```
import pandas as pd

import matplotlib.pyplot as plt

import numpy as np

# Algorithms
from sklearn.naive_bayes import GaussianNB,MultinomialNB,BernoulliNB
from sklearn.svm import SVC
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.pipeline import Pipeline


data =pd.read_csv('/content/SPAM_DATASET.csv', encoding = "ISO-8859-1")


data.head(10)
```

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
0	ham	Go until jurong point, crazy.. Available only ...	NaN	NaN	NaN
1	ham	Ok lar... Joking wif u oni...	NaN	NaN	NaN
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...	NaN	NaN	NaN
3	ham	U dun say so early hor... U c already then say...	NaN	NaN	NaN
4	ham	Nah I don't think he goes to usf, he lives aro...	NaN	NaN	NaN
5	spam	FreeMsg Hey there darling it's been 3 week's n...	NaN	NaN	NaN
6	ham	Even my brother is not like to speak with me. ...	NaN	NaN	NaN
		As per your request 'Melle Melle (Oru			

```
#Drop empty columns
cols = [2,3,4]
data.drop(data.columns[cols],axis=1,inplace=True)
data.head(10)
```

	v1	v2
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina...
3	ham	U dun say so early hor... U c already then say...
4	ham	Nah I don't think he goes to usf, he lives aro...
5	spam	FreeMsg Hey there darling it's been 3 week's n...
6	ham	Even my brother is not like to speak with me. ...
7	ham	As per your request 'Melle Melle (Oru Minnamin...
8	spam	WINNER!! As a valued network customer you have...
9	spam	Had your mobile 11 months or more? U R entitle...

```
#Rename columns as category and message
data.rename(columns = {'v1':'Category', 'v2':'Message'}, inplace = True)
data.head(10)
```

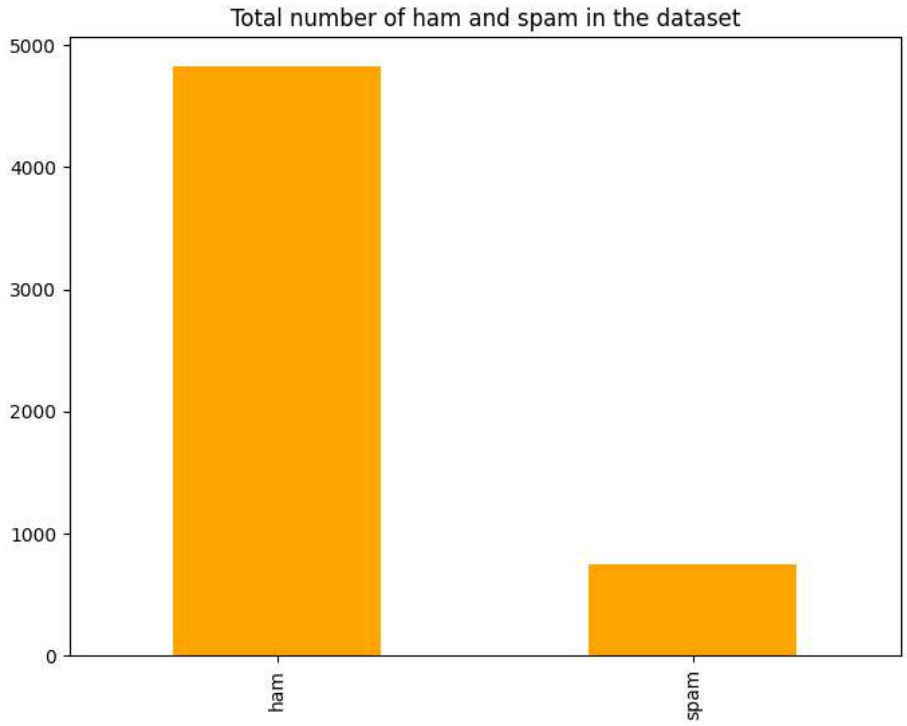
	Category	Message
0	ham	Go until jurong point, crazy.. Available only ...
1	ham	Ok lar... Joking wif u oni...
2	spam	Free entry in 2 a wkly comp to win FA Cup fina

```
print(f'Dataset consist of {data.shape[0]} E-Mails.')
```

```
Dataset consist of 5572 E-Mails.
```

```
plt.figure(figsize=(8,6))

data ['Category'].value_counts().plot.bar(color = ["orange","orange"])
plt.title('Total number of ham and spam in the dataset')
plt.show()
```



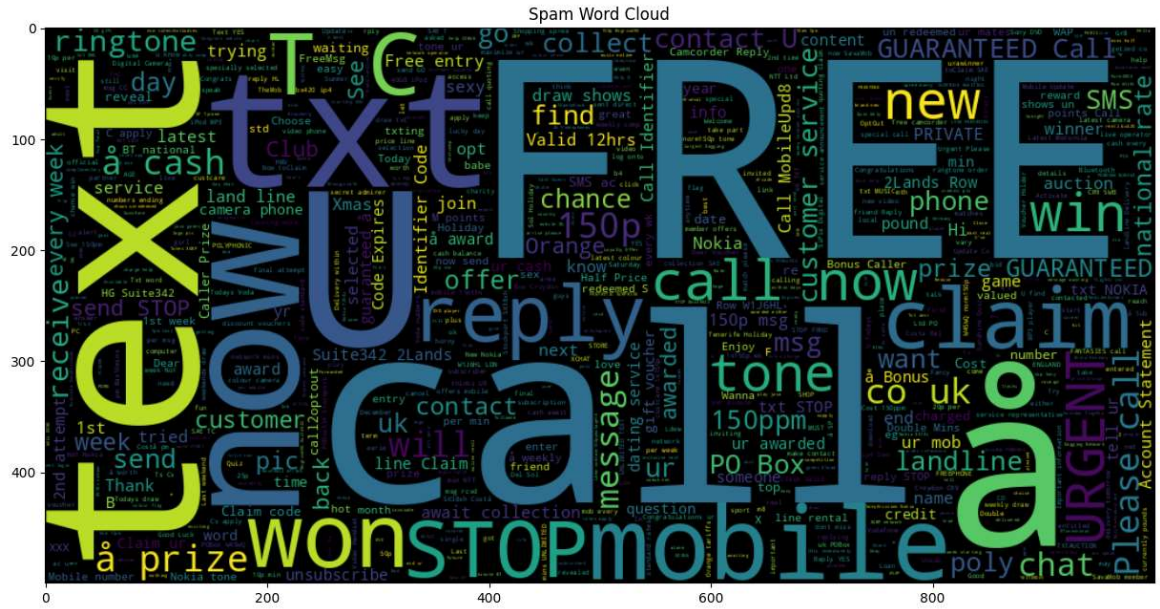

```
from wordcloud import WordCloud

plt.figure(figsize = (15,15))
wc = WordCloud(max_words = 2000 , width = 1000 , height = 500).generate(" ".join(data[data.Category =="ham" ].Message))
plt.imshow(wc , interpolation = 'bilinear')
plt.title("Ham Word Cloud")
```

```
Text(0.5, 1.0, 'Ham Word Cloud')
```

```
plt.figure(figsize = (15,15))
wc = WordCloud(max_words = 2000 , width = 1000 , height = 500).generate(" ".join(data[data.Category =="spam" ].Message))
plt.imshow(wc , interpolation = 'bilinear')
plt.title("Spam Word Cloud")
```

```
Text(0.5, 1.0, 'Spam Word Cloud')
```




```
#0: Ham, 1: Spam
data['Category']=data['Category'].apply(lambda x: 1 if x=='spam' else 0)
data.head()
```

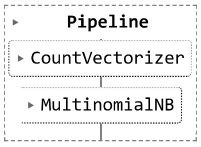
	Category	Message
0	0	Go until jurong point, crazy.. Available only ...
1	0	Ok lar... Joking wif u oni...
2	1	Free entry in 2 a wkly comp to win FA Cup fina...
3	0	U dun say so early hor... U c already then say...
4	0	Nah I don't think he goes to usf, he lives aro...

```
X=data['Message']
Y=data['Category']
```

```
X_train, X_test, y_train, y_test = train_test_split(X,Y)
```

```
#Defineing Naive Baised
clf_NaiveBaised= Pipeline([
    ('vectorizer', CountVectorizer()),
    ('nd', MultinomialNB())
])
```

```
#Fiting the algorithm
clf_NaiveBaised.fit(X_train, y_train)
```




```
#Make prediction on X_test
y_pred_NB=clf_NaiveBaised.predict(X_test)
```

```
naive_acc=accuracy_score(y_test,y_pred_NB)
naive_acc
```

```
0.9820531227566404
```

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB
import numpy as np

def spam_detect(clf, txt):
    a = clf.predict(txt)
    if a == 1:
        print("This is a Spam email\n")
    else:
        print("This is a Real email\n")

message = input("Enter the email message: ")

# Example training data
X_train = np.array(['Buy our products now!',
           'Get a discount on our services.',
           'Hello, please find attached the meeting agenda.',
           'Reminder: Your appointment is tomorrow.'])

y_train = np.array([1, 1, 0, 0])

# Convert text data to numeric features
vectorizer = CountVectorizer()
X_train_numeric = vectorizer.fit_transform(X_train)

# Initialize the Naive Bayes classifier
clf = MultinomialNB()

# Fit the classifier with the training data
clf.fit(X_train_numeric, y_train)

# Convert the input message to numeric features
message_numeric = vectorizer.transform([message])

# Test the email message
spam_detect(clf, message_numeric)
```

```
Enter the email message: i am fine
This is a Real email
```