



Maryam osama
210461
multimedia

Cloud assignment

Book csv

The screenshot shows a JupyterLab environment with a file browser on the left, a code editor in the center, and a table view of data at the bottom. The file browser shows a directory structure with files like 'books.csv' and 'cc2.ipynb'. The code editor contains Python code for importing libraries and reading a CSV file. The table view displays the loaded data with columns for book_id, goodreads_book_id, best_book_id, work_id, books_count, isbn, isbn13, authors, original_publication_year, original_title, and ratings_count.

```
[1]: import pandas as pd
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt
from random import sample
import seaborn as sns
from sklearn import preprocessing
import warnings
from sklearn.preprocessing import MinMaxScaler
warnings.filterwarnings('ignore')
```

```
[6]: df = pd.read_csv("books.csv")
df
```

	book_id	goodreads_book_id	best_book_id	work_id	books_count	isbn	isbn13	authors	original_publication_year	original_title	...	ratings_count
0	1	2767052	2767052	2792775	272	439023483	9.780439e+12	Suzanne Collins	2008.0	The Hunger Games	...	478k
1	2	3	3	4640799	491	439554934	9.780440e+12	J.K. Rowling, Mary GrandPré	1997.0	Harry Potter and the Philosopher's Stone	...	460k
2	3	41865	41865	3212258	226	316015849	9.780316e+12	Stephenie Meyer	2005.0	Twilight	...	386k
3	6	11870085	11870085	16827462	226	525478817	9.780525e+12	John Green	2012.0	The Fault in Our Stars	...	234k
4	12	13335037	13335037	13155899	210	62024035	9.780062e+12	Veronica Roth	2011.0	Divergent	...	400k
...
1349	9925	86737	86737	3877968	52	1582349177	9.781582e+12	Mary Hoffman

Would you like to receive official Jupyter news? Please read the privacy policy. [Open privacy policy](#) Yes No

Cleaning data and display basic statistics

The screenshot shows a Jupyter Notebook environment with a file explorer on the left and a code editor on the right. The file explorer displays a directory with files: 'Untitled Fol...', 'work', 'books.csv', 'cc2.ipynb', and 'Untitled.ipynb'. The code editor shows three cells of Python code. The first cell removes irrelevant attributes and handles missing values. The second cell removes duplicate data and prints the result. The third cell displays basic statistical data using the describe() method.

```
[7]: #removing irrelevant attributes and handling missing values
df.drop(['small_image_url', 'image_url', 'isbn13'], axis='columns', inplace=True)
df = df.dropna()

[8]: #remove duplicate data
df.drop_duplicates(inplace=True)
print(df.duplicated())

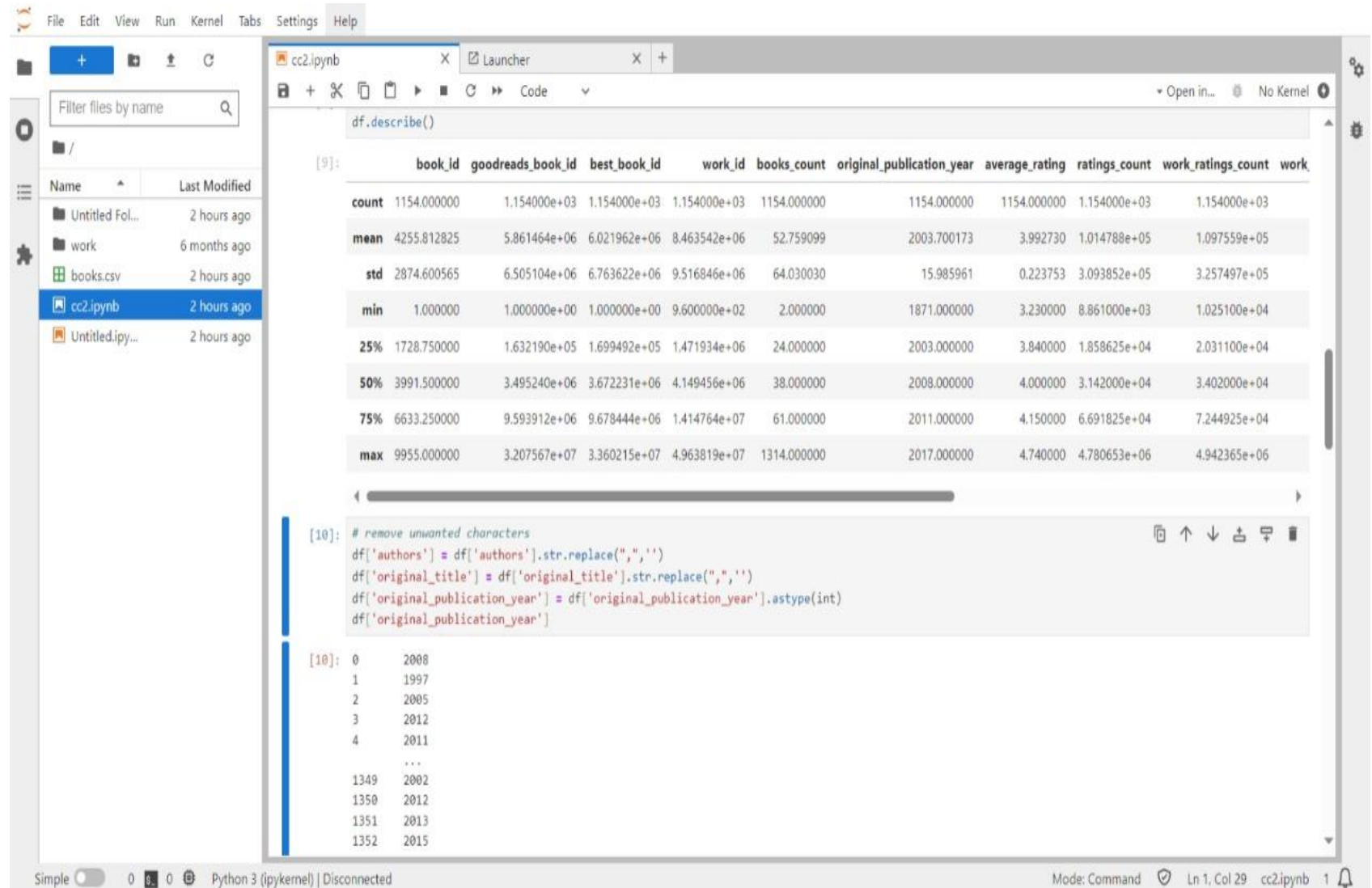
0      False
1      False
2      False
3      False
4      False
...
1349   False
1350   False
1351   False
1352   False
1353   False
Length: 1154, dtype: bool

[9]: #basic statistical data
df.describe()
```

	book_id	goodreads_book_id	best_book_id	work_id	books_count	original_publication_year	average_rating	ratings_count	work_ratings_count	work
count	1154.000000	1.154000e+03	1.154000e+03	1.154000e+03	1154.000000	1154.000000	1154.000000	1.154000e+03	1.154000e+03	
mean	4255.812825	5.861464e+06	6.021962e+06	8.463542e+06	52.759099	2003.700173	3.992730	1.014788e+05	1.097559e+05	
std	2874.600565	6.505104e+06	6.763622e+06	9.516846e+06	64.030030	15.985961	0.223753	3.093852e+05	3.257497e+05	
min	1.000000	1.000000e+00	1.000000e+00	9.600000e+02	2.000000	1871.000000	3.230000	8.861000e+03	1.025100e+04	
25%	1728.750000	1.632190e+05	1.699492e+05	1.471934e+06	24.000000	2003.000000	3.840000	1.858625e+04	2.031100e+04	
50%	3991.500000	3.495240e+06	3.672231e+06	4.149456e+06	38.000000	2008.000000	4.000000	3.142000e+04	3.402000e+04	

Simple 0 0 Python 3 (ipykernel) | Disconnected Mode: Command Ln 1, Col 1 cc2.ipynb

Displaying the
rest statistics
& removing
unwanted
character



The screenshot shows a Jupyter Notebook interface with a file explorer on the left and a code editor on the right. The file explorer shows a directory structure with files like 'books.csv', 'cc2.ipynb', and 'Untitled.ipynb'. The code editor displays the output of `df.describe()` and a code cell for removing unwanted characters from the 'authors' and 'original_title' columns.

df.describe() Output:

	book_id	goodreads_book_id	best_book_id	work_id	books_count	original_publication_year	average_rating	ratings_count	work_ratings_count	work
count	1154.000000	1.154000e+03	1.154000e+03	1.154000e+03	1154.000000	1154.000000	1154.000000	1.154000e+03	1.154000e+03	
mean	4255.812825	5.861464e+06	6.021962e+06	8.463542e+06	52.759099	2003.700173	3.992730	1.014788e+05	1.097559e+05	
std	2874.600565	6.505104e+06	6.763622e+06	9.516846e+06	64.030030	15.985961	0.223753	3.093852e+05	3.257497e+05	
min	1.000000	1.000000e+00	1.000000e+00	9.600000e+02	2.000000	1871.000000	3.230000	8.861000e+03	1.025100e+04	
25%	1728.750000	1.632190e+05	1.699492e+05	1.471934e+06	24.000000	2003.000000	3.840000	1.858625e+04	2.031100e+04	
50%	3991.500000	3.495240e+06	3.672231e+06	4.149456e+06	38.000000	2008.000000	4.000000	3.142000e+04	3.402000e+04	
75%	6633.250000	9.593912e+06	9.678444e+06	1.414764e+07	61.000000	2011.000000	4.150000	6.691825e+04	7.244925e+04	
max	9955.000000	3.207567e+07	3.360215e+07	4.963819e+07	1314.000000	2017.000000	4.740000	4.780653e+06	4.942365e+06	

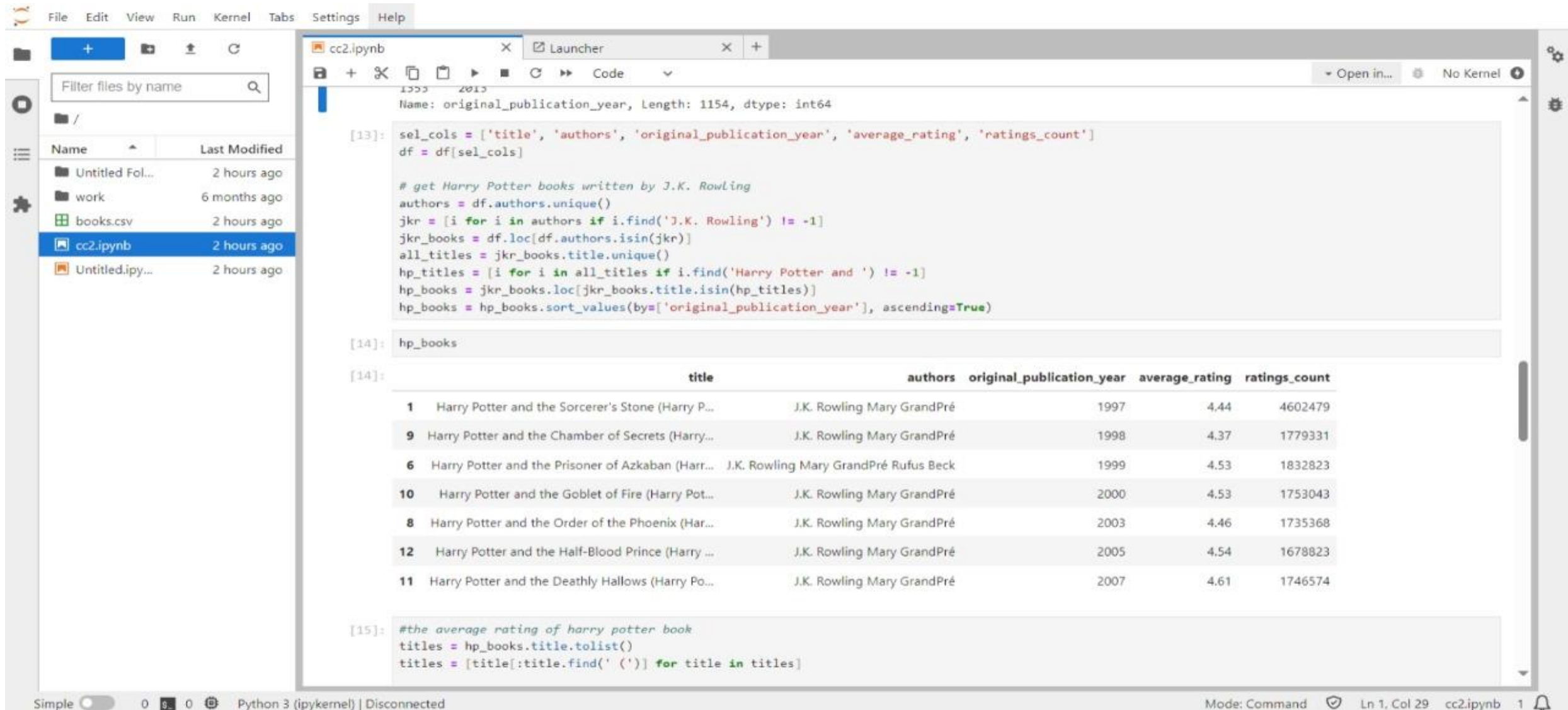
Code Cell [10]:

```
# remove unwanted characters
df['authors'] = df['authors'].str.replace(",","")
df['original_title'] = df['original_title'].str.replace(",","")
df['original_publication_year'] = df['original_publication_year'].astype(int)
df['original_publication_year']
```

Output [10]:

```
0    2008
1    1997
2    2005
3    2012
4    2011
...
1349  2002
1350  2012
1351  2013
1352  2015
```


Focusing on harry potter books for data analysis



The screenshot shows a Jupyter Notebook interface with a file explorer on the left and a code editor on the right. The file explorer shows a directory structure with files like 'books.csv' and 'cc2.ipynb'. The code editor displays the following code:

```
[13]: sel_cols = ['title', 'authors', 'original_publication_year', 'average_rating', 'ratings_count']
df = df[sel_cols]

# get Harry Potter books written by J.K. Rowling
authors = df.authors.unique()
jkr = [i for i in authors if i.find('J.K. Rowling') != -1]
jkr_books = df.loc[df.authors.isin(jkr)]
all_titles = jkr_books.title.unique()
hp_titles = [i for i in all_titles if i.find('Harry Potter and ') != -1]
hp_books = jkr_books.loc[jkr_books.title.isin(hp_titles)]
hp_books = hp_books.sort_values(by=['original_publication_year'], ascending=True)

[14]: hp_books

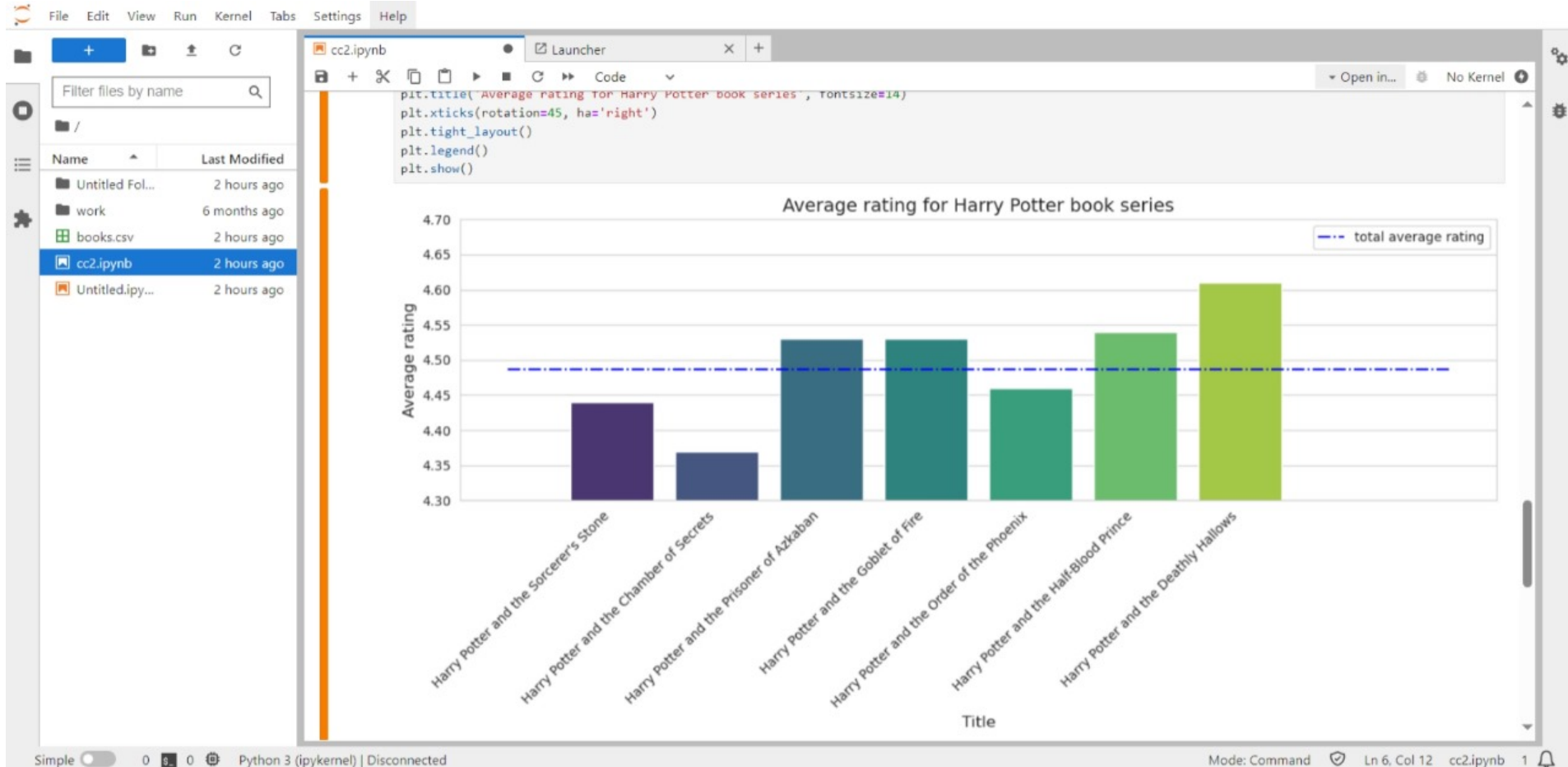
[14]:
```

	title	authors	original_publication_year	average_rating	ratings_count
1	Harry Potter and the Sorcerer's Stone (Harry P...	J.K. Rowling Mary GrandPré	1997	4.44	4602479
9	Harry Potter and the Chamber of Secrets (Harry...	J.K. Rowling Mary GrandPré	1998	4.37	1779331
6	Harry Potter and the Prisoner of Azkaban (Harr...	J.K. Rowling Mary GrandPré Rufus Beck	1999	4.53	1832823
10	Harry Potter and the Goblet of Fire (Harry Pot...	J.K. Rowling Mary GrandPré	2000	4.53	1753043
8	Harry Potter and the Order of the Phoenix (Har...	J.K. Rowling Mary GrandPré	2003	4.46	1735368
12	Harry Potter and the Half-Blood Prince (Harry ...	J.K. Rowling Mary GrandPré	2005	4.54	1678823
11	Harry Potter and the Deathly Hallows (Harry Po...	J.K. Rowling Mary GrandPré	2007	4.61	1746574

```
[15]: #the average rating of harry potter book
titles = hp_books.title.tolist()
titles = [title[:title.find(' ')] for title in titles]
```

The bottom status bar indicates the notebook is running on Python 3 (ipykernel) and is disconnected. The mode is set to Command, and the current position is Line 1, Column 29 in cc2.ipynb.

The result&plot of the average rate of harry potter books



The highest selling books & average rate of harry potter books results

The screenshot displays a Jupyter Notebook environment with a file explorer on the left and a code editor on the right. The file explorer shows a directory with files: 'Untitled Fol...', 'work', 'books.csv', 'cc2.ipynb' (selected), and 'Untitled.ipy...'. The code editor shows the following code and output:

```
[16]: #the average rating of harry potter books
      weighted_avg

[16]: 4.486117244334695

[17]: # the highest selling books
      # Get the book with the highest ratings count
      highest_rated = hp_books.sort_values('ratings_count', ascending=False).iloc[0]

      # Print the title of the highest rated book
      print(highest_rated['title'])

      Harry Potter and the Sorcerer's Stone (Harry Potter, #1)
```

The output of the code shows the average rating of Harry Potter books as 4.486117244334695 and the highest rated book as 'Harry Potter and the Sorcerer's Stone (Harry Potter, #1)'. The notebook interface includes a menu bar (File, Edit, View, Run, Kernel, Tabs, Settings, Help) and a status bar at the bottom indicating 'Python 3 (ipykernel) | Disconnected' and 'Mode: Edit'.