

```
import numpy as np
import pandas as pd

df = pd.read_csv('/content/IMDB Dataset.csv', encoding = 'ISO-8859-1')
df.head()
```

	review	sentiment
0	One of the other reviewers has mentioned that ...	positive
1	A wonderful little production. The...	positive
2	I thought this was a wonderful way to spend ti...	positive
3	Basically there's a family where a little boy ...	negative
4	Petter Mattei's "Love in the Time of Money" is...	positive

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 50000 entries, 0 to 49999
Data columns (total 2 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   review      50000 non-null   object
1   sentiment   50000 non-null   object
dtypes: object(2)
memory usage: 781.4+ KB
```

```
df.groupby('sentiment').describe()
```

sentiment	review			freq
	count	unique	top	
negative	25000	24698	This show comes up with interesting locations ...	3
positive	25000	24884	Loved today's show!!! It was a variety and not...	5

```
df.head()
```

	review	sentiment
0	One of the other reviewers has mentioned that ...	positive
1	A wonderful little production. The...	positive
2	I thought this was a wonderful way to spend ti...	positive
3	Basically there's a family where a little boy ...	negative
4	Petter Mattei's "Love in the Time of Money" is...	positive

```
df['result'] = df['sentiment'].apply(lambda x:1 if x=='positive' else 0)
```

```
df.head()
```

	review	sentiment	result
0	One of the other reviewers has mentioned that ...	positive	1
1	A wonderful little production. The...	positive	1
2	I thought this was a wonderful way to spend ti...	positive	1
3	Basically there's a family where a little boy ...	negative	0
4	Petter Mattei's "Love in the Time of Money" is...	positive	1

```
new_df = df[['sentiment','review','result']]
new_df.head()
```

	sentiment	review	result
0	positive	One of the other reviewers has mentioned that ...	1
1	positive	A wonderful little production. The...	1

```
from sklearn.model_selection import train_test_split as tts
x_train, x_test, y_train, y_test = tts(df.review, df.result)

v = CountVectorizer()

from sklearn.feature_extraction.text import CountVectorizer

v=CountVectorizer()
x_train_count=v.fit_transform(x_train.values)

from sklearn.naive_bayes import MultinomialNB
model=MultinomialNB()
model.fit(x_train_count,y_train)

MultinomialNB()

emails=["How are you brother?", "Free entry"]
email_count=v.transform(emails)
model.predict(email_count)

array([1, 0])

x_test_count=v.transform(x_test)
model.score(x_test_count,y_test)

0.84704
```