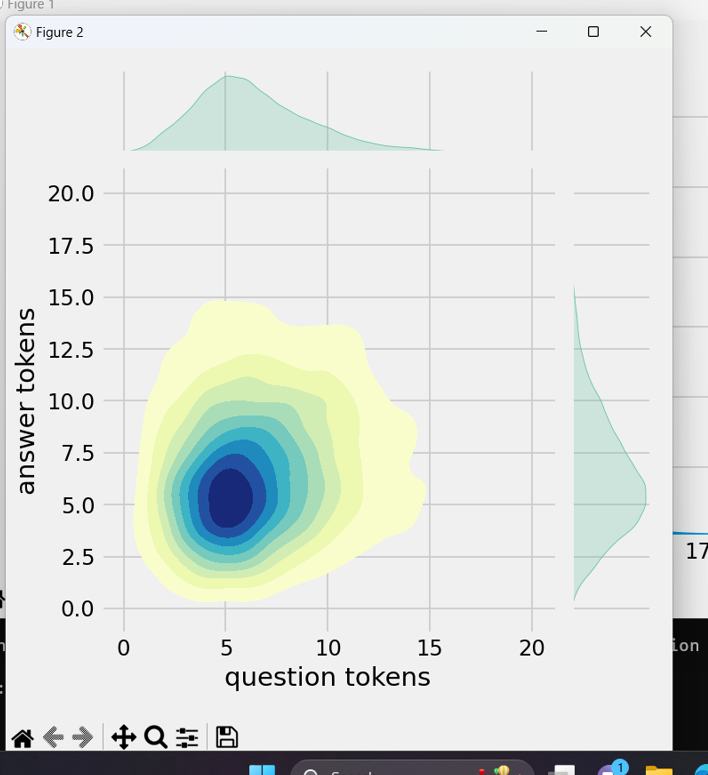
**OUTPUT:**

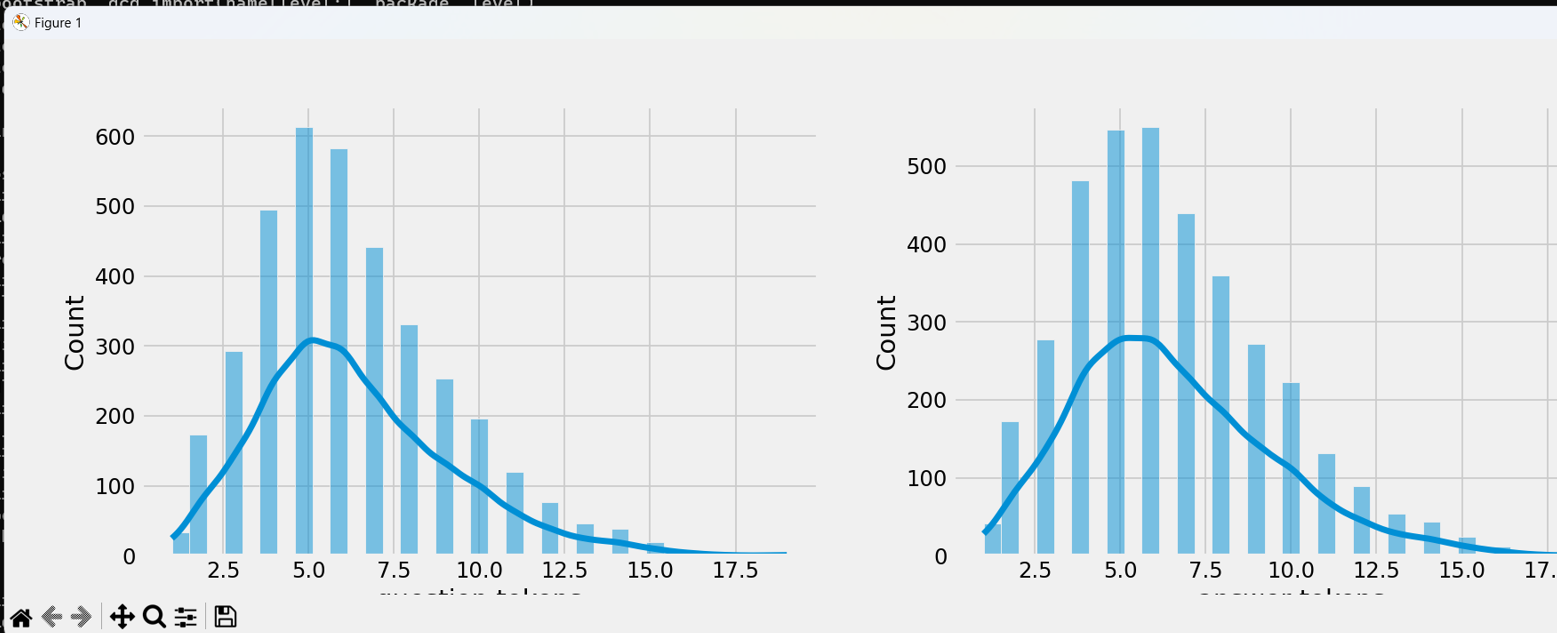
****

**DATA VISTUALIZATION:**

****

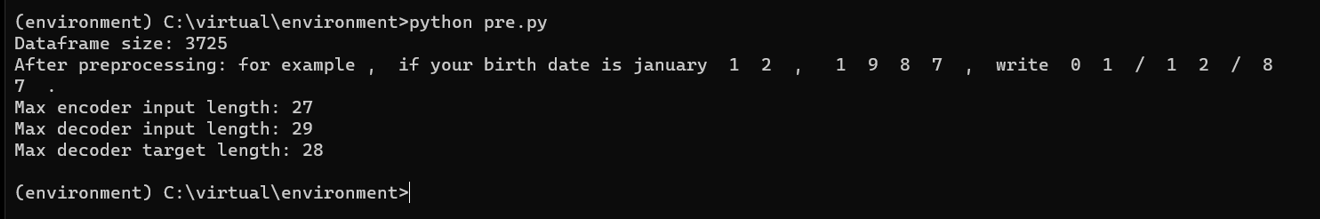
**OUTPUT:**

****

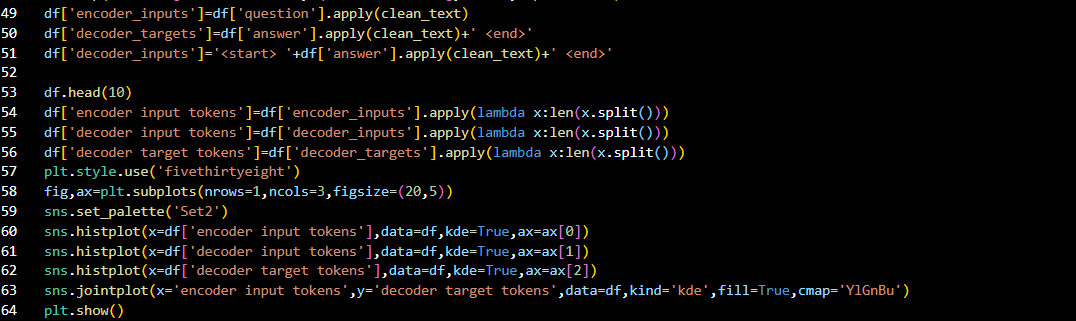
****

**TEXTCLEANING:**

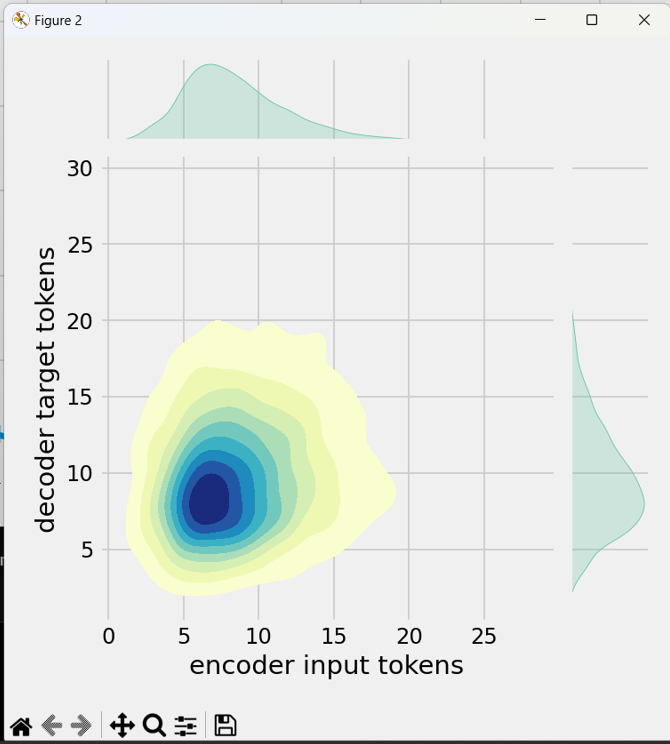
**OUTPUT:**

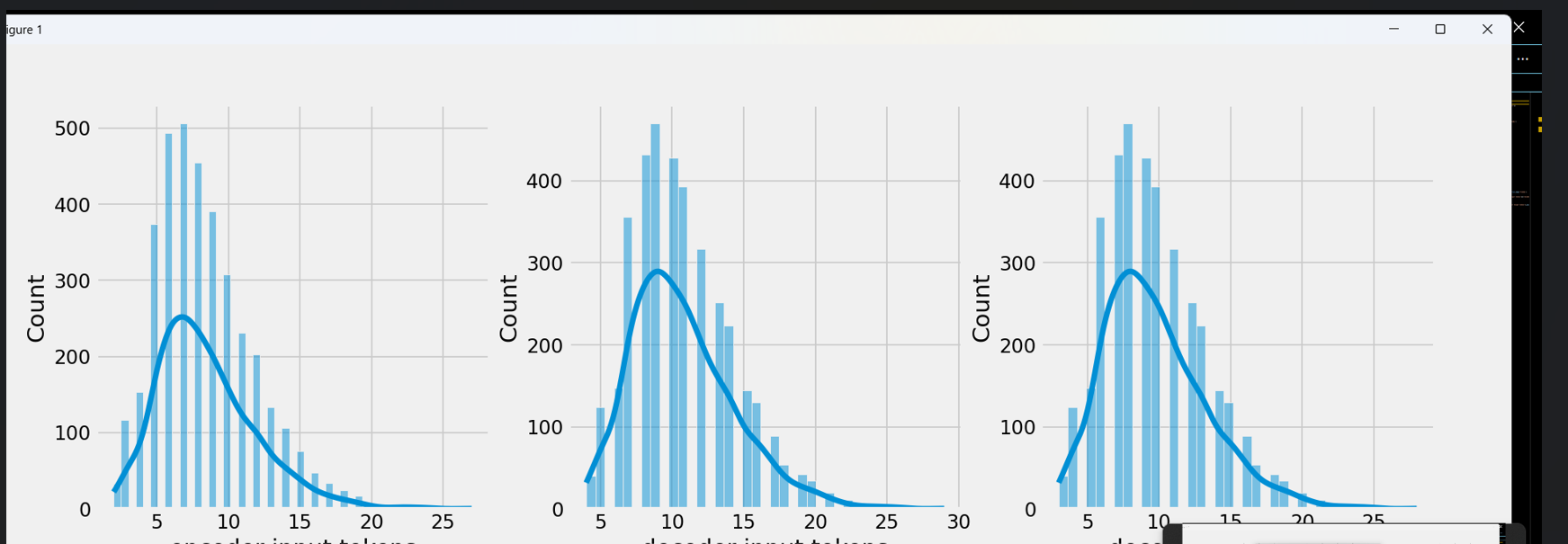
****

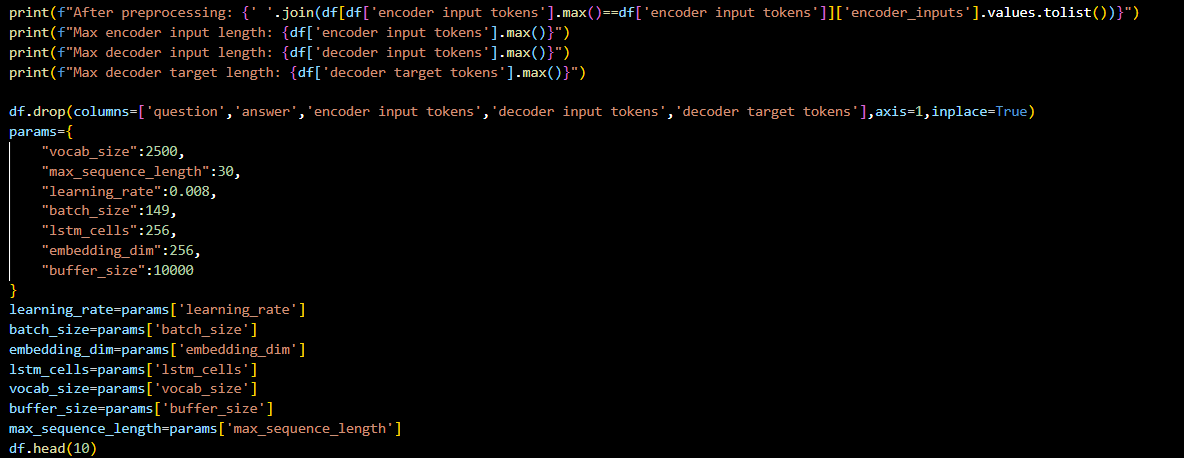
**Decoder and encoder:**

****

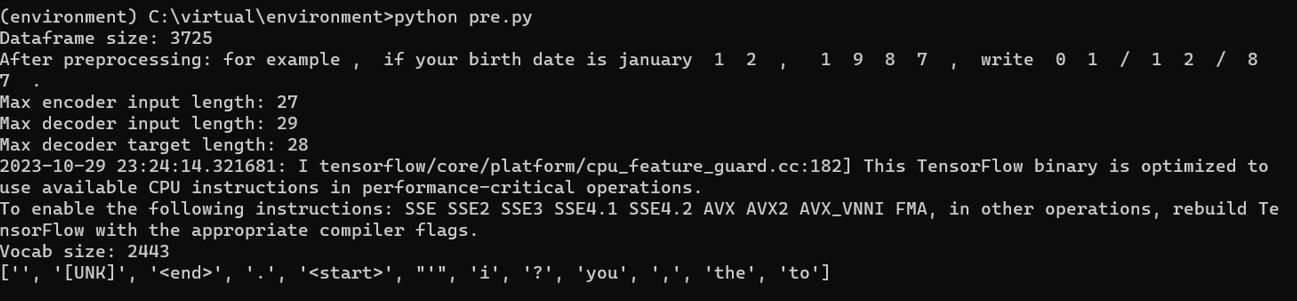
**OUTPUT:**

****

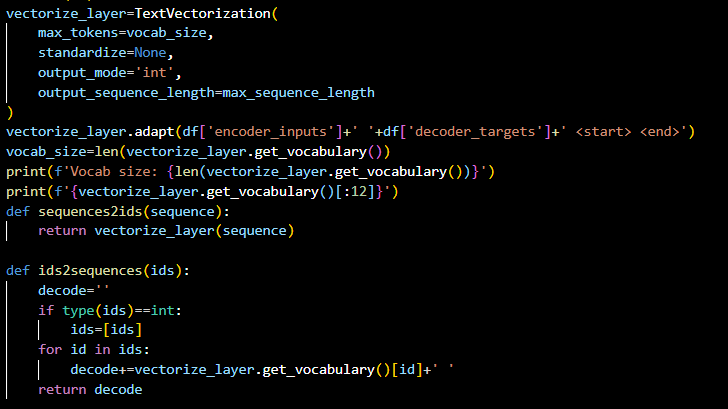
****

****

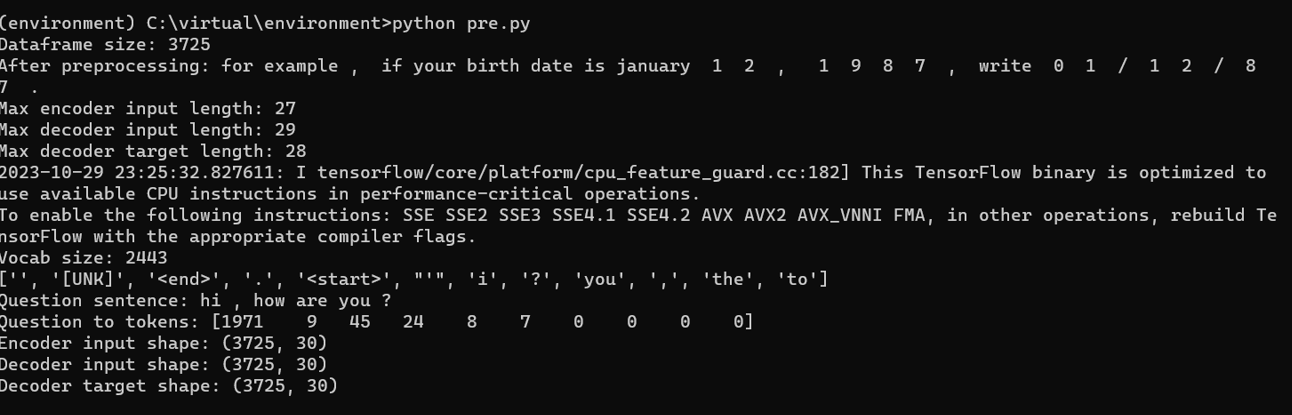
**OUTPUT:**

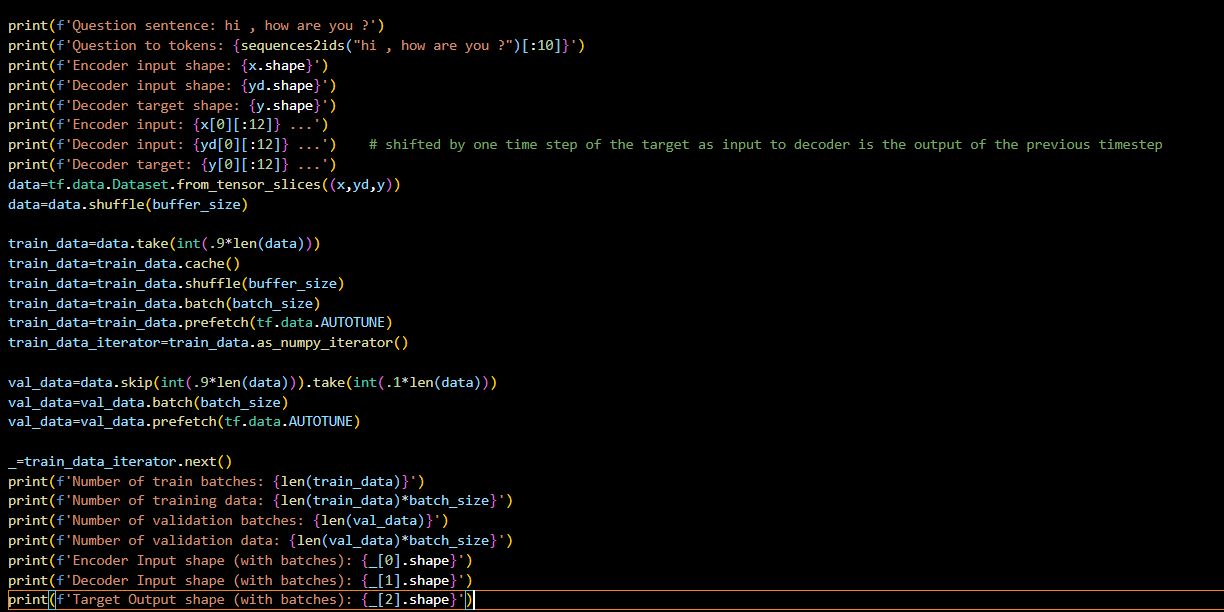
****

**TOKENIZATION:**

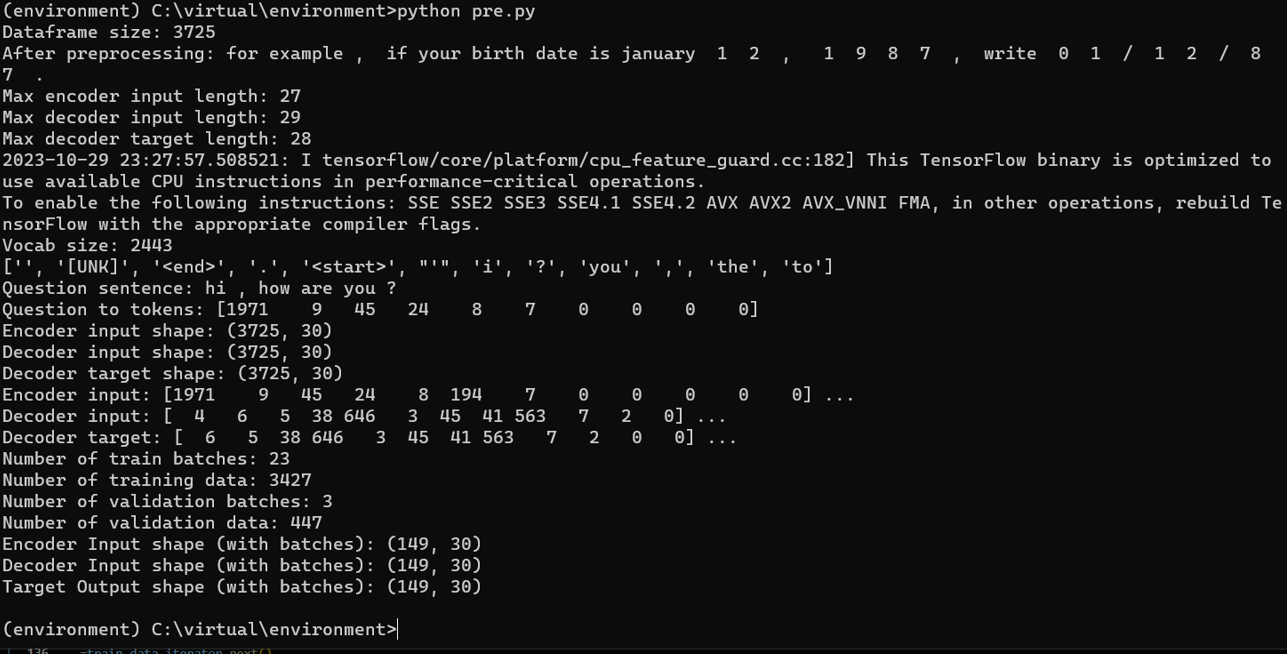
****

**OUTPUT:**

****

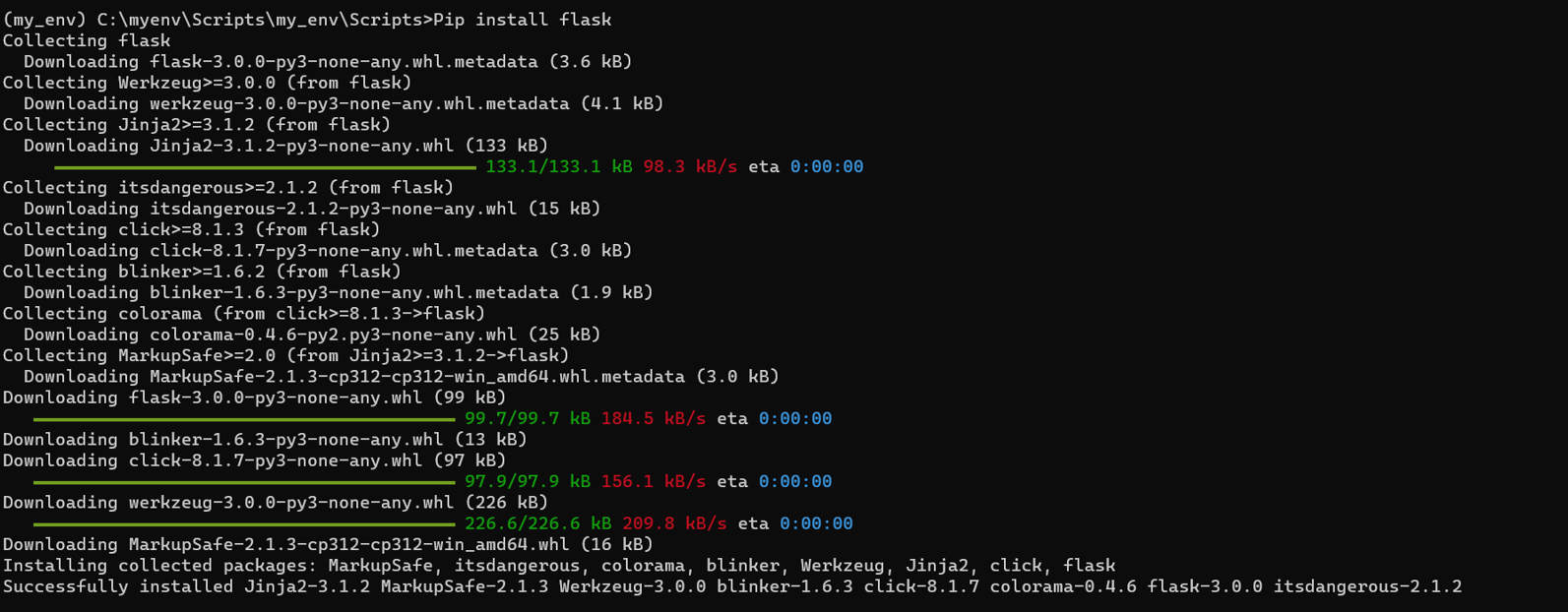
****

**OUTPUT:**

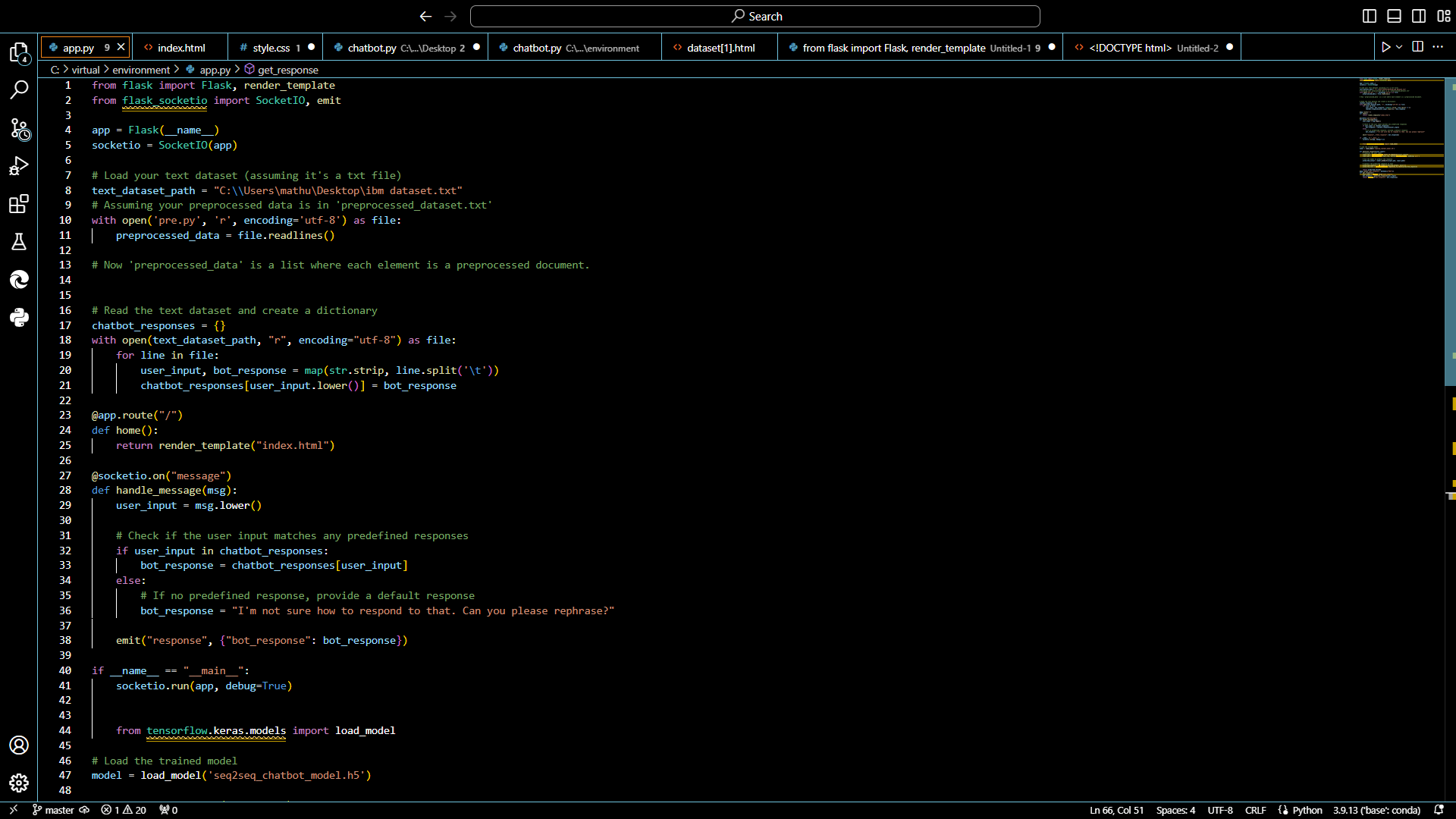
****

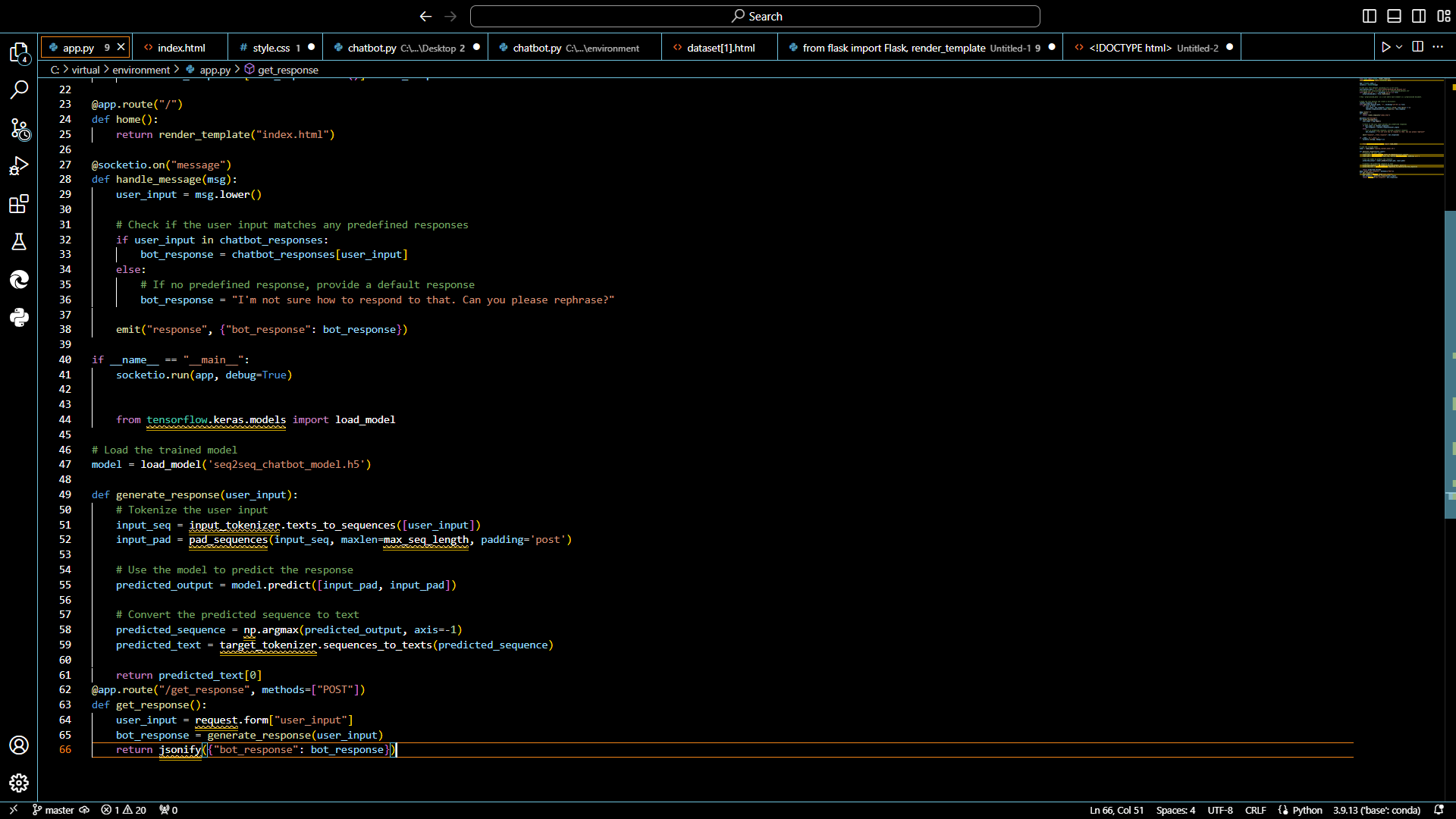
**STEP1 :CREATING A FLASK:**

1. Create a new directory for your Flask project.
2. Inside this directory, create a virtual environment

****

**STEP 2:CREATE FILE NAME APP.PY:**

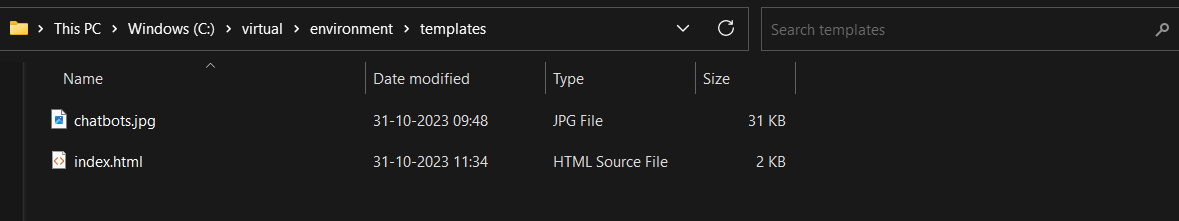
****

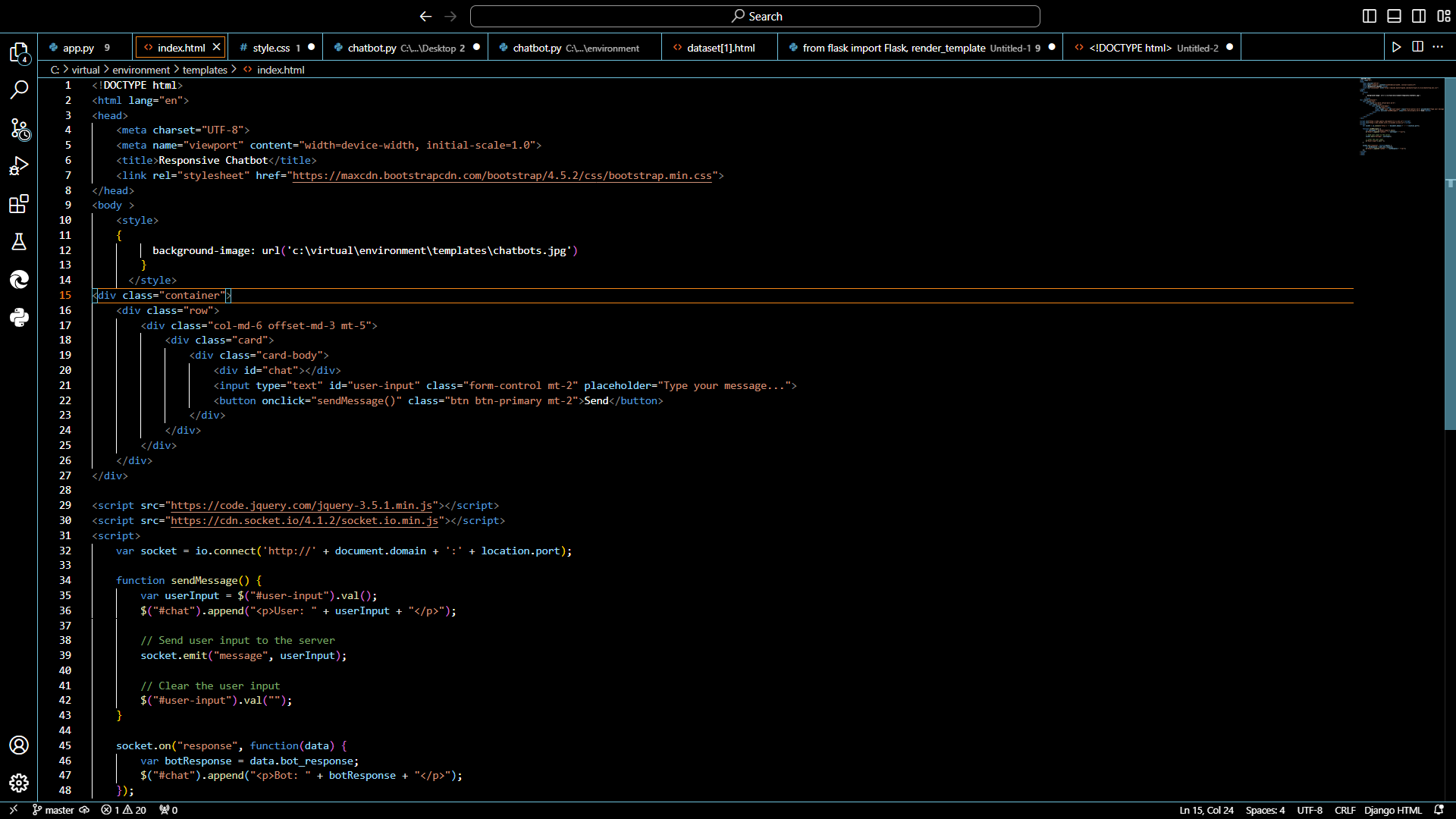


**STEP 3:CREATE HTML FILE IN A FOLDER NAME TEMPLETE:**

1. Create a folder named templates in your project directory.

2. Inside the templates folder, create an HTML file named index.html:

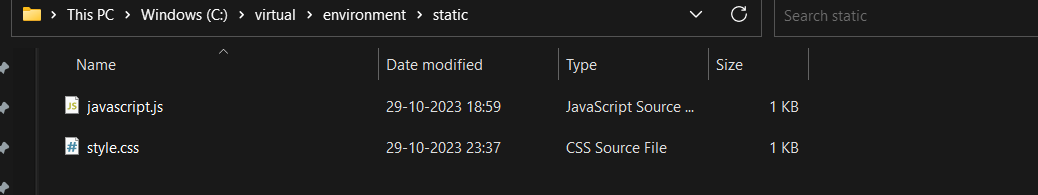


**HTML CODE:** 

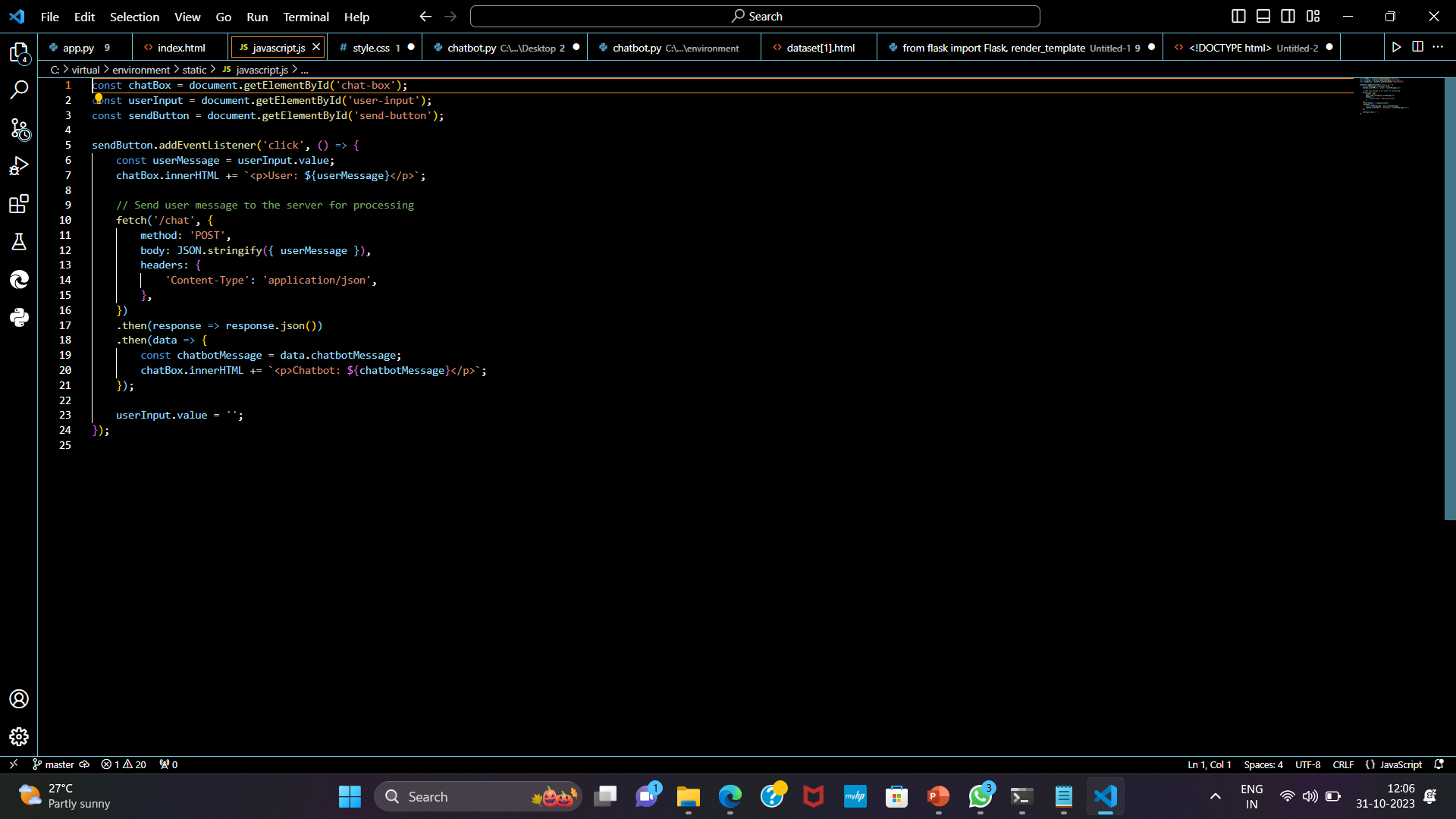
**STEP 4:CREATE A JAVASCRIPT NAMED CHATBOT:**

1.Create a folder named static in your project directory.

2. Inside the static folder, create a JavaScript file named chatbot.js. This file will handle user input and responses from the chatbot.

****

**JAVASCRIPT CODE:**

****

**STEP 5: IMPLEMENT CHATBOT LOGIC IN FLASK**

1. In app.py, add a new route to handle user messages and return chatbot responses.
2. Implement the get\_chatbot\_response function using your existing chatbot logic. This function should take a user message as input and return the chatbot's response.

**STEP 6: RUN YOUR FLASK APP**

1.Run your Flask app using the following command:

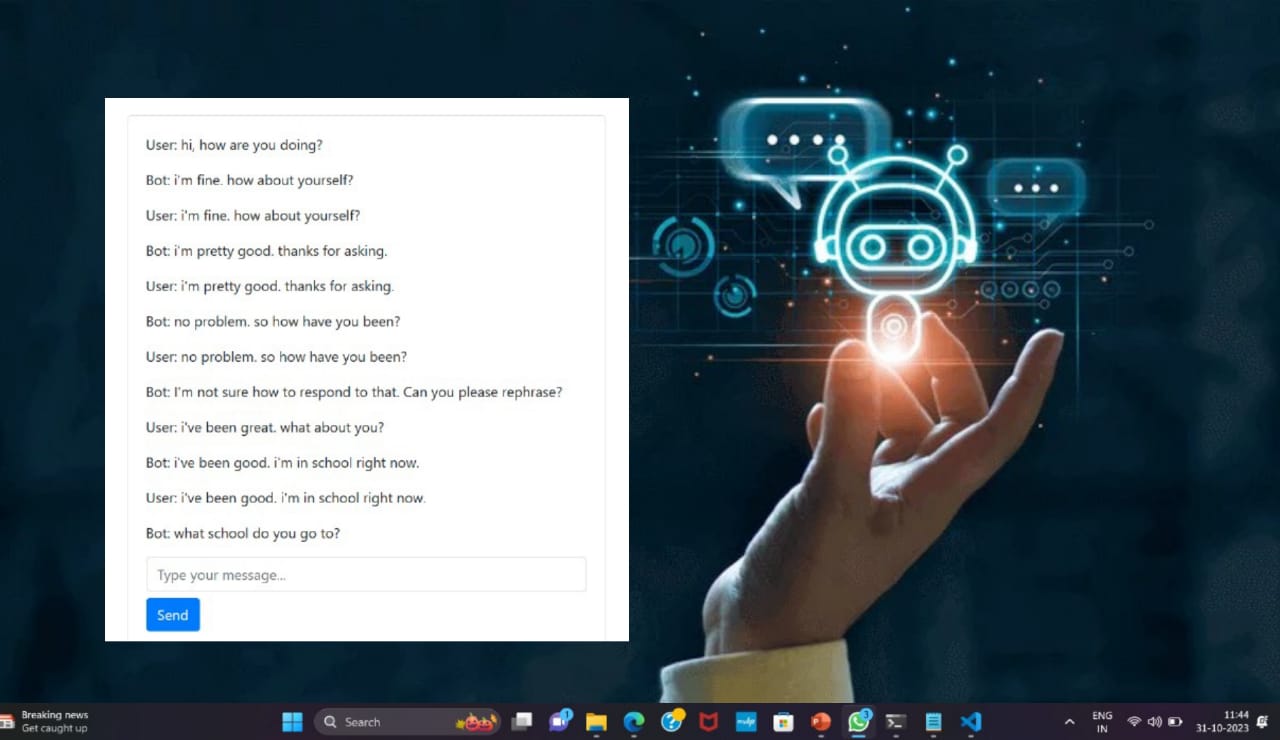


**PYTHON CODE:**

**Python app.py**

Access your chatbot web app by opening a web browser and navigating to [http://127.0.0.1:5000](http://127.0.0.1:5000/) Our chatbot is now integrated into a Flask web app. Users can interact with it through the web interface. Make sure you adapt the chatbot logic and responses to suit your specific use case and chatbot implementation. You can also enhance the web interface to make it more interactive and visually appealing

**OUTPUT OF OUR CHATBOT:**

****

**CONCLUTION :**

In short, creating an effective chatbot involves defining clear objectives, designing a user-friendly interface, integrating AI and NLP technologies, prioritizing security and privacy, ensuring scalability, and incorporating continuous learning. Regular testing, monitoring, and compliance with regulations are essential, along with providing documentation and training. The goal is to create a versatile and efficient tool that enhances user experience and delivers on its intended purpose.

**Hereby, I conclude that those content are not copied and done on my own**