Experiment No: 2

Aim: Data Collection-Select the social media platforms of your choice (Twitter, Facebook, LinkedIn, Youtube,Web blogs etc) ,connect to and capture social media data for business (scraping, crawling,parsing)

```
import requests
import pandas as pd

video_id = "iX-U9HLRnuk"
api_key = "AIzaSyDL2_rH__2JrGPYAN71ryuO9fG3WxXyT04"

# Get video info (optional)
video_info_url =
f"https://www.googleapis.com/youtube/v3/videos?part=snippet&id={video_id}&key={api_key}"
video_info_response = requests.get(video_info_url)
video_info_data = video_info_response.json()

# Get comments (first page only)
comments_url =
f"https://www.googleapis.com/youtube/v3/commentThreads?part=snippet&videoId={video_id}&key={api_key
}"
comments_response = requests.get(comments_url)
comments_data = comments_response.json()

# Extract comments
comments = [
    item["snippet"]["topLevelComment"]["snippet"]["textOriginal"]
    for item in comments_data.get("items", [])
]

# Simple rule-based sentiment function
def simple_sentiment(comment):
    positive_words = ["good", "great", "awesome", "nice", "love", "excellent", "best", "amazing"]
    negative_words = ["bad", "worst", "awful", "terrible", "boring", "hate", "poor", "dislike"]
    comment_lower = comment.lower()

    if any(word in comment_lower for word in positive_words):
        return "Positive"
    elif any(word in comment_lower for word in negative_words):
        return "Negative"
    else:
        return "Neutral"

# Analyze all comments
comment_list = []
sentiment_list = []

for comment in comments:
    sentiment = simple_sentiment(comment)
    comment_list.append(comment)
    sentiment_list.append(sentiment)
    print(f"{comment} : {sentiment}")

# Save to DataFrame
sentiment_df = pd.DataFrame({
    "Comments": comment_list,
    "Sentiment": sentiment_list
})

# Display the results
print("\nSentiment DataFrame:")
print(sentiment_df.head())
```