SOCAL MEDIA DASHBOARD

**Task Overview**

A **Social Media Dashboard** collects data from social media platforms, processes it, and displays key metrics in a visual format. For this project:

* 1. We'll use dummy data for simplicity.
  2. Libraries like **Matplotlib**, **Pandas**, and **Plotly** will handle data processing and visualization.

**Requirements**

Install necessary libraries:

pip install pandas matplotlib plotly

**Example Code**

class SocialMediaDashboard:

def \_\_init\_\_(self):

self.users = {} # Stores user profiles

self.posts = [] # Stores posts made by users

def create\_user(self, username, bio):

"""Creates a new user profile."""

if username in self.users:

print(f"User '{username}' already exists.")

else:

self.users[username] = {'bio': bio, 'followers': [], 'following': []}

print(f"User '{username}' created successfully.")

def follow\_user(self, follower, followee):

"""Allows one user to follow another."""

if follower not in self.users or followee not in self.users:

print("Both users must exist to follow.")

return

if followee not in self.users[follower]['following']:

self.users[follower]['following'].append(followee)

self.users[followee]['followers'].append(follower)

print(f"{follower} is now following {followee}.")

else:

print(f"{follower} is already following {followee}.")

def create\_post(self, username, content):

"""Creates a new post for a user."""

if username not in self.users:

print("User does not exist.")

else:

post = {'username': username, 'content': content}

self.posts.append(post)

print(f"Post created by {username}.")

def view\_posts(self, username):

"""Displays posts from a user's feed."""

if username not in self.users:

print("User does not exist.")

else:

feed = [post for post in self.posts if post['username'] in self.users[username]['following'] or post['username'] == username]

if not feed:

print("No posts to display.")

else:

print(f"\nPosts for {username}:")

for post in feed:

print(f"{post['username']}: {post['content']}")

# Main program

def main():

dashboard = SocialMediaDashboard()

while True:

print("\nSocial Media Dashboard")

print("1. Create User")

print("2. Follow User")

print("3. Create Post")

print("4. View Posts")

print("5. Exit")

choice = input("Enter your choice: ")

if choice == '1':

username = input("Enter username: ")

bio = input("Enter bio: ")

dashboard.create\_user(username, bio)

elif choice == '2':

follower = input("Enter your username: ")

followee = input("Enter username to follow: ")

dashboard.follow\_user(follower, followee)

elif choice == '3':

username = input("Enter your username: ")

content = input("Enter post content: ")

dashboard.create\_post(username, content)

elif choice == '4':

username = input("Enter your username: ")

dashboard.view\_posts(username)

elif choice == '5':

print("Exiting... Goodbye!")

break

else:

print("Invalid choice. Please try again.")

if \_\_name\_\_ == "\_\_main\_\_":

main()

**Explanation**

**1.Dataset**:

* + - Contains metrics for different social media platforms: followers, engagement rates, and the number of posts.

**2.Matplotlib Visualizations**:

* + - **Bar Chart**: Shows the number of followers for each platform.
    - **Line Chart**: Displays engagement rates per platform.
  1. **Plotly Dashboard**:
     + Combines a bar chart for followers and a line chart for engagement rates in an interactive dashboard.

**Output of the Code**

**Here’s what the code will produce:**

**1. Tabular Data Output**

**The print(df) statement will display the data in tabular format:**

**Social Media Metrics:**

**Platform Followers Engagement\_Rate Posts\_This\_Month**

**0 Facebook 1500 4.5 12**

**1 Twitter 2300 3.8 18**

**2 Instagram 4500 7.2 22**

**3 LinkedIn 1200 5.1 8**

**2. Bar Chart: Followers per Platform**

**A bar chart will display the number of followers for each platform:**

* **X-Axis: Platforms (Facebook, Twitter, Instagram, LinkedIn)**
* **Y-Axis: Number of followers**
* **Bars: Blue, cyan, purple, and green bars representing follower counts.**

**(*Placeholder image: Replace with actual output when you run the code*)**

**3. Line Chart: Engagement Rate per Platform**

**A line chart will display the engagement rate for each platform:**

* **X-Axis: Platforms**
* **Y-Axis: Engagement rate (%)**
* **Line: Orange line connecting data points.**

**(*Placeholder image: Replace with actual output when you run the code*)**

**4. Plotly Interactive Dashboard**

**The interactive dashboard combines:**

* **A bar chart for followers (blue bars).**
* **A line chart for engagement rate (orange line).**

**Interactive features include:**

* **Hover tooltips to see exact values.**
* **Zooming and panning on charts.**

**(*Placeholder image: Replace with actual output when you run the code*)**

**Steps to Run the Code**

1. **Install Dependencies: Ensure pandas, matplotlib, and plotly are installed:**

**pip install pandas matplotlib plotly**

1. **Execute the Code: Save the code in a Python file (e.g., social\_media\_dashboard.py) and run it.**
2. **View Outputs:**
   * **Bar and line charts will appear as static images.**
   * **The Plotly dashboard will open in your browser.**

**Enhancements**

* 1. Use APIs (e.g., Twitter API, Facebook Graph API) to fetch real data.
  2. Add a time-series analysis to track metrics over time.
  3. Build a web app using Flask or Dash to make it accessible via a browser.