App.py code successful

import streamlit as st  
import preprocessor,helper  
import matplotlib.pyplot as plt  
import seaborn as sns  
  
st.sidebar.title("Whatsapp Chat Analysis")  
  
uploaded\_file = st.sidebar.file\_uploader("Choose a file")  
if uploaded\_file is not None:  
 bytes\_data = uploaded\_file.getvalue()  
 data= bytes\_data.decode("utf-8")  
 df = preprocessor.preprocess(data)  
  
 st.dataframe(df)  
  
 #fetch unique users  
 user\_list = df['user'].unique().tolist()  
 user\_list.remove('group\_notification')  
 user\_list.sort()  
 user\_list.insert(0, "Overall")  
  
 selected\_user = st.sidebar.selectbox("show analysis with respect to",user\_list)  
  
 if st.sidebar.button("show Analysis"):  
  
 num\_messages, words, num\_media\_messages, num\_links = helper.fetch\_stats(selected\_user,df)  
 st.title("Top Statistics of Whatsapp Chat")  
  
 col1, col2, col3, col4 = st.columns(4)  
  
  
 with col1:  
 st.header('Total Messages')  
 st.title(num\_messages)  
 with col2:  
 st.header('Total Words')  
 st.title(words)  
 with col3:  
 st.header('Media Shared')  
 st.title(num\_media\_messages)  
 with col4:  
 st.header('Links Shared')  
 st.title(num\_links)  
  
  
  
 # finding the busiest user in the group  
  
 if selected\_user == "Overall":  
 st.title("Most Busy Users")  
 x, new\_df =helper.most\_busy\_users(df)  
 fig, ax = plt.subplots()  
  
 col1,col2 = st.columns(2)  
  
 with col1:  
 ax.bar(x.index,x.values, color='red')  
 plt.xticks(rotation='vertical')  
 st.pyplot(fig)  
  
 with col2:  
 st.dataframe(new\_df)  
  
 # most common words  
 most\_common\_df = helper.most\_common\_words(selected\_user,df)  
 st.dataframe(most\_common\_df)  
  
 fig,ax = plt.subplots()  
  
 ax.bar(most\_common\_df[0], most\_common\_df[1],color='orange')  
 plt.xticks(rotation = 'vertical')  
 st.title("Most Common Words")  
  
 st.pyplot(fig)  
  
 #emoji analysis  
 emoji\_df = helper.emoji\_helper(selected\_user,df)  
 st.title("Emoji Analysis")  
  
 col1,col2 = st.columns(2)  
  
 with col1:  
 st.dataframe(emoji\_df)  
  
 with col2:  
 fig,ax =plt.subplots()  
 ax.pie(emoji\_df[1].head(), labels=emoji\_df[0].head(), autopct="%0.2f")  
 st.pyplot(fig)  
  
 #Monthly timeline  
 st.title("Monthly Timeline Analysis")  
 timeline=helper.monthly\_timeline(selected\_user,df)  
 fig,ax = plt.subplots()  
 ax.plot(timeline['time'], timeline['message'],color="green")  
 plt.xticks(rotation="vertical")  
 st.pyplot(fig)  
  
 #Daily timeline  
  
 # daily timeline  
 st.title("Daily Timeline")  
 daily\_timeline = helper.daily\_timeline(selected\_user, df)  
 fig, ax = plt.subplots()  
 ax.plot(daily\_timeline['only\_date'], daily\_timeline['message'], color='brown')  
 plt.xticks(rotation='vertical')  
 st.pyplot(fig)  
  
 #activity map  
 st.title("Activity Map")  
 col1,col2 =st.columns(2)  
  
 with col1:  
 st.header("Most Busy Day")  
 busy\_day = helper.week\_activity\_map(selected\_user,df)  
 fig,ax= plt.subplots()  
 ax.bar(busy\_day.index, busy\_day.values, color="purple")  
 plt.xticks(rotation='vertical')  
 st.pyplot(fig)  
  
 with col2:  
 st.header("Most busy month")  
 busy\_month = helper.month\_activity\_map(selected\_user, df)  
 fig, ax = plt.subplots()  
 ax.bar(busy\_month.index, busy\_month.values, color='magenta')  
 plt.xticks(rotation='vertical')  
 st.pyplot(fig)  
  
 st.title("Online Activity Map")  
 user\_heatmap = helper.activity\_heatmap(selected\_user, df)  
 fig, ax = plt.subplots()  
 ax = sns.heatmap(user\_heatmap)  
 st.pyplot(fig)