Name: كريم اشراف احمد Id: 22030171

Harry Potter Analysis

First, We read the data, preform some cleaning and preprocessing, and make sure the data is ready to be analyzed

```
data = pd.read_csv("books.csv")

data.head()
```

[5] ✓ 0.0s Python

	book_id	goodreads_book_id	best_book_id	work_id	books_count	isbn	isbn13	authors	original_pub
0	1	2767052	2767052	2792775	272	439023483	9.780439e+12	Suzanne Collins	
1	2	3	3	4640799	491	439554934	9.780440e+12	J.K. Rowling, Mary GrandPré	
2	3	41865	41865	3212258	226	316015849	9.780316e+12	Stephenie Meyer	
3	6	11870085	11870085	16827462	226	525478817	9.780525e+12	John Green	
4	12	13335037	13335037	13155899	210	62024035	9.780062e+12	Veronica Roth	

5 rows × 23 columns

Filling null numeric data:

```
missing_values = data.isnull().sum()
print("Missing Values:\n", missing_values)

numeric_columns = data.select_dtypes(include=['number']).columns

data[numeric_columns] = data[numeric_columns].fillna(data[numeric_columns].median())
```

✓ 0.0s Python

```
Missing Values:
 book_id                      0
goodreads_book_id             0
best_book_id                  0
work_id                       0
books_count                   0
isbn                         52
isbn13                       44
authors                       0
original_publication_year     3
original_title               52
title                         0
language_code               109
average_rating                0
ratings_count                 0
work_ratings_count            0
work_text_reviews_count       0
```

Filling null categorical columns:

```
# Identify categorical columns
categorical_columns = data.select_dtypes(include=['object']).columns

# Impute missing values with mode for categorical columns only
data[categorical_columns] = data[categorical_columns].fillna(data[categorical_columns].mode().iloc[0])
```

[8] ✓ 0.0s Python

```
# All missing values handeled succesfully

missing_values = data.isnull().sum()
print("Missing Values:\n", missing_values)
```

[9] ✓ 0.0s Python

```
Missing Values:
 book_id                    0
goodreads_book_id           0
best_book_id                0
work_id                     0
books_count                 0
isbn                        0
isbn13                      0
authors                     0
original_publication_year   0
original_title              0
```

```
# Convert numeric columns to numeric data types
numeric_columns = data.select_dtypes(include=['number']).columns
data[numeric_columns] = data[numeric_columns].apply(pd.to_numeric)

# Convert categorical columns to categorical data types
categorical_columns = data.select_dtypes(include=['object']).columns
data[categorical_columns] = data[categorical_columns].astype('category')
```

[10] ✓ 0.0s Python

```
# Check the data types of each column
print(data.dtypes)
```

[11] ✓ 0.0s Python

```
book_id                     int64
goodreads_book_id           int64
best_book_id                int64
work_id                     int64
books_count                 int64
isbn                     category
isbn13                    float64
authors                  category
original_publication_year  float64
```

```python
# Check for duplicate rows
duplicate_rows = data.duplicated()

# Count the number of duplicate rows
num_duplicate_rows = duplicate_rows.sum()
print("Number of duplicate rows:", num_duplicate_rows)

# Remove duplicate rows
data = data.drop_duplicates()

# Reset the index after dropping rows
data.reset_index(drop=True, inplace=True)
```

[12] ✓ 0.0s Python

```
Number of duplicate rows: 0
```

```python
numerical_columns_subset = ['original_publication_year', 'average_rating', 'ratings_count']
categorical_columns_subset = ['language_code']

# Histograms for numerical variables
for column in numerical_columns_subset:
    plt.figure(figsize=(8, 6))
    sns.histplot(data[column], kde=True)
    plt.title(f'Histogram of {column}')
    plt.xlabel(column)
    plt.ylabel('Frequency')
    plt.show()

# Bar plots for categorical variables
for column in categorical_columns_subset:
    plt.figure(figsize=(8, 6))
    sns.countplot(data=data, x=column)
    plt.title(f'Bar Plot of {column}')
    plt.xlabel(column)
    plt.ylabel('Count')
    plt.xticks(rotation=45)
    plt.show()
```

✓ 1.1s Python

Let’s start with the haryy potter analysis output:

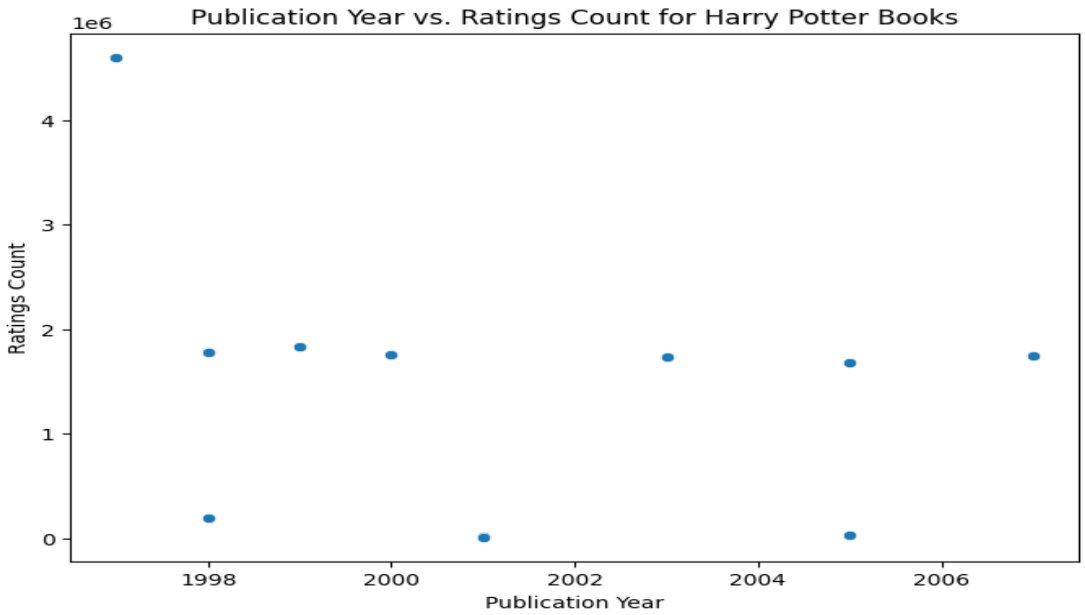
```
Number of Harry Potter books: 11

Harry Potter Books:
                                                    title
\
1     Harry Potter and the Sorcerer's Stone (Harry P...
6     Harry Potter and the Prisoner of Azkaban (Harr...
8     Harry Potter and the Order of the Phoenix (Har...
```

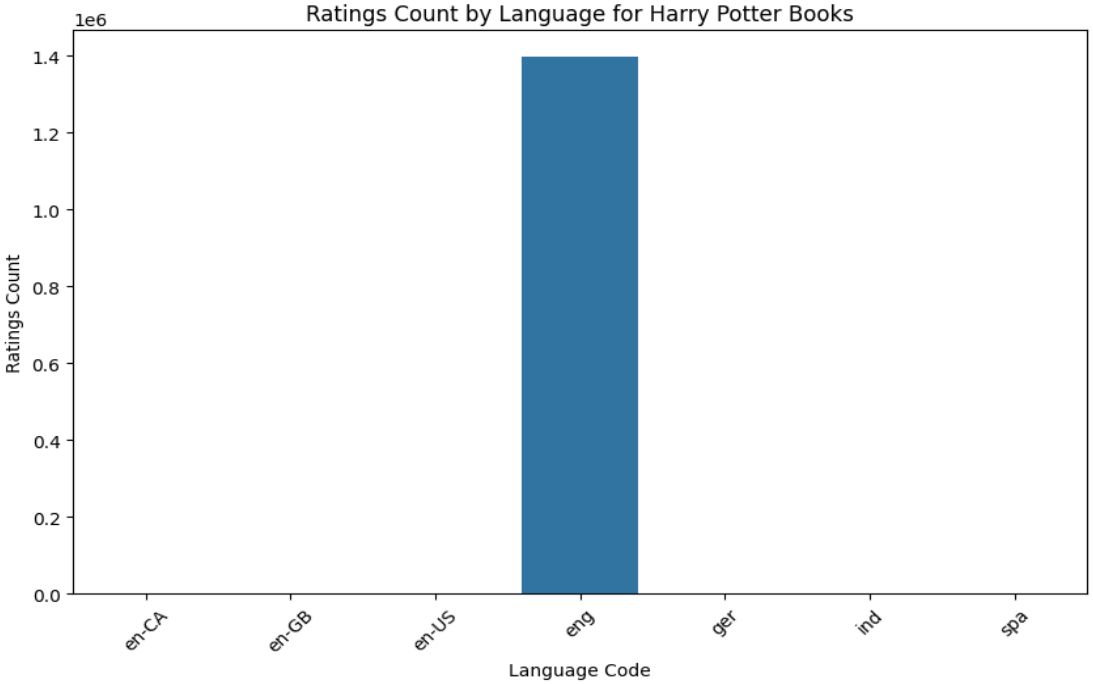
```
9     Harry Potter and the Chamber of Secrets (Harry...
10    Harry Potter and the Goblet of Fire (Harry Pot...
11    Harry Potter and the Deathly Hallows (Harry Po...
12    Harry Potter and the Half-Blood Prince (Harry ...
96              Harry Potter Boxset (Harry Potter, #1-7)
613         Harry Potter Collection (Harry Potter, #1-6)
1036 The Magical Worlds of Harry Potter: A Treasury...
1266 Harry Potter Schoolbooks Box Set: Two Classic ...

                                          authors
original_publication_year  \
1                  J.K. Rowling, Mary GrandPré
1997.0
6     J.K. Rowling, Mary GrandPré, Rufus Beck
1999.0
8                  J.K. Rowling, Mary GrandPré
2003.0
9                  J.K. Rowling, Mary GrandPré
1998.0
10                 J.K. Rowling, Mary GrandPré
2000.0
11                 J.K. Rowling, Mary GrandPré
2007.0
12                 J.K. Rowling, Mary GrandPré
2005.0
96                               J.K. Rowling
1998.0
```

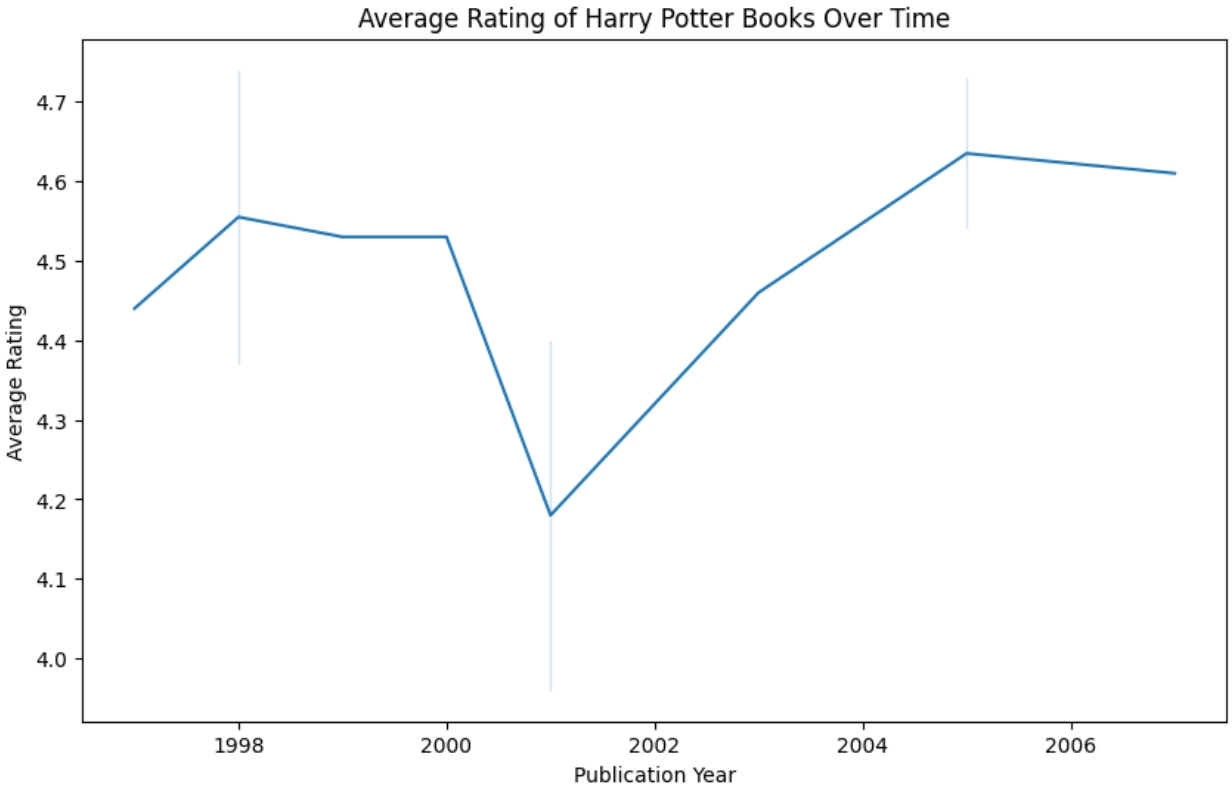
From this chart we can conclude that harry potter's first book was the best and took the highest ratings while the rest was always at the average ratings



All books are written in english



2001-2005 was harry potter’s production peak



Thank You 😊😊