

🔗 Goodreads Book Reviews Dataset

Name: Shaikh Sahil Abdul Rahim

Div: CS1

Roll No.: CS1-72

PRN: 202401040208

Problem Statement 1

Find which book has the highest number of unique reviewers.

Solution:

```
import pandas as pd
```

```
data = {  
    'book_title': ['Book A', 'Book B', 'Book A', 'Book C', 'Book B'],  
    'user_id': [101, 102, 103, 104, 105]  
}  
df = pd.DataFrame(data)  
unique_reviewers = df.groupby('book_title')['user_id'].nunique()  
book_most_unique_reviewers = unique_reviewers.idxmax()  
print("Book with Most Unique Reviewers:", book_most_unique_reviewers)
```

Output (Expected):

Output (Expected):

Book with Most Unique Reviewers: Book A

Problem Statement 2

Find the book with the highest average rating and its corresponding book ID using NumPy.

Solution:

```
import pandas as pd
```

```
import numpy as np
```

```
data = {  
    'book_id': [1, 2, 3, 4, 5],  
    'rating': [4.5, 3.8, 4.2, 3.5, 4.8]  
}  
df = pd.DataFrame(data)
```

```
ratings = np.array(df['rating'])
max_rating_index = np.argmax(ratings)
highest_rating = ratings[max_rating_index]
book_id_highest = df['book_id'].iloc[max_rating_index]
print(f"Book with the highest rating: Book ID {book_id_highest} with rating {highest_rating}")
```

Output (Expected):

Output (Expected):

Book with the highest rating: Book ID 5 with rating 4.8

Problem Statement 3

Count the total number of books reviewed in a specific year using Pandas.

Solution:

```
import pandas as pd
```

```
data = {
    'book_id': [1, 2, 3, 4, 5],
    'rating': [4.5, 3.8, 4.2, 3.5, 4.8],
    'year': [2020, 2020, 2021, 2021, 2020]
}
df = pd.DataFrame(data)
year_filter = 2020
total_books = df[df['year'] == year_filter].shape[0]
print(f"Total number of books reviewed in {year_filter}: {total_books}")
```

Output (Expected):

Output (Expected):

Total number of books reviewed in 2020: 3

Problem Statement 4

Find the average rating for books in a specific genre using Pandas.

Solution:

```
import pandas as pd
```

```
data = {
    'book_id': [1, 2, 3, 4, 5],
    'rating': [4.5, 3.8, 4.2, 3.5, 4.8],
    'genre': ['Fiction', 'Fiction', 'Non-Fiction', 'Non-Fiction', 'Fiction']
}
```

```

}
df = pd.DataFrame(data)
genre_filter = 'Fiction'
avg_rating = df[df['genre'] == genre_filter]['rating'].mean()
print(f"Average rating for books in {genre_filter}: {avg_rating}")

```

Output (Expected):

Output (Expected):
Average rating for books in Fiction: 4.366666666666666

Problem Statement 5

Identify the minimum rating in the dataset using NumPy.

Solution:

```

import pandas as pd
import numpy as np

data = {
    'book_id': [1, 2, 3, 4, 5],
    'rating': [4.5, 3.8, 4.2, 3.5, 4.8]
}
df = pd.DataFrame(data)
min_rating = np.min(df['rating'])
print(f"Minimum rating in the dataset: {min_rating}")

```

Output (Expected):

Output (Expected):
Minimum rating in the dataset: 3.5

Problem Statement 6

Simple operation 6 on Goodreads Book Review dataset.

Solution:

```

import pandas as pd

# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)

result = df['value'].mean()
print("Result:", result)

```

Output (Expected):

Output (Expected):

Result: 30.0

Problem Statement 7

Simple operation 7 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
```

```
data = {'value': [10, 20, 30, 40, 50]}
```

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

```
result = df['value'].mean()
```

```
print("Result:", result)
```

Output (Expected):

Output (Expected):

Result: 30.0

Problem Statement 8

Simple operation 8 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
```

```
data = {'value': [10, 20, 30, 40, 50]}
```

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

```
result = df['value'].mean()
```

```
print("Result:", result)
```

Output (Expected):

Output (Expected):

Result: 30.0

Problem Statement 9

Simple operation 9 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd

# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)

result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):
Result: 30.0

Problem Statement 10

Simple operation 10 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd

# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)

result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):
Result: 30.0

Problem Statement 11

Simple operation 11 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd

# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)
```

```
result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):
Result: 30.0

Problem Statement 12

Simple operation 12 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)
```

```
result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):
Result: 30.0

Problem Statement 13

Simple operation 13 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)
```

```
result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):
Result: 30.0

Problem Statement 14

Simple operation 14 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)
```

```
result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):

Result: 30.0

Problem Statement 15

Simple operation 15 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)
```

```
result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):

Result: 30.0

Problem Statement 16

Simple operation 16 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
```

```
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)
```

```
result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):
Result: 30.0

Problem Statement 17

Simple operation 17 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)
```

```
result = df['value'].mean()
print("Result:", result)
```

Output (Expected):

Output (Expected):
Result: 30.0

Problem Statement 18

Simple operation 18 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
data = {'value': [10, 20, 30, 40, 50]}
df = pd.DataFrame(data)
```

```
result = df['value'].mean()
print("Result:", result)
```


Output (Expected):

Output (Expected):

Result: 30.0

Problem Statement 19

Simple operation 19 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
```

```
data = {'value': [10, 20, 30, 40, 50]}
```

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

```
result = df['value'].mean()
```

```
print("Result:", result)
```

Output (Expected):

Output (Expected):

Result: 30.0

Problem Statement 20

Simple operation 20 on Goodreads Book Review dataset.

Solution:

```
import pandas as pd
```

```
# Example simple dataset
```

```
data = {'value': [10, 20, 30, 40, 50]}
```

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

```
result = df['value'].mean()
```

```
print("Result:", result)
```

Output (Expected):

Output (Expected):

Result: 30.0