

Start coding or [generate](#) with AI.

```
from google.colab import files
uploaded = files.upload()
```




Choose Files



 No file chosen

```
import pandas as pd

# Load the dataset
df = pd.read_csv('/content/df_file.csv')

# Display the first 5 rows to verify
df.head()
```



	Text	Label	
0	Budget to set scene for election\n\n Gordon B...	0	
1	Army chiefs in regiments decision\n\n Militar...	0	
2	Howard denies split over ID cards\n\n Michael...	0	
3	Observers to monitor UK election\n\n Minister...	0	
4	Kilroy names election seat target\n\n Ex-chat...	0	

Next steps:

[Generate code with df](#)

 [View recommended plots](#)


[New interactive sheet](#)



1. Loading the Dataset and Displaying the First 5 Records

```
import pandas as pd
import numpy as np

# Load the dataset
df = pd.read_csv('df_file.csv')

# Display the first 5 records
df.head()
```



	Text	Label	
0	Budget to set scene for election\n\n Gordon B...	0	
1	Army chiefs in regiments decision\n\n Militar...	0	
2	Howard denies split over ID cards\n\n Michael...	0	
3	Observers to monitor UK election\n\n Minister...	0	
4	Kilroy names election seat target\n\n Ex-chat...	0	

Next steps:


[Generate code with df](#)

 [View recommended plots](#)

[New interactive sheet](#)


2. Finding the Shape (Rows, Columns) of the Dataset

```
df.shape
```

 (2225, 2)

3. Displaying the Column Names of the Dataset

```
df.columns
```

 Index(['Text', 'Label'], dtype='object')

4. Checking the Data Types of Each Column

```
df.dtypes
```

```

0
Text    object
Label   int64

dtype: object

```

5. Finding Missing (Null) Values

```
df.isnull().sum()
```

```

0
Text    0
Label    0

dtype: int64

```

6. Finding the Number of Unique Labels in the 'Label' Column

```
df['Label'].nunique()
```

```
5
```

7. Displaying the Count of Each Label (0s and 1s)

```
df['Label'].value_counts()
```

```

count
Label
1      511
4      510
0      417
2      401
3      386

dtype: int64

```

8. Calculating the Average Length of Text in the Dataset

```
df['text_length'] = df['Text'].apply(len)
np.mean(df['text_length'])
```

```
np.float64(2275.363595505618)
```

9. Finding the Minimum and Maximum Text Length

```
print("Minimum length:", df['text_length'].min())
print("Maximum length:", df['text_length'].max())
```

```

Minimum length: 507
Maximum length: 25597

```


10. Finding the Standard Deviation of Text Lengths

```
np.std(df['text_length'])
```

```
1370.4745873382267
```

11. Finding the Row with the Longest Text

```
df.loc[df['text_length'].idxmax()]
```




	63
Text	Terror powers expose 'tyranny'\n\n The Lord C...
Label	0
text_length	25597

dtype: object

12. Finding the Row with the Shortest Text

```
df.loc[df['text_length'].idxmin()]
```






	174
Text	Blunkett hints at election call\n\n Ex-Home S...
Label	0
text_length	507

dtype: object


13. Adding a New Column Showing the Number of Words in Each Text

```
df['word_count'] = df['Text'].apply(lambda x: len(str(x).split()))
df[['Text', 'word_count']].head()
```



	Text	word_count	
0	Budget to set scene for election\n\n Gordon B...	538	
1	Army chiefs in regiments decision\n\n Militar...	500	
2	Howard denies split over ID cards\n\n Michael...	540	
3	Observers to monitor UK election\n\n Minister...	496	
4	Kilroy names election seat target\n\n Ex-chat...	440	

14. Finding the Average Number of Words in the Texts

```
np.mean(df['word_count'])
 np.float64(384.04044943820224)
```

15. Displaying the Texts with More Than 100 Words

```
df[df['word_count'] > 100]
```



	Text	Label	text_length	word_count	
0	Budget to set scene for election\n \n Gordon B...	0	3147	538	
1	Army chiefs in regiments decision\n \n Militar...	0	3026	500	
2	Howard denies split over ID cards\n \n Michael...	0	3167	540	
3	Observers to monitor UK election\n \n Minister...	0	3094	496	
4	Kilroy names election seat target\n \n Ex-chat...	0	2614	440	
...	
2220	India opens skies to competition\n \n India wi...	4	1431	225	
2221	Yukos bankruptcy 'not US matter'\n \n Russian ...	4	2289	398	
2222	Survey confirms property slowdown\n \n Governm...	4	1883	322	
2223	High fuel prices hit BA's profits\n \n British...	4	2422	406	
2224	US trade gap hits record in 2004\n \n The gap ...	4	1962	328	

2224 rows × 4 columns

16. Sorting the Dataset Based on Word Count (Highest First)

```
df.sort_values(by='word_count', ascending=False)
```



	Text	Label	text_length	word_count	
63	Terror powers expose 'tyranny'\n \n The Lord C...	0	25597	4432	
1421	Scissor Sisters triumph at Brits\n \n US band ...	3	19288	3482	
412	Minimum wage increased to Â£5.05\n \n The mini...	0	18497	3295	
1078	Losing yourself in online gaming\n \n Online r...	2	16249	2969	
299	Kilroy launches 'Veritas' party\n \n Ex-BBC ch...	0	13921	2393	
...	
568	Yelling takes Cardiff hat-trick\n \n European ...	1	810	122	
753	Solskjaer raises hopes of return\n \n Manchest...	1	725	120	
866	Worcester v Sale (Fri)\n \n Sixways\n \n Frida...	1	829	115	
872	Tottenham bid £8m for Forest duo\n \n Not...	1	738	114	
174	Blunkett hints at election call\n \n Ex-Home S...	0	507	89	

2225 rows × 4 columns

17. Replacing Missing Text Values with "No Text"

```
df['Text'].fillna('No Text', inplace=True)
```



<ipython-input-23-8e8ebcf65de6>:1: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained assignment. The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values is a copy.
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, inplace=True)

```
df['Text'].fillna('No Text', inplace=True)
```

18. Finding How Many Texts Are Exactly Empty (Length 0)


```
len(df[df['text_length'] == 0])
```





0

19. Checking Basic Statistics of Numerical Columns


```
df.describe()
```





	Label	text_length	word_count	
count	2225.000000	2225.000000	2225.000000	
mean	2.027416	2275.363596	384.040449	
std	1.438694	1370.782663	238.174497	
min	0.000000	507.000000	89.000000	
25%	1.000000	1455.000000	246.000000	
50%	2.000000	1979.000000	332.000000	
75%	3.000000	2815.000000	471.000000	
max	4.000000	25597.000000	4432.000000	

20. Creating a Pivot Table Showing Average Text Length for Each Label

```
pd.pivot_table(df, values='text_length', index='Label', aggfunc=np.mean)
```



<ipython-input-26-4819488f6d09>:1: FutureWarning: The provided callable <function mean at 0x7fce76301ee0> is currently using DataFrameGr
pd.pivot_table(df, values='text_length', index='Label', aggfunc=np.mean)

	text_length	
Label		
0	2695.824940	
1	1906.545988	
2	2987.690773	
3	1938.230570	
4	1996.194118	