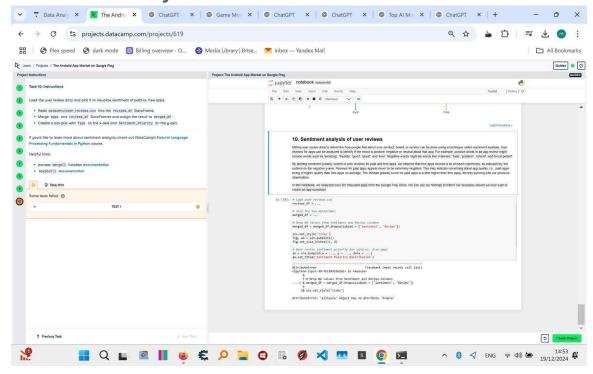
## **Sentiment Analysis of User Reviews**



## **Question:**

Load the user review data and plot it to visualize sentiment of paid vs. free apps:

- 1. Read 'datasets/user reviews.csv' into the reviews df DataFrame.
- 2. Merge apps and reviews\_df DataFrames and assign the result to merged\_df.
- 3. Create a box plot with Type on the x-axis and Sentiment\_Polarity on the y-axis.

## Code:

import pandas as pd import seaborn as sns import matplotlib.pyplot as plt

# Load the user reviews data reviews df = pd.read csv('datasets/user reviews.csv')

# Merge the DataFrames
merged\_df = pd.merge(apps, reviews\_df, on='App', how='inner')

# Drop NA values from Sentiment and Review columns
merged\_df = merged\_df.dropna(subset=['Sentiment', 'Review'])

```
# Set seaborn style
sns.set_style('ticks')
plt.figure(figsize=(10, 6))

# Create a box plot
sns.boxplot(x='Type', y='Sentiment_Polarity', data=merged_df)
plt.title('Sentiment Polarity Distribution')
plt.show()
```

## **Explanation:**

- 1. `pd.read\_csv()` is used to load the user review data from a CSV file into a DataFrame called `reviews df`.
- 2. `pd.merge()` combines `apps` and `reviews\_df` DataFrames on the 'App' column using an inner join, creating `merged\_df`.
- 3. `dropna()` removes rows where 'Sentiment' or 'Review' columns have missing values.
- 4. `sns.set\_style('ticks')` sets the visual style of the plots to 'ticks' for better readability.
- 5. A new figure is created with `plt.figure(figsize=(10, 6))` to set the size of the plot.
- 6. `sns.boxplot()` creates a box plot with 'Type' (Paid/Free) on the x-axis and 'Sentiment\_Polarity' on the y-axis, using the `merged\_df` DataFrame.
- 7. The title of the plot is set using `plt.title()`.
- 8. `plt.show()` displays the plot.