Date: 01-01-2024

EXPERIMENT NO.: 2

AIM: To study data wrangling in Python by performing required operations on given .csv files

SOFTWARE USED: Jupyter Notebook

THEORY:

Data Wrangling is the cleaning and transforming of one type of data to another type to make it more appropriate into a processed format. Data wrangling involves processing the data in various formats and analyses and get them to be used with another set of data and bringing them together into valuable insights. It further includes data aggregation, data visualization, and training statistical models for prediction. Data wrangling is one of the most important steps of the data science process.

Data wrangling involves several steps to clean, transform, and prepare raw data for analysis:

1) Data discovery: Describes how to understand your data. This is the first step to familiarize yourself with your data.
2) Structuring: The next step is to organize the data. Raw data is typically unorganized and much of it may not be useful for the end product. This step is important for easier computation and analysis in the later steps.
3) Cleaning: There are many different forms of cleaning data, for example one form of cleaning data is catching dates formatted in a different way and another form is removing outliers that will skew results and also formatting null values. This step is important in assuring the overall quality of the data.
4) Enriching: At this step determine whether or not additional data would benefit the data set that could be easily added.
5) Validating: to assure data consistency as well as quality and security
6) Publishing: This is the final step which involves preparing the data for further analysis.

pandas is a popular open-source Python library used for data manipulation and analysis.The primary data structures in Pandas are:

- Series: A one-dimensional labeled array capable of holding any data type (e.g., integers, strings, floating-point numbers, Python objects, etc.). It's similar to a Python list or dictionary but with additional functionalities.

- DataFrame: A two-dimensional labeled data structure with columns of potentially different types. It's similar to a spreadsheet or SQL table, where data is organized in rows and columns.

The following are some common data wrangling tasks performed with Pandas:

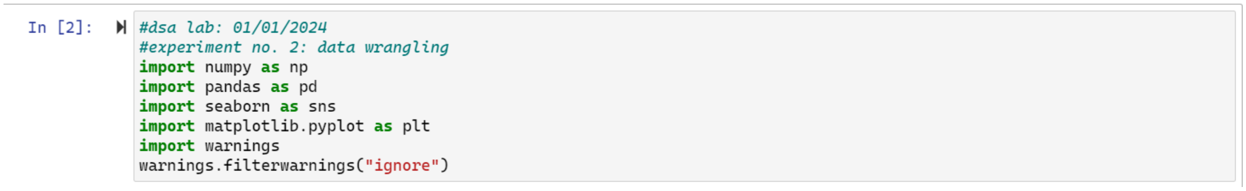
- Data Loading: Pandas provides functions to read data from various file formats such as CSV, Excel, JSON, SQL databases, and more. The read_csv(), read_excel(), read_json(), and read_sql() functions are commonly used for this purpose.

- Data Filtering and Selection: Pandas allows selecting specific rows and columns based on certain criteria using boolean indexing, label-based indexing (loc), or integer-based indexing (iloc).

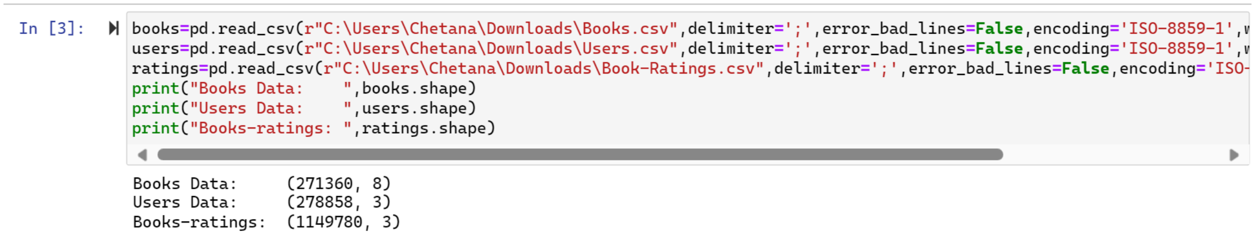
- Data Transformation: This includes tasks like reshaping data, merging datasets, pivoting, and grouping/aggregating data. Pandas offers functions like merge(), concat(), pivot_table(), and groupby() to perform these operations efficiently.

- Data Cleaning: This involves handling missing values, removing duplicates, correcting erroneous data, and converting data types. Pandas provides methods like dropna(), fillna(), drop_duplicates(), and various string manipulation functions for data cleaning.

OUTPUT:



```
In [2]:  #dsa lab: 01/01/2024
         #experiment no. 2: data wrangling
         import numpy as np
         import pandas as pd
         import seaborn as sns
         import matplotlib.pyplot as plt
         import warnings
         warnings.filterwarnings("ignore")
```



```
In [3]:  books=pd.read_csv(r"C:\Users\Chetana\Downloads\Books.csv",delimiter=';',error_bad_lines=False,encoding='ISO-8859-1',w
         users=pd.read_csv(r"C:\Users\Chetana\Downloads\Users.csv",delimiter=';',error_bad_lines=False,encoding='ISO-8859-1',w
         ratings=pd.read_csv(r"C:\Users\Chetana\Downloads\Book-Ratings.csv",delimiter=';',error_bad_lines=False,encoding='ISO-
         print("Books Data:     ",books.shape)
         print("Users Data:     ",users.shape)
         print("Books-ratings:  ",ratings.shape)

         Books Data:      (271360, 8)
         Users Data:      (278858, 3)
         Books-ratings:  (1149780, 3)
```

In [4]:
```
print("Columns: ",list(books.columns))
books.head()
```

```
Columns:  ['ISBN', 'Book-Title', 'Book-Author', 'Year-Of-Publication', 'Publisher', 'Image-URL-S', 'Image-URL-M',
'Image-URL-L']
```

Out[4]:

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher	Image-URL-S	Image-U
0	0195153448	Classical Mythology	Mark P. O. Morford	2002	Oxford University Press	http://images.amazon.com/images/P/0195153448.0...	http://images.amazon.com/images/P/01951534
1	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada	http://images.amazon.com/images/P/0002005018.0...	http://images.amazon.com/images/P/00020050
2	0060973129	Decision in Normandy	Carlo D'Este	1991	HarperPerennial	http://images.amazon.com/images/P/0060973129.0...	http://images.amazon.com/images/P/00609731
3	0374157065	Flu: The Story of the Great Influenza Pandemic...	Gina Bari Kolata	1999	Farrar Straus Giroux	http://images.amazon.com/images/P/0374157065.0...	http://images.amazon.com/images/P/03741570
4	0393045218	The Mummies of Urumchi	E. J. W. Barber	1999	W. W. Norton &amp; Company	http://images.amazon.com/images/P/0393045218.0...	http://images.amazon.com/images/P/03930452

In [5]:
```
books.drop(['Image-URL-S', 'Image-URL-M', 'Image-URL-L'], axis=1, inplace=True)
books.head(5)
```

Out[5]:

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher
0	0195153448	Classical Mythology	Mark P. O. Morford	2002	Oxford University Press
1	0002005018	Clara Callan	Richard Bruce Wright	2001	HarperFlamingo Canada
2	0060973129	Decision in Normandy	Carlo D'Este	1991	HarperPerennial
3	0374157065	Flu: The Story of the Great Influenza Pandemic...	Gina Bari Kolata	1999	Farrar Straus Giroux
4	0393045218	The Mummies of Urumchi	E. J. W. Barber	1999	W. W. Norton &amp; Company

In [6]:
```
books.isnull().sum()
```

Out[6]:
```
ISBN                   0
Book-Title             0
Book-Author            1
Year-Of-Publication    0
Publisher              2
dtype: int64
```

In [7]:
```
books.loc[books['Book-Author'].isnull(),:]
```

Out[7]:

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher
187689	9627982032	The Credit Suisse Guide to Managing Your Perso...	NaN	1995	Edinburgh Financial Publishing

In [8]:
```
books.loc[books['Publisher'].isnull(),:]
```

Out[8]:

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher
128890	193169656X	Tyrant Moon	Elaine Corvidae	2002	NaN
129037	1931696993	Finders Keepers	Linnea Sinclair	2001	NaN

In [9]:
```
books.at[187689,'Book-Author']='Other'
books.at[128890,'Publisher']='Other'
books.at[129037,'Publisher']='Other'
```

In [10]:

```
books['Year-Of-Publication'].unique()
```

Out[10]:

```
array([2002, 2001, 1991, 1999, 2000, 1993, 1996, 1988, 2004, 1998, 1994,
       2003, 1997, 1983, 1979, 1995, 1982, 1985, 1992, 1986, 1978, 1980,
       1952, 1987, 1990, 1981, 1989, 1984, 0, 1968, 1961, 1958, 1974,
       1976, 1971, 1977, 1975, 1965, 1941, 1970, 1962, 1973, 1972, 1960,
       1966, 1920, 1956, 1959, 1953, 1951, 1942, 1963, 1964, 1969, 1954,
       1950, 1967, 2005, 1957, 1940, 1937, 1955, 1946, 1936, 1930, 2011,
       1925, 1948, 1943, 1947, 1945, 1923, 2020, 1939, 1926, 1938, 2030,
       1911, 1904, 1949, 1932, 1928, 1929, 1927, 1931, 1914, 2050, 1934,
       1910, 1933, 1902, 1924, 1921, 1900, 2038, 2026, 1944, 1917, 1901,
       2010, 1908, 1906, 1935, 1806, 2021, '2000', '1995', '1999', '2004',
       '2003', '1990', '1994', '1986', '1989', '2002', '1981', '1993',
       '1983', '1982', '1976', '1991', '1977', '1998', '1992', '1996',
       '0', '1997', '2001', '1974', '1968', '1987', '1984', '1988',
       '1963', '1956', '1970', '1985', '1978', '1973', '1980', '1979',
       '1975', '1969', '1961', '1965', '1939', '1958', '1950', '1953',
       '1966', '1971', '1959', '1972', '1955', '1957', '1945', '1960',
       '1967', '1932', '1924', '1964', '2012', '1911', '1927', '1948',
       '1962', '2006', '1952', '1940', '1951', '1931', '1954', '2005',
       '1930', '1941', '1944', 'DK Publishing Inc', '1943', '1938',
       '1900', '1942', '1923', '1920', '1933', 'Gallimard', '1909',
       '1946', '2008', '1378', '2030', '1936', '1947', '2011', '2020',
       '1919', '1949', '1922', '1897', '2024', '1376', '1926', '2037'],
      dtype=object)
```

In [12]:

```
pd.set_option('display.max_colwidth', -1)
```

In [13]:

```
books.loc[books['Year-Of-Publication']=='DK Publishing Inc',:]
```

Out[13]:

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher
209538	078946697X	DK Readers: Creating the X-Men, How It All Began (Level 4: Proficient Readers)\";Michael Teitelbaum"	2000	DK Publishing Inc	http://images.amazon.com/images/P/078946697X.01.THUMBZZZ.jpg
221678	0789466953	DK Readers: Creating the X-Men, How Comic Books Come to Life (Level 4: Proficient Readers)\";James Buckley"	2000	DK Publishing Inc	http://images.amazon.com/images/P/0789466953.01.THUMBZZZ.jpg

In [14]:

```
books.loc[books['Year-Of-Publication']=='Gallimard',:]
```

Out[14]:

	ISBN	Book-Title	Book-Author	Year-Of-Publication	Publisher
220731	2070426769	Peuple du ciel, suivi de 'Les Bergers\";Jean-Marie Gustave Le ClÃ?Â©zio"	2003	Gallimard	http://images.amazon.com/images/P/2070426769.01.THUMBZZZ.jpg

In [22]:

```
books.at[209538,'Publisher']='DK Publishing Inc'
books.at[209538,'Year-Of-Publication']=2000
books.at[209538,'Book-Title']='DK Readers: Creating the X-Men, How it All Began (level 4: Proficient Readers)'
books.at[209538,'Book-Author']='Michael Teitelbaum'

books.at[221678, 'Publisher']='DK Publishing Inc'
books.at[221678,'Year-Of-Publication']=2000
books.at[221678,'Book-Title']='DK Readers: Creating the X-Men, How Comic Books came to Life (level4)'
books.at[221678,'Book-Author']='James Buckley'

books.at[220731, 'Publisher']='Gallimard'
books.at[220731,'Year-Of-Publication']=2003
books.at[220731,'Book-Title']='Peuple du ciel, suivi de' 'Les Bergers'
books.at[220731,'Book-Author']='JJean-Marie Gustave Le ClÃ?Â©zio'
```

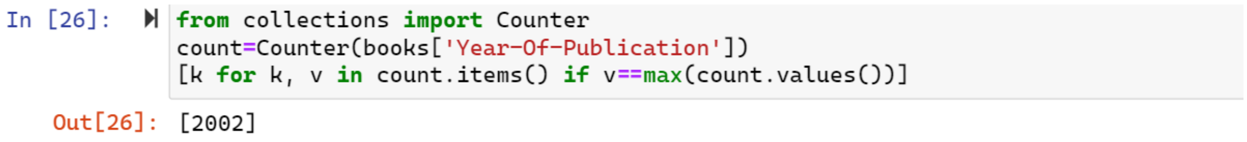
In [23]:

```
books['Year-Of-Publication']=books['Year-Of-Publication'].astype(int)
```

In [24]:

```
print(sorted(list(books['Year-Of-Publication'].unique())))
```

```
[0, 1376, 1378, 1806, 1897, 1900, 1901, 1902, 1904, 1906, 1908, 1909, 1910, 1911, 1914, 1917, 1919, 1920, 1921, 192
2, 1923, 1924, 1925, 1926, 1927, 1928, 1929, 1930, 1931, 1932, 1933, 1934, 1935, 1936, 1937, 1938, 1939, 1940, 194
1, 1942, 1943, 1944, 1945, 1946, 1947, 1948, 1949, 1950, 1951, 1952, 1953, 1954, 1955, 1956, 1957, 1958, 1959, 196
0, 1961, 1962, 1963, 1964, 1965, 1966, 1967, 1968, 1969, 1970, 1971, 1972, 1973, 1974, 1975, 1976, 1977, 1978, 197
9, 1980, 1981, 1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997, 199
8, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2008, 2010, 2011, 2012, 2020, 2021, 2024, 2026, 2030, 2037, 203
8, 2050]
```



```
In [26]:  from collections import Counter
          count=Counter(books['Year-Of-Publication'])
          [k for k, v in count.items() if v==max(count.values())]

Out[26]:  [2002]
```

CONCLUSION:

Thus, in the given experiment, data wrangling streamlines the process of preparing raw data on books and their related information, enhancing the quality and usability of given data. By handling missing values, cleaning inconsistencies, and structuring data appropriately, data wrangling ensures accurate analysis. It creates a transparent and efficient system for data management which enables for important business decisions.