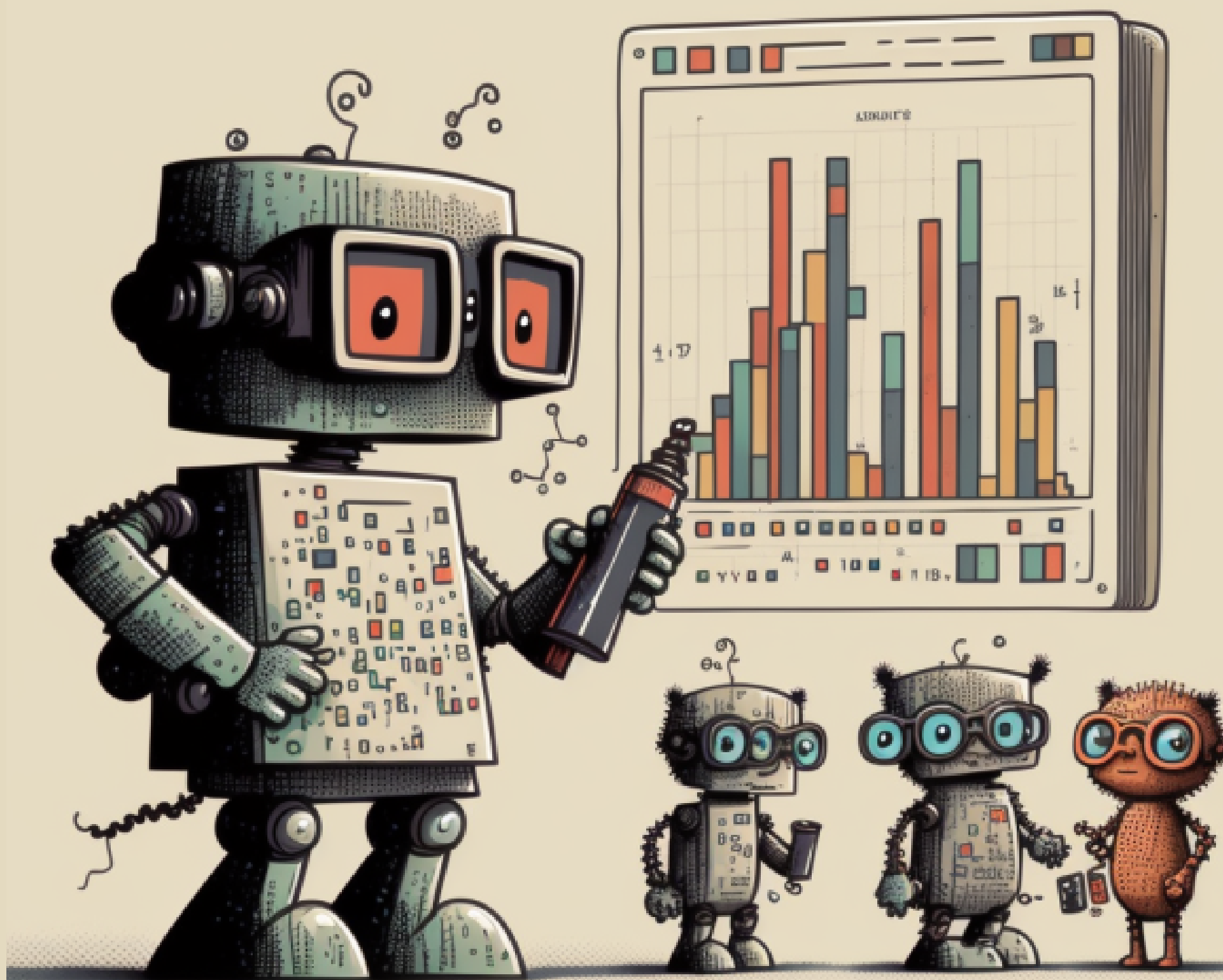


# Chat GPT 3 + Python



# 1 - Here is the first step. We have to describe in detail what we want.



Hi, help me create a streamlit web app, using pandas , plotly, seaborn - showing exploratory data analysis of Iris kaggle real dataset ? make sure we show various types of plots, with summary and description etc. Let's build a dashboard



Sure, here is a basic outline of how to create a Streamlit web app for exploratory data analysis of the Iris dataset:



1. Start by installing Streamlit and the necessary libraries (pandas, plotly, seaborn) using pip.
2. Import the libraries and load the Iris dataset using pandas.
3. Create a basic layout for the app using Streamlit's functions for creating text, buttons, and other widgets.
4. Create a function for displaying the basic statistics and information about the dataset.
5. Create a function for visualizing the data using different types of plots such as histograms, scatter plots, box plots, etc. You can use Plotly and Seaborn for creating these plots.
6. Create a function to show the summary of the dataset
7. Create a function to show the description of the dataset
8. Create a function to show the correlation of the dataset
9. Use Streamlit's `st.sidebar` function to create a sidebar for selecting different types of plots and options for the data visualization.
10. Use Streamlit's `st.write` function to display the output of the functions in the app.
11. Finally, use the `streamlit run` command to run the app locally.

This is just a basic outline, and you may need to add more details and customizations based on your specific requirements.

# 2 - Now the AI starts with the Magic.

## The code was compiled and tested in VSCODE



OK , provide me the best python code to make the exploratory analysis from the real iris data set that you suggested me above



Here is some sample Python code that demonstrates how to perform exploratory data analysis on the Iris dataset using the libraries and techniques that I mentioned in my previous response:



Copy code

```
import streamlit as st
import pandas as pd
import seaborn as sns
import plotly.express as px

# Load the Iris dataset
df = pd.read_csv("https://raw.githubusercontent.com/mwaskom/seaborn-data/master/iris.csv")

# Create a basic layout for the app
st.title("Exploratory Data Analysis of Iris Dataset")
st.sidebar.title("Iris Dataset EDA")

# Create a function to display basic statistics and information about the dataset
def show_dataset_info():
    st.write("**Shape:**")
    st.write(df.shape)
    st.write("**Columns:**")
    st.write(df.columns)
    st.write("**Head:**")
    st.write(df.head())
```

Regenerate response

# 3 - Chat GPT 3 gave me the wrong code. OK, I asked him to fix it himself and that's it. New code is done



is this part of the code corret ? If no , do it again please



```
# Create a function to visualize the data using different types of plots
def show_plots():
    plot_type = st.sidebar.selectbox("Select a plot type", ["Histogram", "Scatter Plot", "Box Plot"])
    if plot_type == "Histogram":
        st.write("Histogram")
        sns.histplot(data=df, x='species', y='sepal_length', element='step')
        st.pyplot()
    elif plot_type == "Scatter Plot":
        st.write("Scatter Plot")
        sns.scatterplot(data=df, x='sepal_length', y='sepal_width', hue='species')
        st.pyplot()
    elif plot_type == "Box Plot":
        st.write("Box Plot")
        sns.boxplot(data=df, x='species', y='petal_length')
        st.pyplot()
```



I apologize, but the code that I provided for the `show_plots()` function is not correct. The `sns.histplot()` and `element='step'` is not available in seaborn and `sns.scatterplot(data=df, x='sepal_length', y='sepal_width', hue='species')` it should be `sns.scatterplot(x='sepal_length', y='sepal_width', hue='species', data=df)`



Here is the corrected version of the function:

Copy code

```
def show_plots():
    plot_type = st.sidebar.selectbox("Select a plot type", ["Histogram",
"Scatter Plot", "Box Plot"])
    if plot_type == "Histogram":
```

Regenerate response



perfect , now i need you add the code of violin graph to explore the types please



Sure, here is an example of how to add a violin plot to the `show_plots()` function to explore the Iris species:



Copy code

```
def show_plots():
    plot_type = st.sidebar.selectbox("Select a plot type", ["Histogram",
"Scatter Plot", "Box Plot", "Violin Plot"])
    if plot_type == "Histogram":
        st.write("Histogram")
        for col in df.columns[:-1]:
            st.write(f"Histogram of {col}")
            sns.histplot(data=df, x=col)
            st.pyplot()
    elif plot_type == "Scatter Plot":
        st.write("Scatter Plot")
        sns.scatterplot(x='sepal_length', y='sepal_width',
hue='species',data=df)
        st.pyplot()
    elif plot_type == "Box Plot":
        st.write("Box Plot")
        sns.boxplot(data=df, x='species', y='petal_length')
        st.pyplot()
    elif plot_type == "Violin Plot":
        st.write("Violin Plot")
        sns.violinplot(x='species', y='petal_length', data=df)
        st.pyplot()
```

In this version of the function, I added a new option "Violin Plot" to the `selectbox` and then created a new `elif` block to handle this case. In this block, I used the `sns.violinplot()` function to create a violin plot of the `petal_length` variable grouped by the `species` variable. And then I used `st.pyplot()` to display the plot in the app. Please note that you can change the x,y variables and add more variables that you want to visualize.

# 4 - Subtitles and Table of Contents Ready



thanks , now could you include a subtitle for the app ? you can summarize what the app shows to the user and how they can explore the app please



Yes, you can add a subtitle to the app to provide a brief summary of what the app shows to the user and how they can explore it.



You can use the `st.markdown()` function to add a subtitle with a brief summary of the app, for example:

Copy code

```
st.markdown("### Iris dataset EDA Dashboard")
```

You can also add a summary of the app at the beginning, for example:

Copy code

```
st.markdown("This app allows you to explore the Iris dataset using various types of plots, summary statistics and correlation. You can select different options for the data visualization using the sidebar.")
```

And you can add instruction for the user on how to use the app, for example:

Copy code

```
st.markdown("To explore the data, select the options in the sidebar, and the corresponding plots, summary statistics and correlation will be displayed.")
```

You can place this markdown under the title of your app, so the user will know what the app is about, how to use it and what they can expect to find.

Please let me know if there's anything else I can help you with.

# 5 - To post on Streamlit we have to create a requirements.txt file and upload it along with the .py file to your GitHub.



Sure, here's an example of a `requirements.txt` file that you can use for your Streamlit app:



 Copy code

```
streamlit==0.65.2
pandas==1.2.3
seaborn==0.11.0
plotly==4.12.0
```

This file lists the dependencies that your Streamlit app needs to run, which in this case are `streamlit`, `pandas`, `seaborn` and `plotly`. Each line lists the name of the package, followed by the version number.

You can create this file by running `pip freeze > requirements.txt` on your command line. Please note that the version numbers may change according to the version that you have installed.

Please let me know if there's anything else I can help you with.

# 6 - Dashboard working perfectly. Now we can perform the Exploratory Analysis.

×

### Iris Dataset EDA

MENU - Explore the Iris dataset

- Show dataset info
- Show Plots
- Show Summary
- Show Description
- Show Correlation
- Show Missing Data

### Iris Dataset EDA

MENU - Explore the Iris dataset

- Show dataset info
- Show Plots

Select a plot type

Pairplot

- Show Summary
- Show Description
- Show Correlation
- Show Missing Data

Share ☆ ☰

## Exploratory Data Analysis of Iris Dataset

The Iris dataset is a collection of information about different types of flowers. Each flower is described by 4 different measurements: the length and width of the petals and the length and width of the sepals. These measurements are used to identify 3 different species of Iris flowers: setosa, versicolor, and virginica.

This app allows you to explore the Iris dataset using various types of plots, summary statistics and correlation. You can select different options for the data visualization using the sidebar.

To explore the data, select the options in the sidebar, and the corresponding plots, summary statistics and correlation will be displayed.

Pairplot

