Find_me.py

Explanaition:

This is meant to show a user how to casually look up the information based on a basic search from the four total groups (Books, DVD, Music, Video) as well as entering in a potential category (there are many, but some categories could break the code. Please stick to inputing common things like Comedy and Drama).

In [1]:
```
#find_me.py

import pandas as pd

def find_items(group, category, num_items=5):
    df = pd.read_json('/Users/spencer/Desktop/CSV, JSON, Excel and txt of Amazon Final P
    filtered_df = df[df['group'] == group]

    # Check if 'categories' column exists and is a string before applying 'contains'
    if 'categories' in df.columns and category:
        filtered_df = filtered_df[filtered_df['categories'].astype(str).str.contains(cat

    item_titles = filtered_df['title'].head(num_items).tolist()
    return item_titles

if __name__ == "__main__":
    group = input("Enter the group (e.g., Book, DVD, Music, Video): ")
    category = input("Enter the category (leave empty if not applicable): ")
    items = find_items(group, category)
    print("What we found for you:")
    for item in items:
        print(item)
```

```
Enter the group (e.g., Book, DVD, Music, Video): Book
Enter the category (leave empty if not applicable):
What we found for you:
Patterns of Preaching: A Sermon Sampler
Candlemas: Feast of Flames
World War II Allied Fighter Planes Trading Cards
Life Application Bible Commentary: 1 and 2 Timothy and Titus
Prayers That Avail Much for Business: Executive
```

bookssimilar.py

Explanation:

This runs code to look and see the top 50 books recommended and prints it into an excel file as well as onto the terminal. I will connect the excel files to my project so you can see what was derived from this code.

In [2]:
```
#bookssimilar.py

import pandas as pd
import re
import os

def extract_review_info(reviews_str):
    num_ratings_match = re.search(r'total: (\d+)', reviews_str)
    avg_rating_match = re.search(r'avg rating: (\d)', reviews_str)
    num_ratings = int(num_ratings_match.group(1)) if num_ratings_match else 0
    avg_rating = int(avg_rating_match.group(1)) if avg_rating_match else 0
    return num_ratings, avg_rating

df = pd.read_json('/Users/spencer/Desktop/CSV, JSON, Excel and txt of Amazon Final Proje
books_df = df[df['group'] == 'Book'].copy()
asin_to_title = pd.Series(books_df.title.values, index=books_df.ASIN).to_dict()


def replace_asin_with_title(similar_list):
    if isinstance(similar_list, list) and len(similar_list) > 1:
        return [asin_to_title.get(asin, asin) for asin in similar_list[1:]]
    return []


books_df.loc[:, 'similar_titles'] = books_df['similar'].apply(replace_asin_with_title)
books_info = pd.DataFrame()
books_info['title'] = books_df['title']
books_info['similar'] = books_df['similar_titles'].apply(lambda x: ', '.join(x))
books_info[['num_ratings', 'avg_rating']] = books_df.apply(
    lambda row: extract_review_info(row['reviews']),
    axis=1, result_type='expand'
)

output_dir = '/Users/spencer/Desktop/CS-458-Fall-2023/Amazon Final Project/Excel'
if not os.path.exists(output_dir):
    os.makedirs(output_dir)
top_books = books_info.sort_values(by=['avg_rating', 'num_ratings'], ascending=[False, F
print(top_books)
top_books.to_excel('/Users/spencer/Desktop/CS-458-Fall-2023/Amazon Final Project/Excel/t
```

```
                                                    title  \
91158      Harry Potter and the Sorcerer's Stone (Book 1)
211463  Harry Potter and the Sorcerer's Stone (Book 1 ...
250879  Harry Potter and the Sorcerer's Stone (Book 1 ...
428073  Harry Potter and the Sorcerer's Stone (Book 1,...
526761     Harry Potter and the Sorcerer's Stone (Book 1)
546259  Harry Potter and the Sorcerer's Stone (Book 1,...
23792               Harry Potter and the Sorcerer's Stone
148185     Harry Potter and the Sorcerer's Stone (Book 1)
99487      Harry Potter and the Sorcerer's Stone (Book 1)
84842   Harry Potter and the Goblet of Fire (Book 4 Au...
311376       Harry Potter and the Goblet of Fire (Book 4)
429123       Harry Potter and the Goblet of Fire (Book 4)
527309       Harry Potter and the Goblet of Fire (Book 4)
544342  Harry Potter and the Goblet of Fire (Book 4, A...
128673  Harry Potter and the Goblet of Fire (Book 4 Au...
4739         Harry Potter and the Goblet of Fire (Book 4)
77140   Harry Potter And The Chamber Of Secrets: Colle...
216201   Harry Potter and the Chamber of Secrets (Book 2)
415840   Harry Potter and the Chamber of Secrets (Book 2)
441377            Harry Potter and the Chamber of Secrets
449174   Harry Potter and the Chamber of Secrets (Book 2)
527636  Harry Potter and the Chamber of Secrets (Harry...
534181  Harry Potter and the Chamber of Secrets (Thorn...
544883  Harry Potter and the Chamber of Secrets (Book ...
45703   Harry Potter and the Chamber of Secrets, Brail...
178549  Harry Potter and the Chamber of Secrets (Book ...
189465  Harry Potter and the Chamber of Secrets (Book ...
413858  Harry Potter and the Prisoner of Azkaban (Book...
490274           Harry Potter and the Prisoner of Azkaban
519412           Harry Potter and the Prisoner of Azkaban
35063   Harry Potter and the Prisoner of Azkaban (Book 3)
56100   Harry Potter and the Prisoner of Azkaban (Book 3)
103983  Harry Potter and the Prisoner of Azkaban (Book...
186578  Harry Potter and the Prisoner of Azkaban (Harr...
390076  Harry Potter and the Prisoner of Azkaban (Book...
404634  Harry Potter and the Prisoner of Azkaban (Book 3)
281763                                     Ella Enchanted
42858              Ella Enchanted (rpkg) (Trophy Newbery)
257396                Ella Enchanted (Newbery Honor Book)
350072                                     Ella Enchanted
59288   Taking Charge of Your Fertility: The Definitiv...
76408   Taking Charge of Your Fertility: The Definitiv...
203588  Taking Charge of Your Fertility: The Definitiv...
244882   Have a Nice Day!: A Tale of Blood and Sweatsocks
219819    Have a Nice Day! A Tale of Blood and Sweatsocks
402359   Have a Nice Day : A Tale of Blood and Sweatsocks
218311               Chronicles of Narnia Audio Collection
233808  The Chronicles of Narnia Box Set: Full-Color C...
242863  The Chronicles of Narnia: The Magician's Nephe...
20662                  The Chronicles of Narnia Boxed Set

                                                  similar  num_ratings  \
91158   Harry Potter and the Chamber of Secrets (Book ...         5039
211463  Harry Potter and the Chamber of Secrets (Book ...         5039
250879  Harry Potter and the Chamber of Secrets (Book ...         5039
428073  Harry Potter and the Chamber of Secrets (Book ...         5039
526761  Harry Potter and the Chamber of Secrets (Book ...         5039
546259  Harry Potter and the Chamber of Secrets (Book ...         5039
23792   Harry Potter and the Chamber of Secrets (Book ...         5034
148185  Harry Potter and the Chamber of Secrets (Book ...         5034
99487   Harry Potter and the Chamber of Secrets (Book ...         5033
84842   Harry Potter and the Prisoner of Azkaban (Book...         4924
311376  Harry Potter and the Prisoner of Azkaban (Book...         4924
429123  Harry Potter and the Prisoner of Azkaban (Book...         4924
527309  Harry Potter and the Prisoner of Azkaban (Book...         4924
544342  Harry Potter and the Prisoner of Azkaban (Book...         4924
128673  Harry Potter and the Prisoner of Azkaban (Book...         4922
4739    Harry Potter and the Prisoner of Azkaban (Book...         4921
77140   Harry Potter and the Goblet of Fire (Book 4), ...         2422
216201  Harry Potter and the Goblet of Fire (Book 4), ...         2422
415840  Harry Potter and the Goblet of Fire (Book 4), ...         2422
441377  Harry Potter and the Goblet of Fire (Book 4), ...         2422
449174  Harry Potter and the Goblet of Fire (Book 4), ...         2422
527636  Harry Potter and the Goblet of Fire (Book 4), ...         2422
534181  Harry Potter and the Goblet of Fire (Book 4), ...         2422
544883  Harry Potter and the Goblet of Fire (Book 4), ...         2422
45703   Harry Potter and the Goblet of Fire (Book 4), ...         2419
178549  Harry Potter and the Goblet of Fire (Book 4), ...         2419
189465  Harry Potter and the Goblet of Fire (Book 4), ...         2419
413858  Harry Potter and the Chamber of Secrets (Book ...         2402
490274  Harry Potter and the Chamber of Secrets (Book ...         2402
519412  Harry Potter and the Chamber of Secrets (Book ...         2402
35063   Harry Potter and the Chamber of Secrets (Book ...         2396
56100   Harry Potter and the Chamber of Secrets (Book ...         2396
103983  Harry Potter and the Chamber of Secrets (Book ...         2396
186578  Harry Potter and the Chamber of Secrets (Book ...         2396
390076  Harry Potter and the Chamber of Secrets (Book ...         2396
404634  Harry Potter and the Chamber of Secrets (Book ...         2396
281763  The Two Princesses of Bamarre, The Princess Ta...         1004
42858   The Two Princesses of Bamarre, The Princess Ta...         1001
257396  The Two Princesses of Bamarre, The Princess Ta...         1001
350072  The Two Princesses of Bamarre, The Princess Ta...         1001
59288   B0000533AY, The Fastest Way To Get Pregnant Na...          701
76408   B0000533AY, The Fastest Way To Get Pregnant Na...          701
203588  B0000533AY, The Fastest Way To Get Pregnant Na...          701
244882  Foley is Good: And the Real World is Faker Tha...          673
219819  Foley is Good: And the Real World is Faker Tha...          671
402359  Foley is Good: And the Real World is Faker Tha...          671
218311  A Wrinkle in Time, 1581345151, The Complete C....          610
233808  A Wrinkle in Time, 1581345151, The Complete C....          610
242863  A Wrinkle in Time, 1581345151, The Complete C....          610
```

```
20662   A Wrinkle in Time, 1581345151, The Complete C....          609

        avg_rating
91158            5
211463           5
250879           5
428073           5
526761           5
546259           5
23792            5
148185           5
99487            5
84842            5
311376           5
429123           5
527309           5
544342           5
128673           5
4739             5
77140            5
216201           5
415840           5
441377           5
449174           5
527636           5
534181           5
544883           5
45703            5
178549           5
189465           5
413858           5
490274           5
519412           5
35063            5
56100            5
103983           5
186578           5
390076           5
404634           5
281763           5
42858            5
257396           5
350072           5
59288            5
76408            5
203588           5
244882           5
219819           5
402359           5
218311           5
233808           5
242863           5
20662            5
```

dvdsimilar.py

Explanation:

This runs code to look and see the top 50 dvds recommended and prints it into an excel file as well as onto the terminal. I will connect the excel files to my project so you can see what was derived from this code.

In [3]:
```
#dvdsimilar.py
```

```python
import pandas as pd
import re
import os

def extract_review_info(reviews_str):
    num_ratings_match = re.search(r'total: (\d+)', reviews_str)
    avg_rating_match = re.search(r'avg rating: (\d)', reviews_str)
    num_ratings = int(num_ratings_match.group(1)) if num_ratings_match else 0
    avg_rating = int(avg_rating_match.group(1)) if avg_rating_match else 0

    return num_ratings, avg_rating

df = pd.read_json('/Users/spencer/Desktop/CSV, JSON, Excel and txt of Amazon Final Proje
dvds_df = df[df['group'] == 'DVD'].copy()
asin_to_title = pd.Series(dvds_df.title.values, index=dvds_df.ASIN).to_dict()


def replace_asin_with_title(similar_list):
    if isinstance(similar_list, list) and len(similar_list) > 1:
        return [asin_to_title.get(asin, asin) for asin in similar_list[1:]]
    return []


dvds_df.loc[:, 'similar_titles'] = dvds_df['similar'].apply(replace_asin_with_title)
dvds_info = pd.DataFrame()
dvds_info['title'] = dvds_df['title']
dvds_info['similar'] = dvds_df['similar_titles'].apply(lambda x: ', '.join(x))
dvds_info[['num_ratings', 'avg_rating']] = dvds_df.apply(
    lambda row: extract_review_info(row['reviews']),
    axis=1, result_type='expand'
)

output_dir = '/Users/spencer/Desktop/CS-458-Fall-2023/Amazon Final Project/Excel'
if not os.path.exists(output_dir):
    os.makedirs(output_dir)
top_dvds = dvds_info.sort_values(by=['avg_rating', 'num_ratings'], ascending=[False, Fal
print(top_dvds)
top_dvds.to_excel('/Users/spencer/Desktop/CS-458-Fall-2023/Amazon Final Project/Excel/to
```

```
                                                     title  \
117918                        Ever After - A Cinderella Story
532304                        Ever After - A Cinderella Story
174839                        Labyrinth (Superbit Collection)
548411                                              Labyrinth
137401                                      Band of Brothers
543904                               The Shawshank Redemption
270548                          Newsies (Collector's Edition)
498218                                           Led Zeppelin
407345                                            Toy Story 2
425186    Toy Story - The Ultimate Toy Box (Collector's ...
425188                         Toy Story & Toy Story 2 (2 Pack)
216092                Grave of the Fireflies (Collector's Edition)
475503                                 Grave of the Fireflies
538181                                       N Sync - N the Mix
261416                                         The Iron Giant
497095                                 Glory (Special Edition)
268181    Buffy the Vampire Slayer - The Complete Third ...
161152                            Eddie Izzard - Dress to Kill
528823                           Anne of Green Gables [IMPORT]
100059                                    Anne of Green Gables
545832                       Rear Window (Collector's Edition)
473447             La Femme Nikita - The Complete First Season
545414                                            October Sky
240124                                    The Beatles Anthology
544989                                        The Temptations
546743                                        The Temptations
```

```
546713                     The Good, the Bad and the Ugly
313570                                       Into the Woods
441282                           From the Earth to the Moon
5916                           Backstreet Boys - All Access
283913    The Incredible Adventures of Wallace and Gromit
70331            M*A*S*H - Season One (Collector's Edition)
547113                                    Strictly Ballroom
277045                                        All About Eve
544551                      All About Eve (Special Edition)
546357     Sunset Boulevard (Special Collector's Edition)
483793                                            Tommy Boy
530018                             Cowboy Bebop - Session 1
283674  The Adventures of Robin Hood (Two-Disc Special...
21355                                      Midnight Madness
37547               Cosmos Boxed Set (Collector's Edition)
143162         9/11 - The Filmmakers' Commemorative Edition
44238                                           Being There
53582      Love and Basketball (New Line Platinum Series)
211239                   Lagaan - Once Upon a Time in India
544582                          Diana Krall - Live in Paris
82978                                          12 Angry Men
255202               Cardcaptor Sakura - The Clow (Vol. 1)
372386           Once and Again - The Complete First Season
544949                  Koyaanisqatsi - Life Out of Balance

                                                  similar  num_ratings  \
117918  Never Been Kissed, While You Were Sleeping, 10...          817
532304  Never Been Kissed, While You Were Sleeping, Th...          817
174839  The Dark Crystal, The NeverEnding Story, Legen...          805
548411  The Dark Crystal, The NeverEnding Story, Legen...          805
137401  B0001NBLVI, B00003CXCT, B000634DCW, 074322454X...          786
543904  B00003CXCT, The Green Mile, B00005JMQW, B00005...          757
270548  B000056QE6, Swing Kids, B00009MEJC, Little Wom...          524
498218  Led Zeppelin - The Song Remains the Same, B000...          510
407345  Toy Story, A Bug's Life (Collector's Edition),...          473
425186  Toy Story, A Bug's Life (Collector's Edition),...          473
425188  Toy Story, A Bug's Life (Collector's Edition),...          473
216092  Princess Mononoke, B0001XAPZ6, My Neighbor Tot...          465
475503  Princess Mononoke, B0001XAPZ6, My Neighbor Tot...          465
538181  'N Sync - Most Requested Hit Videos, N Sync - ...          409
261416  B00005JN4W, B00020SK1Y, B00005JMQW, 0375801537...          351
497095  Gettysburg (Widescreen Edition), Amistad, B000...          294
268181  Buffy the Vampire Slayer - The Complete Second...          274
161152  B0000ALA3P, B00069PCDA, B00065HKG6, B00065HKFW...          262
528823  B00005Y7AN, B00005Y7AM, 0553609416, Anne of Gr...          259
100059  B00005Y7AN, B00005Y7AM, 0553609416, Anne of Gr...          257
545832  Vertigo (Collector's Edition), North by Northw...          247
473447  B000777HRA, B0008ENIR0, B0001I55ZQ, B000007Q6J...          225
545414  0385333218, 0783219695, 0440237165, Mr. Hollan...          225
240124  A Hard Day's Night, B0000CEB4V, Yellow Submari...          223
544989  The Five Heartbeats, Standing In The Shadows o...          221
546743  The Five Heartbeats, Standing In The Shadows o...          221
546713  A Fistful of Dollars, B0000AUHPG, For A Few Do...          217
313570  B00005JL6V, Sunday in the Park with George, B0...          187
441282  For All Mankind - Criterion Collection, The Ri...          180
5916    Backstreet Boys - Homecoming: Live in Orlando,...          179
283913  Creature Comforts, Chicken Run, B00004W3HB, B0...          169
70331   M*A*S*H - Season Two (Collector's Edition), M*...          166
547113  B0002V7S34, Dance with Me, Burn the Floor, B00...          164
277045  Sunset Boulevard (Special Collector's Edition)...          162
544551  Sunset Boulevard (Special Collector's Edition)...          162
546357  All About Eve (Special Edition), Chinatown, Ci...          162
483793  Black Sheep, Dumb and Dumber, Beverly Hills Ni...          160
530018  B0001BMM4K, B0001BMM4U, B0001BMM54, B0001BMM5O...          152
283674  The Mark of Zorro, B00005JMR7, B0007OY2PS, B00...          151
21355   B00009AOBK, B0001WTVGG, B0003JAONG, B00076ONW8...          147
```

```
37547   B0000ZG0TA, Stargaze - Hubble's View Of The Un...          145
143162  In Memoriam - New York City, 9/11/01, Remember...          145
44238   The Party, Dr. Strangelove or How I Learned to...          140
53582   The Wood, The Best Man, Brown Sugar, Love Jone...          140
211239  Monsoon Wedding, Asoka, Salaam Bombay! (Widesc...          140
544582  B00067OLN4, Norah Jones - Live in New Orleans,...          137
82978   Inherit the Wind, To Kill a Mockingbird (Colle...          131
255202  Cardcaptor Sakura - Everlasting Memories (Vol....          131
372386  B0009F43FY, B0002DB0FO, B0001IN0T4, B0007OY2MG...          130
544949  Powaqqatsi - Life in Transformation, Baraka (S...          127

        avg_rating
117918           5
532304           5
174839           5
548411           5
137401           5
543904           5
270548           5
498218           5
407345           5
425186           5
425188           5
216092           5
475503           5
538181           5
261416           5
497095           5
268181           5
161152           5
528823           5
100059           5
545832           5
473447           5
545414           5
240124           5
544989           5
546743           5
546713           5
313570           5
441282           5
5916             5
283913           5
70331            5
547113           5
277045           5
544551           5
546357           5
483793           5
530018           5
283674           5
21355            5
37547            5
143162           5
44238            5
53582            5
211239           5
544582           5
82978            5
255202           5
372386           5
544949           5
```

musicsimilar.py

Explanation:

This runs code to look and see the top 50 music recommended and prints it into an excel file as well as onto the terminal. I will connect the excel files to my project so you can see what was derived from this code.

In [4]:
```python
#musicsimilar.py

import pandas as pd
import re
import os

def extract_review_info(reviews_str):
    num_ratings_match = re.search(r'total: (\d+)', reviews_str)
    avg_rating_match = re.search(r'avg rating: (\d)', reviews_str)
    num_ratings = int(num_ratings_match.group(1)) if num_ratings_match else 0
    avg_rating = int(avg_rating_match.group(1)) if avg_rating_match else 0

    return num_ratings, avg_rating

df = pd.read_json('/Users/spencer/Desktop/CSV, JSON, Excel and txt of Amazon Final Proje
musics_df = df[df['group'] == 'Music'].copy()
asin_to_title = pd.Series(musics_df.title.values, index=musics_df.ASIN).to_dict()


def replace_asin_with_title(similar_list):
    if isinstance(similar_list, list) and len(similar_list) > 1:
        return [asin_to_title.get(asin, asin) for asin in similar_list[1:]]
    return []


musics_df.loc[:, 'similar_titles'] = musics_df['similar'].apply(replace_asin_with_title)
musics_info = pd.DataFrame()
musics_info['title'] = musics_df['title']
musics_info['similar'] = musics_df['similar_titles'].apply(lambda x: ', '.join(x))
musics_info[['num_ratings', 'avg_rating']] = musics_df.apply(
    lambda row: extract_review_info(row['reviews']),
    axis=1, result_type='expand'
)

output_dir = '/Users/spencer/Desktop/CS-458-Fall-2023/Amazon Final Project/Excel'
if not os.path.exists(output_dir):
    os.makedirs(output_dir)
top_musics = musics_info.sort_values(by=['avg_rating', 'num_ratings'], ascending=[False,
print(top_musics)
top_musics.to_excel('/Users/spencer/Desktop/CS-458-Fall-2023/Amazon Final Project/Excel/
```

```
                                              title  \
532306          Looking For-Best of David Hasselhoff
61614                                   Josh Groban
428746                                 Now or Never
433590          Tim McGraw and the Dancehall Doctors
482364                                 No Name Face
482368                                 No Name Face
51436                                   Kind Of Blue
484602                                  Kind of Blue
253426                              Wish You Were Here
116914                                     Songbird
276452            Rent (1996 Original Broadway Cast)
70096    Who Is Jill Scott?: Words and Sounds, Vol. 1
272442    Who Is Jill Scott? Words and Sounds, Vol. 1
320542                                  Version 2.0
103013                        Live at Luther College
468171                                    The Bends
```

```
171285                                              Grace
269000                                             Weezer
243352                                           Illmatic
369                                   The Book of Secrets
371                                   The Book of Secrets
286697                                            Sublime
387898                                            Sublime
77284                                       Siamese Dream
357741                                      Siamese Dream
531440                              Siamese Dream [Clean]
324183                                       Achtung Baby
416123                                       Achtung Baby
197670                          MTV Unplugged in New York
197722                              Unplugged in New York
509417                                        Golden Road
531746                                        Golden Road
359161                                   Wide Open Spaces
303361                                 Little Earthquakes
399972                                      Back in Black
399974                                      Back in Black
474131                          Dónde Están los Ladrones?
163263                          Fumbling Towards Ecstasy
92916             Big Bad Voodoo Daddy [Big Bad Records]
263730                 Big Bad Voodoo Daddy [Interscope]
168516            All Things Must Pass [DIGI-PAK EDITION]
414888               All Things Must Pass [BOXED EDITION]
253708                              Blood Sugar Sex Magik
226878                                   This Time Around
471174                    Enter the Wu-Tang (36 Chambers)
270149                    Enter The Wu-Tang (36 Chambers)
173144                                           Aquemini
99854                                     40 Oz to Freedom
99857                                     40 Oz to Freedom
80400                                        Rust in Peace

                                                   similar  num_ratings  \
532306                                                             1084
61614   B0000CFW87, Josh Groban in Concert (with Bonus...          990
428746  B0000DYVRR, Into Your Head, Another Earthquake...          978
433590  Tim McGraw - Greatest Hits, B0002IQF7M, Set Th...          612
482364  Stanley Climbfall, B0007PALCU, Stanley Climbfa...          555
482368  Stanley Climbfall, B0007PALCU, Stanley Climbfa...          555
51436   Time Out, B0000A118M, Thelonious Monk with Joh...          541
484602  Time Out, B0000A118M, Thelonious Monk with Joh...          541
253426  Dark Side of the Moon 30th Anniversary Edition...          527
116914  Time After Time, Live at Blues Alley, Imagine,...          522
276452  B0000TB01Y, B0000BZK1R, Aida (2000 Original Br...          515
70096   B0002S94RK, Experience: Jill Scott, Floetic, B...          513
272442  B0002S94RK, Experience: Jill Scott, Floetic, B...          513
320542  Garbage, Beautifulgarbage, B0007Y8A06, Angelfi...          513
103013  Live at Red Rocks 8.15.95, B0000UJLMS, Under t...          494
468171  Ok Computer, Kid A, Pablo Honey, Amnesiac, Hai...          480
171285  Sketches (For My Sweetheart the Drunk) [CD-Ext...          477
269000  Weezer (Green Album), Pinkerton, Maladroit, Do...          445
243352  It Was Written, Reasonable Doubt, Stillmatic, ...          438
369     B0002VEX32, The Mask and Mirror, B0002VEX1O, B...          416
371     B0002VEX32, The Mask and Mirror, B0002VEX1O, B...          416
286697  40 Oz to Freedom, Second Hand Smoke, Robbin' t...          407
387898  40 Oz to Freedom, Second Hand Smoke, Robbin' t...          407
77284   Mellon Collie and the Infinite Sadness, Gish, ...          398
357741  Mellon Collie and the Infinite Sadness, Gish, ...          398
531440  Mellon Collie and the Infinite Sadness, Gish, ...          398
324183  The Joshua Tree, War, All That You Can't Leave...          395
416123  The Joshua Tree, War, All That You Can't Leave...          395
197670  Nevermind, In Utero, Bleach, Incesticide, From...          369
197722  Nevermind, In Utero, Bleach, Incesticide, From...          369
```

```
509417  B0002VEU62, keith urban, B0002VL0Z6, B0002IQF7...          360
531746  B0002VEU62, keith urban, B0002VL0Z6, B0002IQF7...          360
359161       Fly, Home, B0000DYJM4, Come on Over, Breathe          332
303361  Under the Pink, Boys for Pele, From the Choirg...          328
399972  Highway to Hell [Expanded], Highway to Hell, F...          325
399974  Highway to Hell [Expanded], Highway to Hell, F...          325
474131  Pies Descalzos, Laundry Service, Shakira: MTV ...          325
163263   Surfacing, B0000C6E4D, Solace, Mirrorball, Touch          319
92916   This Beautiful Life, B00009YRSB, Zoot Suit Rio...          311
263730  This Beautiful Life, B00009YRSB, Zoot Suit Rio...          311
168516  B0000E6I1J, B0000CEB4V, Brainwashed, B00014TJ7...          309
414888  B0000E6I1J, B0000CEB4V, Brainwashed, B00014TJ7...          309
253708  Californication, One Hot Minute, Mother's Milk...          296
226878  Middle of Nowhere, 3 Car Garage, B0002CHI9W, B...          282
471174  Wu-Tang Forever, Liquid Swords, Only Built 4 C...          276
270149  Wu-Tang Forever, Liquid Swords, Only Built 4 C...          275
173144  Atliens, Stankonia, Southernplayalisticadillac...          273
99854   Sublime, Second Hand Smoke, Robbin' the Hood, ...          264
99857   Sublime, Second Hand Smoke, Robbin' the Hood, ...          264
80400   Peace Sells...But Who's Buying?, Countdown to ...          259

        avg_rating
532306           5
61614            5
428746           5
433590           5
482364           5
482368           5
51436            5
484602           5
253426           5
116914           5
276452           5
70096            5
272442           5
320542           5
103013           5
468171           5
171285           5
269000           5
243352           5
369              5
371              5
286697           5
387898           5
77284            5
357741           5
531440           5
324183           5
416123           5
197670           5
197722           5
509417           5
531746           5
359161           5
303361           5
399972           5
399974           5
474131           5
163263           5
92916            5
263730           5
168516           5
414888           5
253708           5
226878           5
```

```
471174          5
270149          5
173144          5
99854           5
99857           5
80400           5
```

videosimilar.py

Explanation:

This runs code to look and see the top 50 video recommended and prints it into an excel file as well as onto the terminal. I will connect the excel files to my project so you can see what was derived from this code.

In [5]:
```python
#videosimilar.py

import pandas as pd
import re
import os

def extract_review_info(reviews_str):
    num_ratings_match = re.search(r'total: (\d+)', reviews_str)
    avg_rating_match = re.search(r'avg rating: (\d)', reviews_str)
    num_ratings = int(num_ratings_match.group(1)) if num_ratings_match else 0
    avg_rating = int(avg_rating_match.group(1)) if avg_rating_match else 0

    return num_ratings, avg_rating

df = pd.read_json('/Users/spencer/Desktop/CSV, JSON, Excel and txt of Amazon Final Proje
videos_df = df[df['group'] == 'Video'].copy()
asin_to_title = pd.Series(videos_df.title.values, index=videos_df.ASIN).to_dict()


def replace_asin_with_title(similar_list):
    if isinstance(similar_list, list) and len(similar_list) > 1:
        return [asin_to_title.get(asin, asin) for asin in similar_list[1:]]
    return []


videos_df.loc[:, 'similar_titles'] = videos_df['similar'].apply(replace_asin_with_title)
videos_info = pd.DataFrame()
videos_info['title'] = videos_df['title']
videos_info['similar'] = videos_df['similar_titles'].apply(lambda x: ', '.join(x))
videos_info[['num_ratings', 'avg_rating']] = videos_df.apply(
    lambda row: extract_review_info(row['reviews']),
    axis=1, result_type='expand'
)

output_dir = '/Users/spencer/Desktop/CS-458-Fall-2023/Amazon Final Project/Excel'
if not os.path.exists(output_dir):
    os.makedirs(output_dir)
top_videos = videos_info.sort_values(by=['avg_rating', 'num_ratings'], ascending=[False,
print(top_videos)
top_videos.to_excel('/Users/spencer/Desktop/CS-458-Fall-2023/Amazon Final Project/Excel/
```

```
                                                title  \
231128                    Ever After - A Cinderella Story
525483                    Ever After - A Cinderella Story
354451                                          Labyrinth
302316                                   Band of Brothers
468022    The Shawshank Redemption (Widescreen Edition)
485554                           The Shawshank Redemption
512294                           The Shawshank Redemption
```

```
146556                                          Newsies
91825                                       Toy Story 2
401263                                      Toy Story 2
19292                            Grave of the Fireflies
91286                            Grave of the Fireflies
454545                               *NSYNC: *N The Mix
104904              The Iron Giant (Widescreen Edition)
231438                                   The Iron Giant
54504                                             Glory
95198               Glory (Edited for Educational Uses)
372145                                          Scrooge
537482                                          Scrooge
266199                             Anne of Green Gables
509755                                      Rear Window
52943                               Pride and Prejudice
12441                                       October Sky
221356                                      October Sky
139387                                  The Temptations
124979                   The Good, the Bad and the Ugly
39548                                    Into the Woods
542179                                          Ed Wood
362932                       From the Earth to the Moon
339887                      Backstreet Boys: All Access
277696                               Pink Floyd - Pulse
315570  The Incredible Adventures of Wallace and Gromit
151290           M*A*S*H TV Season One - 3 Tape Box Set
165892                                Strictly Ballroom
338630                                 Sunset Boulevard
56676                                     All About Eve
161227                                    All About Eve
185882                                        Tommy Boy
363640                            Black Sheep/Tommy Boy
276205                                 The Pirate Movie
492830                     The Adventures of Robin Hood
282131                                 Midnight Madness
37545            Cosmos Boxed Set (Collector's Edition)
143165     9/11 - The Filmmakers' Commemorative Edition
27319                                       Being There
127846                              Love and Basketball
285326                Lagaan: Once upon a Time in India
305970     Raiders of the Lost Ark (Widescreen Edition)
459359                          Raiders of the Lost Ark
192129                          Raiders of the Lost Ark

                                                   similar  num_ratings  \
231128  B00006ZXSL, 6304765266, B00000K31Q, B00005LOKQ...          817
525483  B00006ZXSL, 6304765266, B00005LOKQ, B00000K31Q...          817
354451  B00000JPH6, B00005LKHZ, B000063UR2, B00003CXDD...          805
302316  B0001NBLVI, B00003CXCT, B000634DCW, 074322454X...          786
468022  B00003CXCT, B00003CWQU, B000634DCW, B00005JMQW...          758
485554  B00003CXCT, B00003CWQU, B000634DCW, B00005JMQW...          758
512294  B00003CXCT, B00003CWQU, B000634DCW, B00005JMQW...          758
146556  B000056QE6, B000065V3W, B00009MEJC, 0767851013...          524
91825   B000059XUT, B00007LVCM, B00001QEE7, B00005JKDR...          473
401263  B000059XUT, B00007LVCM, B00001QEE7, B00005JKDR...          473
19292   B00003CXBK, B0001XAPZ6, B00003CXCZ, B00000JL3V...          465
91286   B00003CXBK, B0001XAPZ6, B00003CXCZ, B00000JL3V...          465
454545  B00007ELD0, B00006L90M, B000054OWC, B00004YZGQ...          409
104904  B00005JN4W, B00020SK1Y, B00005JMQW, 0375801537...          351
231438  B00005JN4W, B00020SK1Y, B00005JMQW, 0375801537...          351
54504   B00003CXA6, 0783231202, B00009OOFA, B00002ND77...          294
95198   B00003CXA6, 0783231202, B00009OOFA, B00002ND77...          294
372145  B00000JT8Z, B00000K3CJ, 0780623746, B00001O2G7...          284
537482  B00000JT8Z, B00000K3CJ, 0780623746, B00001O2G7...          284
266199  B00005Y7AN, B00005Y7AM, 0553609416, B00005NGVV...          259
509755  0783226055, 0790749815, 0783225849, 0783240236...          246
```

```
52943   0800141660, B00003JRCQ, 014025157X, B00006JDVX...          241
12441   0385333218, 0783219695, 0440237165, 6305428352...          225
221356                                                             225
139387  B00005RYOQ, B00008J2HC, 6305428409, B00005YUOA...          221
124979  B0000AUHPG, B00000K0DM, 0792839056, B00005NTNW...          217
39548   B00005JL6V, 630530209X, B000002WAB, 0930452933...          187
542179  B00004Z4WX, B0002W4TNA, B00005JMJG, B0007CNXUK...          186
362932  0780022319, B000092T6N, 0783219695, B00004U2MS...          180
339887  B00004ZELJ, B00004ZEMK, B00005AAF2, B00005QCV8...          179
277696  6301334175, B00006LI4S, B0000DBJDM, B000002B35...          175
315570  B000051YMM, B00003CXJ4, B00004W3HB, B0000AYK1R...          169
151290  B000066STL, B000078UJW, B00008WJE5, B00008YGS0...          166
165892  B0002V7S34, 0767812387, 0783240279, B000063141...          164
338630  B00006RCO1, B000022TSH, B00003CX9E, B00003CXBU...          163
56676   B00003CXCW, B00007G1ZM, B00008LDNZ, B0000DJZ8R...          161
161227  B00003CXCW, B00007G1ZM, B00008LDNZ, B0000DJZ8R...          161
185882  B00005JL1T, 0780618556, B00000K3U4, 630529142X...          160
363640  B00005JL1T, 0780618556, B00000K3U4, 630529142X...          160
276205  B0006J28MS, B00000IQW7, B0006SSP9O, B0003JAONG...          154
492830  B00008LDO2, B00005JMR7, B0007OY2PS, B00005JMR6...          152
282131  B00009AOBK, B0001WTVGG, B0003JAONG, B00076ONW8...          147
37545   B0000ZG0TA, B00004VWUF, 0780631315, B00005BIFZ...          145
143165  B00006BS70, B0000635YC, B0000A02UG, B000067IZM...          145
27319   B00005JKH9, B000055Y0X, 6305882592, B0001AG01M...          140
127846  B000035Z28, 0783240201, B00005JLON, B00000JGHO...          140
285326  B00006AW0I, B00005RYLQ, B00007KQ9V, B00005NG3Q...          140
305970  Indiana Jones and the Last Crusade, Indiana Jo...          138
459359  Indiana Jones and the Last Crusade, Indiana Jo...          138
192129  Indiana Jones and the Last Crusade, Indiana Jo...          137

        avg_rating
231128           5
525483           5
354451           5
302316           5
468022           5
485554           5
512294           5
146556           5
91825            5
401263           5
19292            5
91286            5
454545           5
104904           5
231438           5
54504            5
95198            5
372145           5
537482           5
266199           5
509755           5
52943            5
12441            5
221356           5
139387           5
124979           5
39548            5
542179           5
362932           5
339887           5
277696           5
315570           5
151290           5
165892           5
338630           5
```

```
56676            5
161227           5
185882           5
363640           5
276205           5
492830           5
282131           5
37545            5
143165           5
27319            5
127846           5
285326           5
305970           5
459359           5
192129           5
```

total.py

Explanation:

This code just gives a graph that will output the total per group from the given amazon-meta.txt (then transformed into amazon-meta.json)

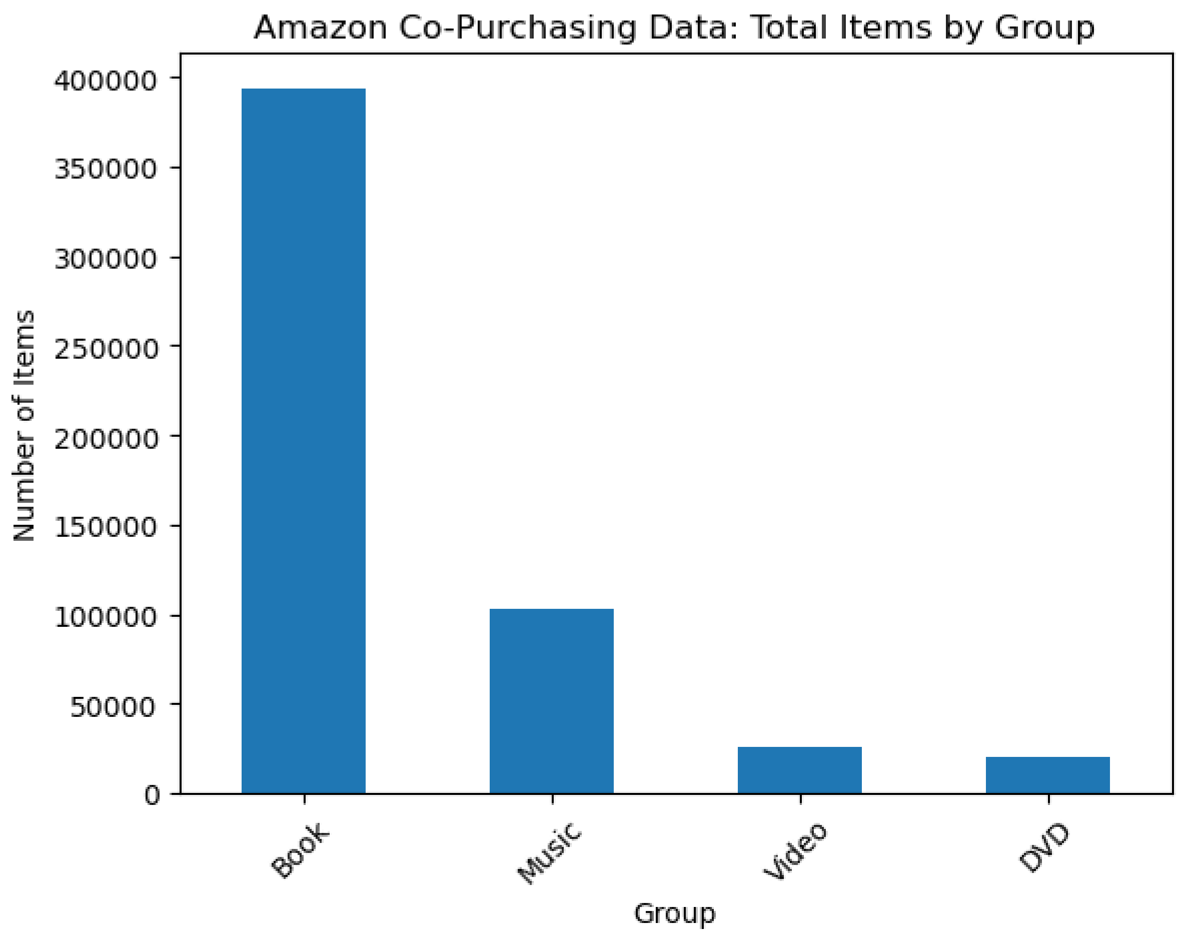
In [6]:

```
#total.py

import pandas as pd
import matplotlib.pyplot as plt


df = pd.read_json('/Users/spencer/Desktop/CSV, JSON, Excel and txt of Amazon Final Proje
included_groups = ['Book', 'Music', 'Video', 'DVD']
filtered_df = df[df['group'].isin(included_groups)]
group_counts = filtered_df['group'].value_counts()

# Plot the data
group_counts.plot(kind='bar')
plt.title('Amazon Co-Purchasing Data: Total Items by Group')
plt.xlabel('Group')
plt.ylabel('Number of Items')
plt.xticks(rotation=45)
plt.show()
```



predict_me_unified.py

Explanation:

This code is used in the run_it.py file that will provide a user interface that will both suggest the top 5 of a given group as well as predict using mean square based on each group.

In [7]:
```
#predict_me_unified.py

import pandas as pd
import os
import re
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import mean_squared_error
import numpy as np

def load_and_prepare_data(filepath, category):
    df = pd.read_json(filepath)
    df = df[df['group'] == category]
    df['num_ratings'], df['avg_rating'] = zip(*df['reviews'].map(extract_review_info))
    df['num_similar'] = df['similar'].apply(lambda x: len(x) - 1 if isinstance(x, list)
    return df

def extract_review_info(reviews_str):
    num_ratings_match = re.search(r'total: (\d+)', reviews_str)
    avg_rating_match = re.search(r'avg rating: (\d)', reviews_str)
    num_ratings = int(num_ratings_match.group(1)) if num_ratings_match else 0
    avg_rating = int(avg_rating_match.group(1)) if avg_rating_match else 0

    return num_ratings, avg_rating

def train_predict_model(df):
    X = df[['num_ratings', 'avg_rating', 'num_similar']]
    y = df['salesrank']
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_stat
    model = RandomForestRegressor(n_estimators=100, random_state=42)
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    rmse = np.sqrt(mean_squared_error(y_test, y_pred))
    print(f"RMSE: {rmse}")
    df['predicted_salesrank'] = model.predict(X)
    return df

def get_top_books(filepath):
    df = load_and_prepare_data(filepath, 'Book')
    top_books = df.sort_values(by=['avg_rating', 'num_ratings'], ascending=[False, False
    return top_books[['title', 'avg_rating', 'num_ratings']].to_dict(orient='records')


def predict_book_choice(filepath):
    df = load_and_prepare_data(filepath, 'Book')
    df = train_predict_model(df)
    return df[['title', 'predicted_salesrank']]

# Functions for DVDs
def get_top_dvds(filepath):
    df = load_and_prepare_data(filepath, 'DVD')
    top_dvds = df.sort_values(by=['avg_rating', 'num_ratings'], ascending=[False, False]
    return top_dvds[['title', 'avg_rating', 'num_ratings']].to_dict(orient='records')

def predict_dvd_choice(filepath):
    df = load_and_prepare_data(filepath, 'DVD')
    df = train_predict_model(df)
    return df[['title', 'predicted_salesrank']]

# Functions for Music
def get_top_musics(filepath):
    df = load_and_prepare_data(filepath, 'Music')
    top_musics = df.sort_values(by=['avg_rating', 'num_ratings'], ascending=[False, Fals
    return top_musics[['title', 'avg_rating', 'num_ratings']].to_dict(orient='records')

def predict_music_choice(filepath):
    df = load_and_prepare_data(filepath, 'Music')
    df = train_predict_model(df)
    return df[['title', 'predicted_salesrank']]

# Functions for Videos
def get_top_videos(filepath):
    df = load_and_prepare_data(filepath, 'Video')
    top_videos = df.sort_values(by=['avg_rating', 'num_ratings'], ascending=[False, Fals
    return top_videos[['title', 'avg_rating', 'num_ratings']].to_dict(orient='records')

def predict_video_choice(filepath):
    df = load_and_prepare_data(filepath, 'Video')
    df = train_predict_model(df)
    return df[['title', 'predicted_salesrank']]
```

run_it.py

Explanation:

This is the application that will provide a User Interface that will suggest the top 5 of any selected group as well as predict the top 5 suggestions using mean square. The program does take some time based on your computer setup, but it should bring you an answer fairly quickly.

```
In [ ]: #run_it.py

        import sys
        from PyQt5.QtWidgets import QApplication, QMainWindow, QPushButton, QVBoxLayout, QWidget
        from predict_me_unified import get_top_books, predict_book_choice, get_top_dvds, predict

        JSON_FILE_PATH = '/Users/spencer/Desktop/CSV, JSON, Excel and txt of Amazon Final Projec


        class PredictMeApp(QMainWindow):
            def __init__(self):
                super().__init__()

                # Main Window Properties
                self.setWindowTitle('Predict Me: Recommendations')
                self.setGeometry(100, 100, 800, 600)

                # Central Widget and Layout
                central_widget = QWidget()
                self.setCentralWidget(central_widget)
                layout = QVBoxLayout(central_widget)

                # Category Selection
                self.category_combo = QComboBox()
                self.category_combo.addItems(['Books', 'DVDs', 'Music', 'Videos'])
                layout.addWidget(self.category_combo)

                # Suggest Button
                self.suggest_button = QPushButton('Suggest Top 5')
                self.suggest_button.clicked.connect(self.suggest_top_five)
                layout.addWidget(self.suggest_button)

                # Predict Button
                self.predict_button = QPushButton('Predict My Choice')
                self.predict_button.clicked.connect(self.predict_choice)
                layout.addWidget(self.predict_button)


                self.scroll_area = QScrollArea()
                self.results_label = QLabel('Results will be displayed here')
                self.results_label.setWordWrap(True)  # Enable word wrap
                self.scroll_area.setWidget(self.results_label)
                self.scroll_area.setWidgetResizable(True)
                layout.addWidget(self.scroll_area)
                self.category_combo.setMaximumWidth(200)
                self.suggest_button.setMaximumWidth(200)
                self.predict_button.setMaximumWidth(200)


            def suggest_top_five(self):
                category = self.category_combo.currentText()
                if category == 'Books':
                    results = get_top_books(JSON_FILE_PATH)
                elif category == 'DVDs':
                    results = get_top_dvds(JSON_FILE_PATH)
                elif category == 'Music':
                    results = get_top_musics(JSON_FILE_PATH)
                elif category == 'Videos':
                    results = get_top_videos(JSON_FILE_PATH)

                # Formatting the results for display
                formatted_results = f"Top 5 Suggested {category} Based on Avg Rating and Num Rat
                for item in results:
                    title = item.get('title', 'N/A')
                    avg_rating = item.get('avg_rating', 'N/A')
                    num_ratings = item.get('num_ratings', 'N/A')
                    formatted_results += f"Title: {title}, Avg Rating: {avg_rating}, Num Ratings

                self.results_label.setText(formatted_results)



            def predict_choice(self):
                category = self.category_combo.currentText()
                if category == 'Books':
                    prediction = predict_book_choice(JSON_FILE_PATH)
                elif category == 'DVDs':
                    prediction = predict_dvd_choice(JSON_FILE_PATH)
                elif category == 'Music':
                    prediction = predict_music_choice(JSON_FILE_PATH)
                elif category == 'Videos':
                    prediction = predict_video_choice(JSON_FILE_PATH)

                # Sort by predicted_salesrank and get the top 5
                top_predictions = prediction.sort_values(by='predicted_salesrank').head(5)

                # Formatting the prediction for display
                formatted_prediction = "\n".join([f"{item['title']}: {item['predicted_salesrank'
                self.results_label.setText(f"Top 5 Predicted {category} Choices:\n{formatted_pre


        if __name__ == '__main__':
            app = QApplication(sys.argv)
            main_window = PredictMeApp()
            main_window.show()
            sys.exit(app.exec_())
```

In [ ]:

```
```