NAME - Sakshi Mane

CLASS - CS2

Roll no. - CS2:81

PRN - 202401040238

# EDS THEORY ACTIVITY NO.1

## Yelp Reviews Assignment - Pandas and NumPy Solutions

This notebook contains 20 problem statements formulated on the Yelp Reviews dataset along with their solutions using Pandas and NumPy.

```
import pandas as pd
import numpy as np

# Simulated Yelp dataset
data = {
    'business_id': ['b1', 'b2', 'b3', 'b4', 'b5', 'b1', 'b2', 'b3', 'b4', 'b5'],
    'user_id': ['u1', 'u2', 'u3', 'u4', 'u5', 'u6', 'u7', 'u8', 'u9', 'u10'],
    'stars': [5, 4, 3, 5, 2, 5, 1, 4, 3, 2],
    'text': [
        "Amazing food and service!",
        "Good place but a bit pricey.",
        "Average experience.",
        "Loved everything about it!",
        "Not worth the hype.",
        "Fantastic meals!",
        "Terrible service.",
        "Nice ambiance and tasty food.",
        "Decent but could improve.",
        "Disappointing visit."
    ],
    'city': ['Las Vegas', 'Las Vegas', 'Toronto', 'Toronto', 'Phoenix', 'Phoenix', 'Charlotte', 'Charlotte', 'Las Vegas', 'Toronto']
}

yelp_df = pd.DataFrame(data)
yelp_df
```

# FOLLOWING ARE THE QUESTIONS WITH THEIR CODE (in Jupyter Nb) - - - -

```python
# 1. Find the total number of reviews
yelp_df.shape[0]
```

Python

```python
# 2. Find the number of unique businesses
yelp_df['business_id'].nunique()
```

Python

```python
# 3. Find the average rating across all reviews
yelp_df['stars'].mean()
```

Python

```python
# 4. Find the highest rated review
yelp_df[yelp_df['stars'] == yelp_df['stars'].max()]
```

Python

```python
# 5. Find the lowest rated review
yelp_df[yelp_df['stars'] == yelp_df['stars'].min()]
```

Python

```python
# 6. Count reviews by city
yelp_df['city'].value_counts()
```

Python

```python
# 7. Average stars given per city
yelp_df.groupby('city')['stars'].mean()
```

Python

```python
# 8. Find businesses with more than 1 review
yelp_df['business_id'].value_counts()[yelp_df['business_id'].value_counts() > 1]
```

Python

```python
# 9. List all users who rated 5 stars
yelp_df[yelp_df['stars'] == 5]['user_id']
```

Python

```python
# 10. Find reviews that mention "service"
yelp_df[yelp_df['text'].str.contains("service", case=False)]
```

Python

```python
# 11. Find the review text length for each review
yelp_df['text_length'] = yelp_df['text'].apply(len)
yelp_df[['user_id', 'text_length']]
```

Python

```python
# 12. Find the average text length
yelp_df['text_length'].mean()
```

Python

```python
# 13. Create a new column "Positive" where stars >=4 are positive
yelp_df['Positive'] = np.where(yelp_df['stars'] >= 4, True, False)
yelp_df[['user_id', 'Positive']]
```

Python

```python
# 14. Count number of positive and negative reviews
yelp_df['Positive'].value_counts()
```

Python

```python
# 15. Find businesses with the most 5-star ratings
yelp_df[yelp_df['stars'] == 5]['business_id'].value_counts()
```

Python

```python
# 16. Find average rating per business
yelp_df.groupby('business_id')['stars'].mean()
```

Python

```python
# 17. Identify users who gave below 3 stars
yelp_df[yelp_df['stars'] < 3]['user_id']
```

Python

```python
# 18. Check if any review has word "Amazing"
yelp_df['text'].str.contains("Amazing", case=False).any()
```

Python

```python
# 19. Find percentage of 5-star reviews
(yelp_df[yelp_df['stars'] == 5].shape[0] / yelp_df.shape[0]) * 100
```

Python

```python
# 20. Create a summary statistics table for stars
yelp_df['stars'].describe()
```

Python