```
!pip install -q nltk seaborn matplotlib

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import nltk
from nltk.sentiment.vader import SentimentIntensityAnalyzer
from google.colab import files
import zipfile
import io

sns.set(style='whitegrid')
plt.style.use('ggplot')
nltk.download('vader_lexicon')

uploaded = files.upload()
filename = next(iter(uploaded))

with zipfile.ZipFile(io.BytesIO(uploaded[filename])) as z:
    csv_filename = z.namelist()[0]
    df = pd.read_csv(z.open(csv_filename))

df = df[['Score', 'Text']].dropna().head(500)
df['Score'] = df['Score'].astype(int)

plt.figure(figsize=(10, 5))
sns.countplot(data=df, x='Score', palette='Blues')
plt.title('Count of Reviews by Star Rating')
plt.xlabel('Star Rating')
plt.ylabel('Number of Reviews')
plt.show()

sia = SentimentIntensityAnalyzer()
vader_scores = [sia.polarity_scores(str(text)) for text in df['Text']]
vader_df = pd.DataFrame(vader_scores)

df_vader = pd.concat([df.reset_index(drop=True), vader_df], axis=1)
grouped         =        df_vader.groupby('Score')[['pos',        'neu',        'neg',
'compound']].mean().reset_index()

fig, axs = plt.subplots(1, 4, figsize=(20, 5))
sns.barplot(data=grouped, x='Score', y='pos', ax=axs[0], color='green')
sns.barplot(data=grouped, x='Score', y='neu', ax=axs[1], color='gold')
sns.barplot(data=grouped, x='Score', y='neg', ax=axs[2], color='darkred')
sns.barplot(data=grouped, x='Score', y='compound', ax=axs[3], color='limegreen')
axs[0].set_title('Average Positive Sentiment')
axs[1].set_title('Average Neutral Sentiment')
axs[2].set_title('Average Negative Sentiment')
axs[3].set_title('Average Compound Sentiment')
plt.tight_layout()
plt.show()

print("\nMost Positive Review:")
print(df_vader.sort_values('pos', ascending=False).iloc[0]['Text'])
```

```python
print("\nMost Negative Review:")
print(df_vader.sort_values('neg', ascending=False).iloc[0]['Text'])

specific_rating = 1
filtered_reviews = df_vader[df_vader['Score'] == specific_rating]

print(f"\nFiltered Reviews with Score = {specific_rating}:")
for i, row in filtered_reviews.iterrows():
    print(f"\nReview: {row['Text']}")
    print(f"Compound Score: {row['compound']}")
```