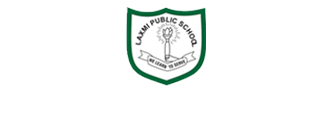
LAXMI PUBLIC SCHOOL

X-20, Institutional Area, Karkardooma, Delhi 110092



Session 2021-2022

A SYNOPSIS REPORT ON

**“MOVIE DATA ANALYSIS”**

Submitted to: Submitted by:

Ms. Jyotsna Upadhyay Lakshay Gupta

PROJECT LOGBOOK

PROJECT NAME- **MOVIE DATA ANALYSIS**

SCHOOL NAME- LAXMI PUBLIC SCHOOL

YEAR/CLASS- **2021-2022 / 12TH**

TEACHER NAME- MS. JYOTSNA UPADHYAY

INDEX

1. ACKNOWLEGMENT
2. CERTIFICATE
3. INTRODUCTION
4. TEAM ROLE AND PROJECT PLAN
5. DESCRIBE OF CSV FILE
6. PYTHON MATPLOTLIB & DATA VISUALIZATION
7. TOOLS
8. HARDWARE AND SOFTWARE USED
9. BIBLOGRAPHY/SOURCES

ACKNOWNLEGEMENT

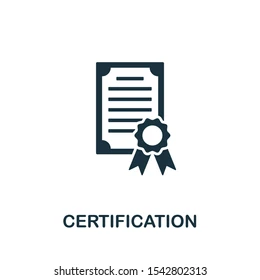
In the accomplishment of this project successfully, many people have best own upon me their blessing and heart pledged support on this time. I am utilizing to thank all the people who have been concerned with this project.

I would like to thank my Informatics practices teacher Ms. Jyotsna Upadhyay whose valuable guidance has been the ones that had me patch this project and make it full proof success his suggestion and his instruction has served as a major contributor to was the completion of the project.

I would like to thank My Parents and Friends who have helped me with their valuable suggestions and guidance has been helpful in various phases of the completion of the project.

CERTIFICATE

This to certify that Lakshay Gupta of XII-B, C of LAXMI PUBLIC SCHOOL has successfully completed the research on the Synopsis Project. Under the guidance of Ms. Jyotsna Upadhyay (Informatics Practices Teacher) during the year 2021-2022.



INTRODUCTION

A film, also called a movie, motion picture or moving picture, is a work of visual art used to simulate experiences that communicate ideas, stories, perceptions, feelings, beauty, or atmosphere through the use of moving images. These images are generally accompanied by sound, and more rarely, other sensory stimulations.

The word "cinema", short for cinematography, is often used to refer to filmmaking and the film industry, and to the art form that is the result of it.

The moving images of a film are created by photographing actual scenes with a motion-picture camera, by photographing drawings or miniature models using traditional animation techniques, by means of CGI and computer animation, or by a combination of some or all of these techniques, and other visual effects Traditionally, films were recorded onto celluloid film stock through a photochemical process and then shown through a movie projector onto a large screen.

PROJECT PLAN

|  |  |  |
| --- | --- | --- |
| ROLE | ROLE DESCRIPTION | NAME |
| FRONTEND  &  Backend | Python Coding   * Create Data Frame * CSV file * Various output   According to selected options  From the menu.   * Draw graph * Data Visualization * Collect data from * IMDb.com * Project documentation | LAKSHAY GUPTA |

**DESCRIBE OF CSV FILE**

A **comma-separated values** (**CSV**) file is a delimited text file that uses a comma to separate values. Each line of the file is a data record. Each record consists of one or more fields, separated by commas. The use of the comma as a field separator is the source of the name for this file format. A CSV file typically stores tabular data (numbers and text) in plain text, in which case each line will have the same number of fields.

The CSV file format is not fully standardized. Separating fields with commas is the foundation, but commas in the data or embedded line breaks have to be handled specially. Some implementations disallow such content while others surround the field with quotation marks, which yet again creates the need for escaping these if they are present in the data.

The term "CSV" also denotes several closely related delimiter-separated formats that use other field delimiters such as semicolons. These include tab-separated values and space-separated values. A delimiter guaranteed not to be part of the data greatly simplifies parsing.

Alternative delimiter-separated files are often given a .csv extension despite the use of a non-comma field separator. This loose terminology can cause problems in data exchange. Many applications that accept CSV files have options to select the delimiter character and the quotation character. Semicolons are often used instead of commas in many European locales in order to use the comma as the decimal separator and, possibly, the period as a decimal grouping character.

PYTHON MATPLOTLIB

A Python matplotlib script is structured so that a few lines of code are all that is required in most instances to generate a visual data plot. The matplotlib scripting layer overlays two APIs:

* The **pyplot** API is a hierarchy of Python code objects topped by **matplotlib.pyplot**
* An OO (Object-Oriented) API collection of objects that can be assembled with greater flexibility than pyplot. This API provides direct access to Matplotlib’s backend layers.

**Matplotlib and Pyplot in Python**

The pyplot API has a convenient MATLAB-style stateful interface. In fact, matplotlib was originally written as an open-source alternative for MATLAB. The OO API and its interface is more customizable and powerful than pyplot, but considered more difficult to use.

DATA VISUALIZATION

Data visualization is the graphical representation of information and data. By using [**visual elements like charts, graphs, and maps**](https://www.tableau.com/learn/articles/data-visualization/glossary)**,** data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.

In the world of Big Data, data visualization tools and technologies are essential to analyze massive amounts of information and make data-driven decisions.

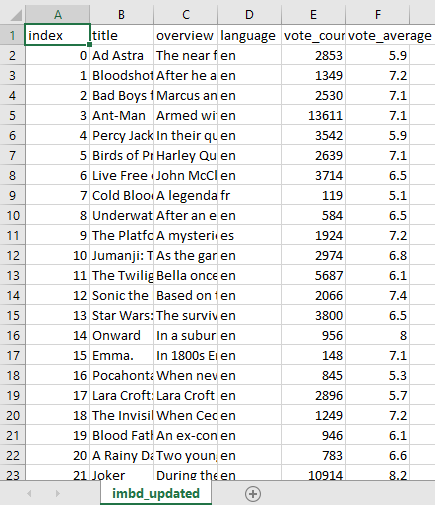
TOOLS

* Sublime Text 3 (Text Editor)
* Command Prompt (For initialize the programs to get the specific outputs and results)
* Microsoft Office Word (for making report)

HARDWARE AND SOFTWARE

* HARDWARE USED
* I3 6 GEN, 8GB RAM, WINDOWS 10
* MINIMUM REQ.
* INTEL ATOM, 2GB RAM, WINDOW 7
* SOFTWARE WE USED
* PYTHON 3.9, CMD, MS WORD, SUBLIME

CSV



SOURCE CODE

import pandas as pd

import numpy as np

import time

import matplotlib.pyplot as plt

import tempfile

import os

df = pd.DataFrame();

csv\_file = "C:/Users/hp/Desktop/PopularMovies/imbd\_updated.csv"

def introduction():

msg = '''

Movies are integral part of our life. We love to watch movies but it is very hard to find out good

movies from the world cinema.

TMDB.org is a crowd-sourced movie information database used by many film-related consoles,

sites and apps, such as XBMC, MythTV and Plex. Dozens of media managers, mobile apps and

social sites make use of its API.

TMDb lists some 80,000 films at time of writing, which is considerably fewer than IMDb.

While not as complete as IMDb, it holds extensive information for most popular/Hollywood films.

This is dataset of the 10,000 most popular movies across the world has been fetched through the

read API. TMDB's free API provides for developers and their team to programmatically fetch and

use TMDb's data. Their API is to use as long as you attribute TMDb as the source of the data

and/or images. Also, they update their API from time to time.

This data set is fetched using exception handling process so the data set contains some null

values as there are missing fields in the tmdb database. Thought it's good for a young analyst

to deal with messing value.

In this project we are going to analyse the same dataset using Python Pandas on windows machine but the

project can be run on any machine support Python and Pandas. Besides pandas we also used matplotlib python

module for visualization of this dataset.

The whole project is divided into four major parts ie reading, analysis, visualization and export. all these

part are further divided into menus for easy navigation

\n\n\n\n'''

for x in msg:

print(x, end='')

time.sleep(0.002)

wait = input('Press any key to continue.....')

def made\_by():

msg = '''

Movie Analysis Made By: Lakshay, Anmol, Farhdeen

Roll No : 14

School Name : Laxmi Public School

session : 2021-22

Thanks for evaluating my Project.

\n\n\n

'''

for x in msg:

print(x, end='')

time.sleep(0.002)

wait = input('Press any key to continue.....')

def read\_csv\_file():

df = pd.read\_csv(csv\_file)

print(df)

# name of function : clear

# purpose : clear output screen

def clear():

for x in range(65):

print()

def data\_analysis\_menu():

df = pd.read\_csv(csv\_file)

while True:

clear()

print('\n\nData Analysis MENU ')

print('\_'\*100)

print('1. Show Whole DataFrame')

print('2. Show Columns')

print('3. Show Top Rows')

print('4. Row Bottom Rows')

print('5. Show Specific Column\n')

print('6. Add a New Record\n')

print('7. Add a New Column\n')

print('8. Delete a Column\n')

print('9. Delete a Record\n')

print('10. Update a Record\n')

print('11. Rating Wise Report \n')

print('12. Language wise Report \n')

print('13. Data Summery\n')

print('14. Exit (Move to main menu)\n')

ch = int(input('Enter your choice:'))

if ch == 1:

print(df)

wait = input()

if ch == 2:

print(df.columns)

wait = input()

if ch == 3:

n = int(input('Enter Total rows you want to show :'))

print(df.head(n))

wait = input()

if ch == 4:

n = int(input('Enter Total rows you want to show :'))

print(df.tail(n))

wait = input()

if ch == 5:

print(df.columns)

col\_name = input('Enter Column Name that You want to print : ')

print(df[col\_name])

wait = input()

if ch==6:

a = input('Enter Index Number :')

b = input('Enter New Movie Name :')

c = input(' Enter Movie Overview :')

d= input('Enter Movie Language :')

e = int(input('Enter Vote Count :'))

f = float(input('Enter Vote Average :'))

data={'index':a,'title':b,'overview':c,'language':d,'vote\_count':e,'vote\_average':f}

df = df.append(data,ignore\_index=True)

print(df)

wait=input()

if ch==7:

col\_name = input('Enter new column name :')

col\_value = int(input('Enter default column value :'))

df[col\_name]=col\_value

print(df)

print('\n\n\n Press any key to continue....')

wait=input()

if ch==8:

col\_name =input('Enter column Name to delete :')

del df[col\_name]

print(df)

print('\n\n\n Press any key to continue....')

wait=input()

if ch==9:

index\_no =int(input('Enter the Index Number that You want to delete :'))

df = df.drop(df.index[index\_no])

print(df)

print('\n\n\n Press any key to continue....')

wait = input()

#update a record - this is to be cover

if ch == 10:

index\_no = int(

input('Enter the Index Number that You want to update :'))

df = df.drop(df.index[index\_no])

print(df)

print('\n\n\n Press any key to continue....')

wait = input()

if ch==11:

g = df.sort\_values(by=['vote\_average','vote\_count'],ascending=False)

clear()

print('Top 20 Movies Based on Rating')

print('-'\*120)

print(g.head(20))

print('\n\n\n Press any key to continue....')

wait=input()

if ch==12:

df1=df.language.unique()

print('Available Languages :',df1)

print('\n\n')

lang1 =input('Enter Language Type :')

df1=df[df.language==lang1]

clear()

print('Top 20 Movies Based on Rating | Language :',lang1)

print('-'\*120)

print(df1.sort\_values(by='vote\_average', ascending=False).head(20))

print('\n\n\n Press any key to continue....')

wait = input()

if ch==13:

print(df.describe())

print("\n\n\nPress any key to continue....")

wait=input()

if ch == 14:

break

# name of function : graph

# purpose : To generate a Graph menu

def graph():

df = pd.read\_csv(csv\_file)

while True:

clear()

print('\nGRAPH MENU ')

print('\_'\*100)

print('1. Whole Data LINE Graph\n')

print('2. Whole Data Bar Graph\n')

print('3. Whole Data Bar Graph- Horizontal\n')

print('4. Exit (Move to main menu)\n')

ch = int(input('Enter your choice:'))

if ch == 1:

g = df.groupby('language')

x = df['language'].unique()

y = g['language'].count()

plt.xticks(rotation='vertical')

plt.xlabel('Language')

plt.ylabel('Total Movies')

plt.title('Language wise movies count')

plt.grid(True)

plt.plot(x, y)

plt.show()

if ch == 2:

g = df.groupby('language')

x = df['language'].unique()

y = g['language'].count()

#plt.xticks(rotation='vertical')

plt.xlabel('Language')

plt.ylabel('Total Movies')

plt.title('Language wise movies count')

plt.bar(x, y)

plt.grid(True)

plt.show()

if ch == 3:

g = df.groupby('language')

x = df['language'].unique()

y = g['language'].count()

#plt.xticks(rotation='vertical')

plt.xlabel('Language')

plt.ylabel('Total Movies')

plt.title('Language wise movies count')

plt.barh(x, y)

plt.grid(True)

plt.show()

if ch == 4:

break

# function name : export\_menu

# purpose : function to generate export menu

def export\_menu():

df = pd.read\_csv(csv\_file)

while True:

clear()

print('\n\nEXPORT MENU ')

print('\_'\*100)

print()

print('1. Text File\n')

print('2. Excel File\n')

print('4. Exit (Move to main menu)')

ch = int(input('Enter your Choice : '))

if ch == 1:

df.to\_csv('C:/Users/hp/Project Export/data.txt')

print('\n\nCheck your new file "newMovies.csv" on C: Drive.....')

wait = input()

if ch == 2:

df.to\_csv('C:/Users/hp/Project Export/data.csv')

print('\n\nCheck your new file "newMovies.xlsx" on C: Drive.....')

wait = input()

if ch == 3:

break

def main\_menu():

clear()

introduction()

while True:

clear()

print('MAIN MENU ')

print('\_'\*100)

print()

print('1. Read CSV File\n')

print('2. Data Analysis Menu\n')

print('3. Graph Menu\n')

print('4. Export Data\n')

print('5. Exit\n')

choice = int(input('Enter your choice :'))

if choice == 1:

read\_csv\_file()

wait = input()

if choice == 2:

data\_analysis\_menu()

wait = input()

if choice == 3:

graph()

wait = input()

if choice == 4:

export\_menu()

wait = input()

if choice == 5:

break

clear()

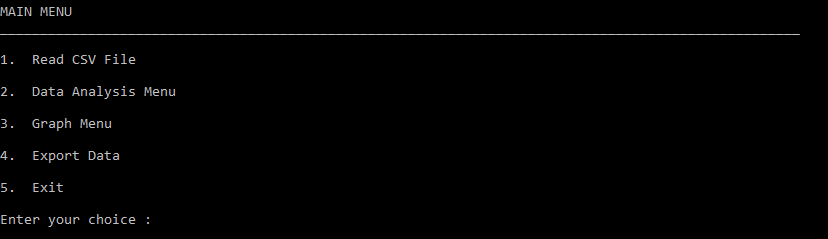
made\_by()

# call your main menu

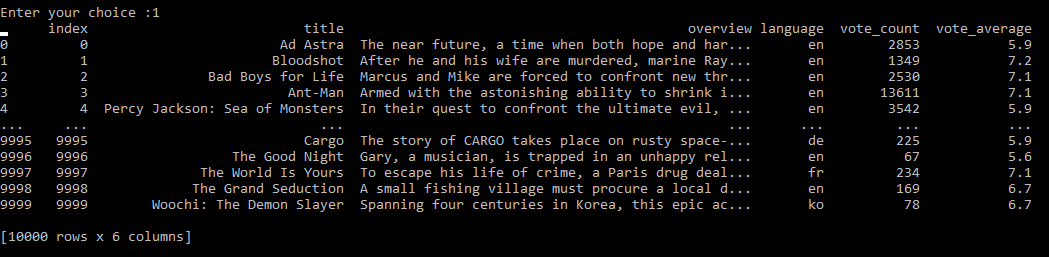
main\_menu()

OUTPUT SCREEN

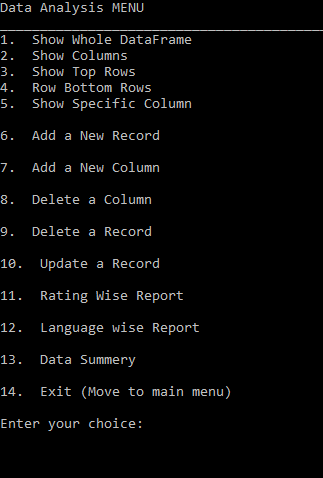
MAIN SCREEN



1. READ CSV FILE



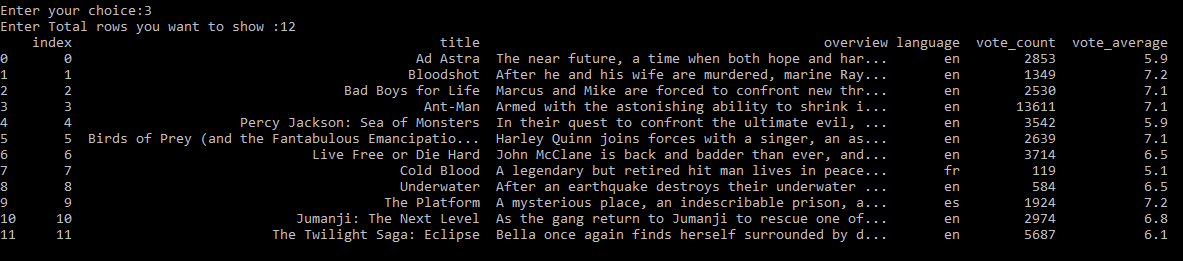
2. DATA ANALYSIS MENU



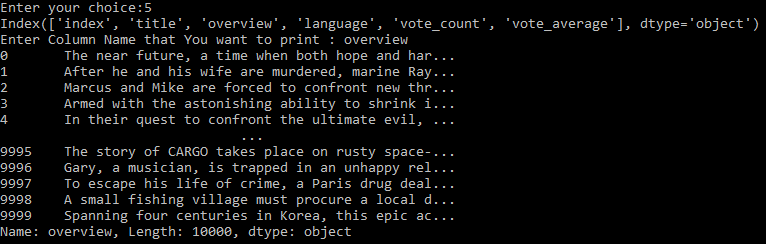
SHOW COLUMNS



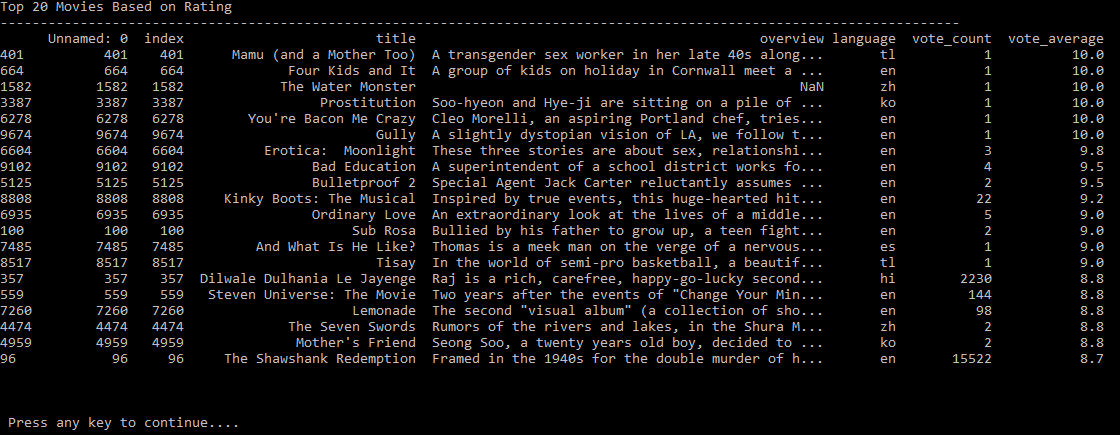
SHOW TOP ROWS NO.12



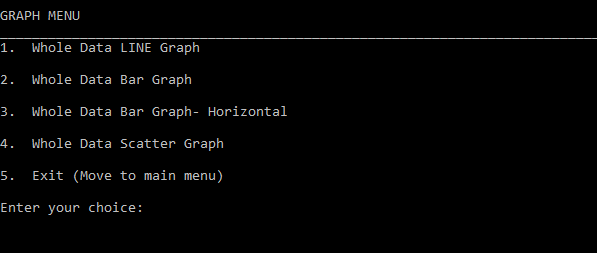
SHOW SPECIFIC COLUMN (Overview)



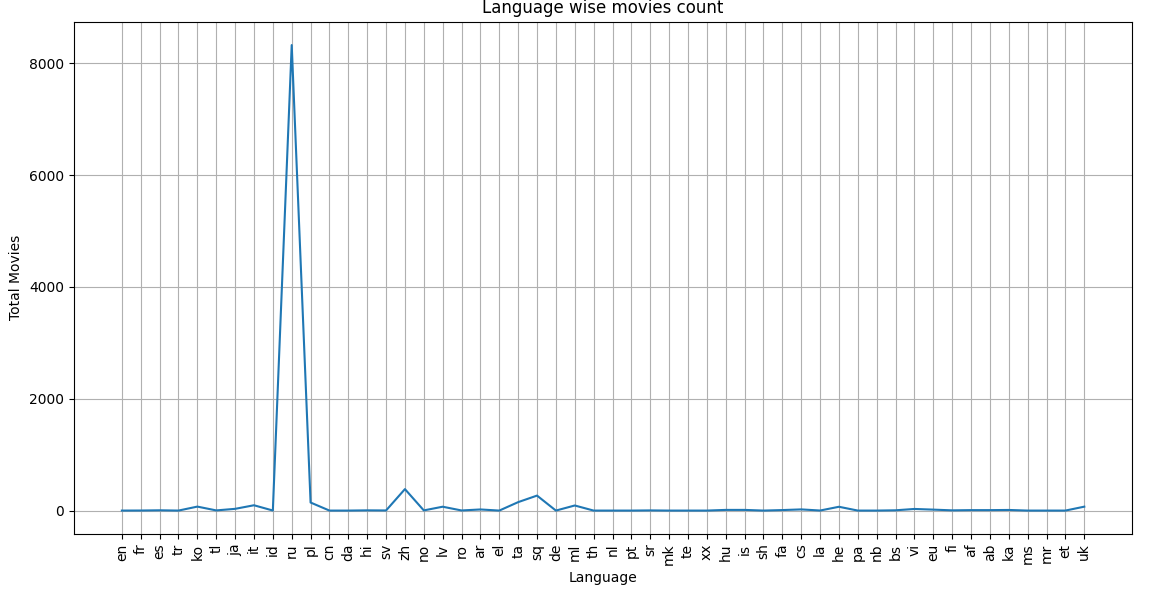
Rating Wise Report



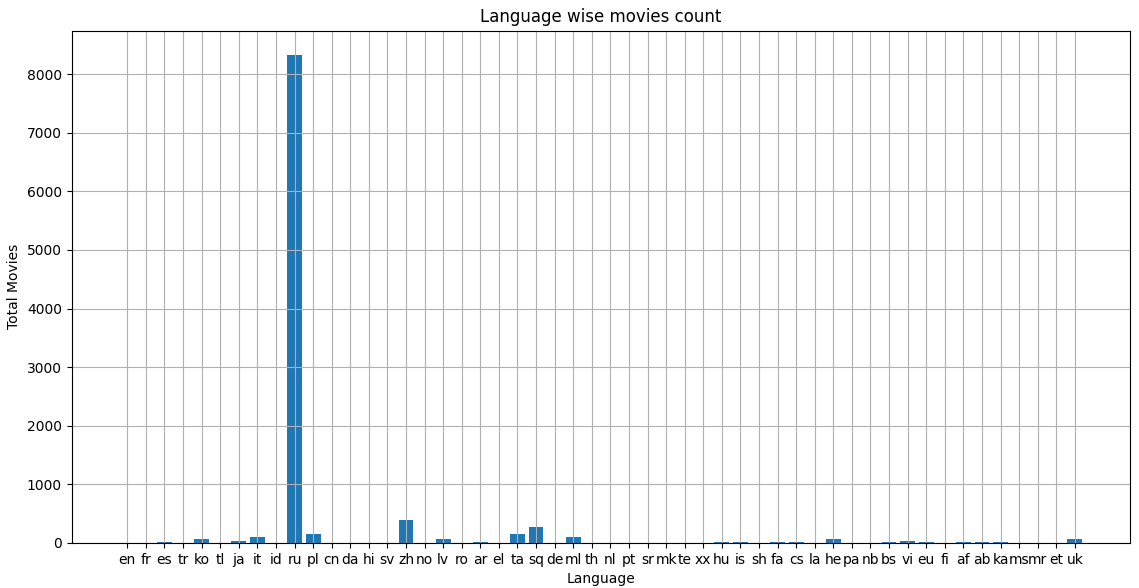
GRAPH MENU



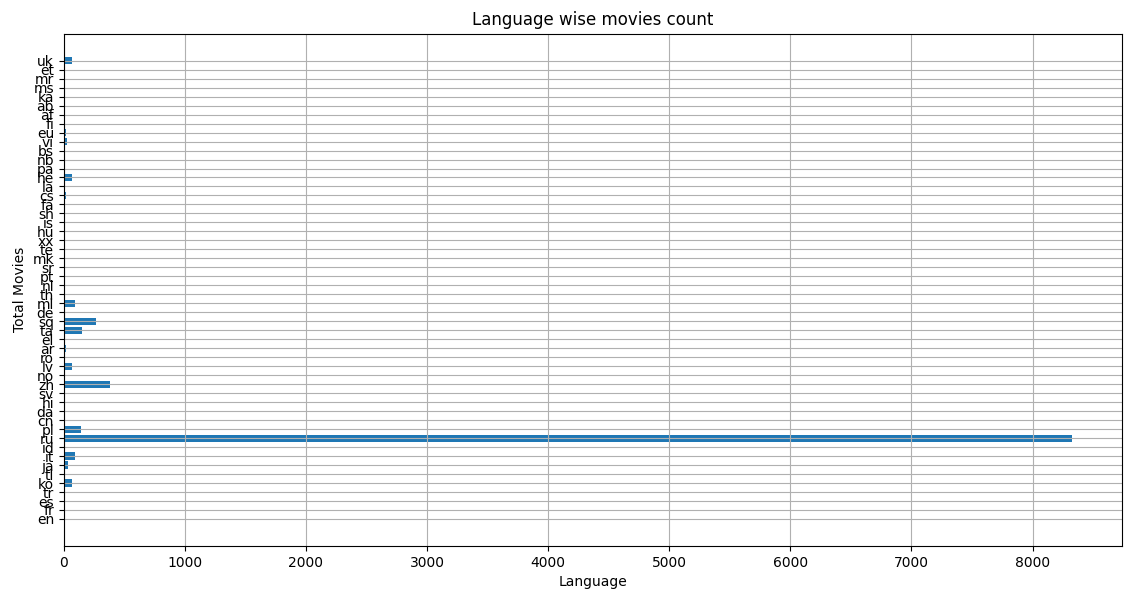
WHOLE DATA LINE GRAPH



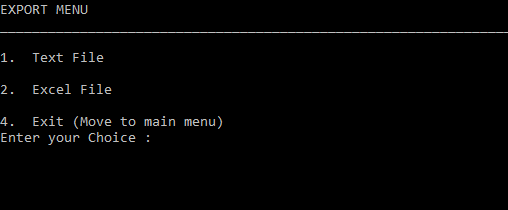
WHOLE DATA BAR GRAPH



WHOLE DATA BAR GRAPH- HORIZONTAL

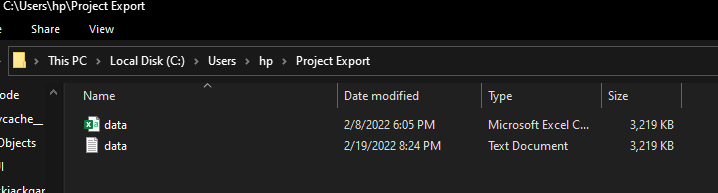


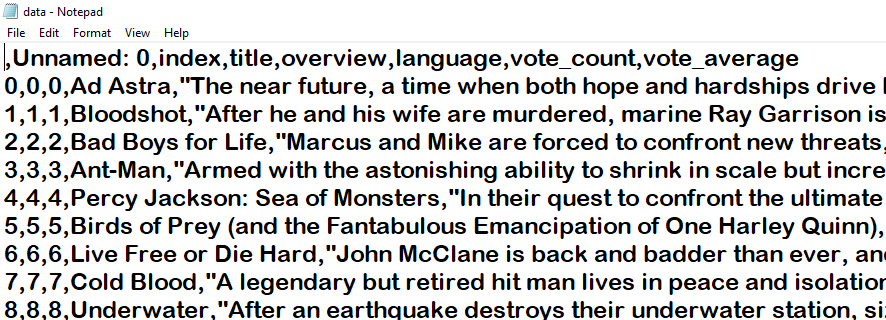
4. EXPORT DATA

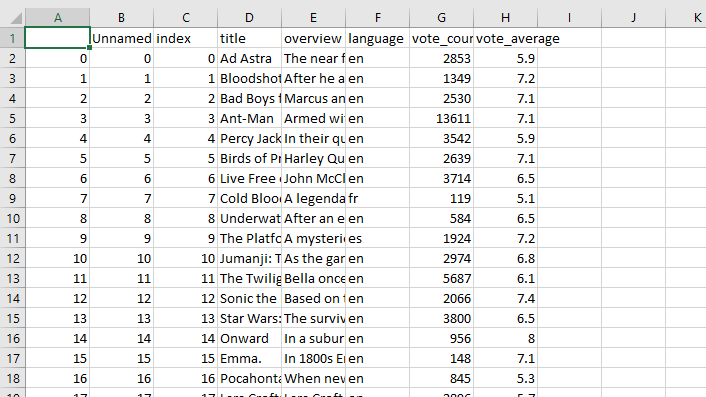


TEXT FILE









BIBLOGRAPHY/SOURCES

1. <https://www.wikipedia.org>

2. <https://www.imdb.com>

3. <https://www.google.co.in>

4. <https://stackoverflow.com>