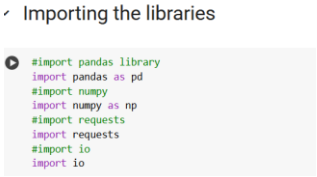
Data Collection and Preprocessing Phase

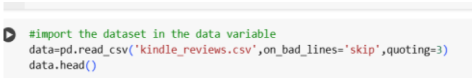
Date	June 2025
Team ID	Team-739774
Project Title	Amazon Kindle Store Review Analysis
Maximum Marks	6 Marks

Data Exploration and Preprocessing Template

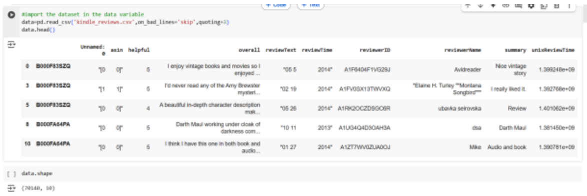
Identifies data sources, assesses quality issues like missing values and duplicates, and implements resolution plans to ensure accurate and reliable analysis.

Section	**Description**
Data Overview	Data overview enables a thorough analysis of Kindle store reviews based on customer ratings and review texts, helping to identify reader preferences, sentiment trends, and product feedback.
Data Preparation	Gather attributes such as review-id, product-id, review-text, star-rating, and verified-purchase. Handle missing values, outliers, and inconsistencies. Clean the text data by removing special characters, HTML tags, and stop words. Encode categorical fields if necessary, and prepare for ML models.
Handling missing values	After loading the dataset, check for null values in each column. If found, take steps such as imputing missing text with placeholder tokens or removing rows if critical fields like review-text or star-rating are missing. Use fill (), median/mode imputation, and visualize with heatmaps for missing data.
Data Visualisation	Visualizing data helps detect trends and patterns in reviews. For Kindle reviews, this includes bar plots of rating distribution, word clouds for frequent terms, and sentiment heatmaps. Use libraries like Matplotlib, Seaborn, and Word Cloud to generate visual insights from the data.
Splitting The Dataset Into Dependent And Independent Variable	For sentiment modeling, split the dataset into independent variables (X) like review-text, and dependent variable (y) such as star-rating or a sentiment label. This prepares the data for supervised learning tasks such as sentiment classification or rating prediction.
Data Preprocessing Code Screenshots	

Import the libraries	 Importing the libraries
	`#import pandas library`
	`import pandas as pd`
	`#import numpy`
	`import numpy as np`
	`#import requests`
	`import requests`
	`#import io`
	`import io`

Importing The Dataset	 `#import the dataset in the data variable`
	`data=pd.read_csv('kindle_reviews.csv',on_bad_lines='skip',quoting=3)`
	`data.head()`

Analyse The Data



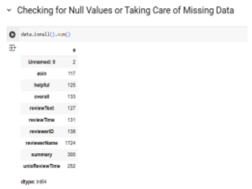
```
#import the dataset in the data variable
data=pd.read_csv('kindle_reviews.csv',on_bad_lines='skip',quoting=3)
data.head()
```

```
    Unnamed:  asin  helpful                                 overall  reviewText  reviewTime  reviewerID        reviewerName                          summary          unixReviewTime
        0
0   B000F83SZQ  "[0   0]"   5   I enjoy vintage books and movies so I enjoyed ...   "05 5   2014"  A1F6404F1VG29J   Avidreader                          Nice vintage story   1.399248e+09
3   B000F83SZQ  "[1   1]"   5   I'd never read any of the Amy Brewster mysteri...   "02 19  2014"  A1FV0SX13TWVXQ   "Elaine H. Turley ""Montana Songbird"""  I really liked it.  1.392768e+09
5   B000F83SZQ  "[0   0]"   4   A beautiful in-depth character description mak...   "05 26  2014"  A1RK2OCZDSGC6R   ubavka seirovska                    Review              1.401062e+09
8   B000FA64PA  "[0   0]"   5   Darth Maul working under cloak of darkness com...   "10 11  2013"  A1UG4Q4D3OAH3A   dsa                                 Darth Maul          1.381450e+09
10  B000FA64PA  "[0   0]"   5   I think I have this one in both book and audio...   "01 27  2014"  A1ZT7WV0ZUA0OJ   Mike                                Audio and book      1.390781e+09
```

```
data.shape
```

```
(70140, 10)
```

Checking for Null Values or Taking Care of Missing Data

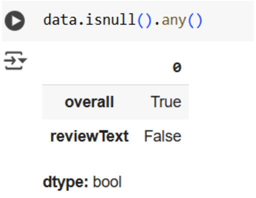


Checking for Null Values or Taking Care of Missing Data

```
data.isnull().sum()
```

```
                 0
Unnamed: 0       2
asin           117
helpful        125
overall        133
reviewText     127
reviewTime     131
reviewerID     138
reviewerName  1724
summary        300
unixReviewTime 252

dtype: int64
```

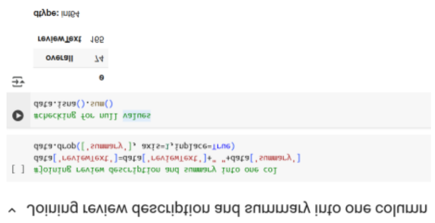


```
data.isnull().any()
```

```
                0
overall      True
reviewText  False

dtype: bool
```

Joining review description and summary into one column



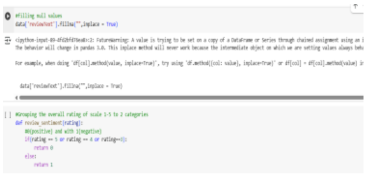
Joining review description and summary into one column

```
#joining review description and summary into one col
data['reviewText']=data['reviewText']+" "+data['summary']
data.drop(['summary'], axis=1,inplace=True)
```

```
#checking for null values
data.isna().sum()
```

```
              0
overall      74
reviewText  165

dtype: int64
```



```
#filling null values
data['reviewText'].fillna("",inplace = True)
```

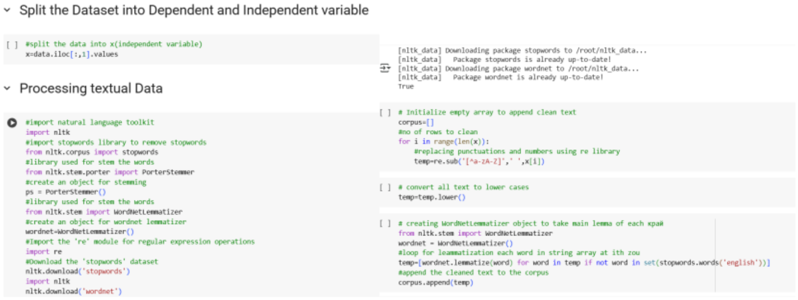
```
<ipython-input-89-4fd2bfd78ea8>:2: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an i
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always beha

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) in


  data['reviewText'].fillna("",inplace = True)
```

```
#Grouping the overall rating of scale 1-5 to 2 categories
def review_sentiment(rating):
    #0(positive) and with 1(negative)
    if(rating == 5 or rating == 4 or rating==3):
        return 0
    else:
        return 1
```

Splitting The Dataset Into Dependent And Independent Variable



Split the Dataset into Dependent and Independent variable

```
#split the data into x(independent variable)
x=data.iloc[:,1].values
```

Processing textual Data

```
#import natural language toolkit
import nltk
#import stopwords library to remove stopwords
from nltk.corpus import stopwords
#library used for stem the words
from nltk.stem.porter import PorterStemmer
#create an object for stemming
ps = PorterStemmer()
#library used for stem the words
from nltk.stem import WordNetLemmatizer
#create an object for wordnet lemmatizer
wordnet=WordNetLemmatizer()
#Import the 're' module for regular expression operations
import re
#Download the 'stopwords' dataset
nltk.download('stopwords')
import nltk
nltk.download('wordnet')
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
True
```

```
# Initialize empty array to append clean text
corpus=[]
#no of rows to clean
for i in range(len(x)):
    #replacing punctuations and numbers using re library
    temp=re.sub('[^a-zA-Z]',' ',x[i])
```

```
# convert all text to lower cases
temp=temp.lower()
```

```
# creating WordNetLemmatizer object to take main lemma of each край
from nltk.stem import WordNetLemmatizer
wordnet = WordNetLemmatizer()
#loop for leammatization each word in string array at ith zou
temp=[wordnet.lemmatize(word) for word in temp if not word in set(stopwords.words('english'))]
#append the cleaned text to the corpus
corpus.append(temp)
```