In [18]:

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
```

In [19]:

```
import pandas as pd

# Try using a different encoding, such as 'ISO-8859-1' or 'latin1'
raw_mail_data = pd.read_csv('C:\\Users\\vaibhav vishal\\OneDrive\\Documents
```

In [20]:

```
print(raw_mail_data)
```

```
        v1                                                  v2 Unnamed: 2  \
0      ham  Go until jurong point, crazy.. Available only ...        NaN   
1      ham                       Ok lar... Joking wif u oni...        NaN   
2     spam  Free entry in 2 a wkly comp to win FA Cup fina...        NaN   
3      ham  U dun say so early hor... U c already then say...        NaN   
4      ham  Nah I don't think he goes to usf, he lives aro...        NaN   
...    ...                                                ...        ...   
5567  spam  This is the 2nd time we have tried 2 contact u...        NaN   
5568   ham               Will Ì_ b going to esplanade fr home?        NaN   
5569   ham  Pity, * was in mood for that. So...any other s...        NaN   
5570   ham  The guy did some bitching but I acted like i'd...        NaN   
5571   ham                          Rofl. Its true to its name        NaN   

     Unnamed: 3 Unnamed: 4  
0           NaN        NaN  
1           NaN        NaN  
2           NaN        NaN  
3           NaN        NaN  
4           NaN        NaN  
...         ...        ...  
5567        NaN        NaN  
5568        NaN        NaN  
5569        NaN        NaN  
5570        NaN        NaN  
5571        NaN        NaN  

[5572 rows x 5 columns]
```

In [21]:

```
# replace the null values with a null string
mail_data = raw_mail_data.where((pd.notnull(raw_mail_data)),'')
```

In [22]:

```
# printing the first 5 rows of the dataframe
mail_data.head()
```

Out[22]:

	v1	v2	Unnamed: 2	Unnamed: 3	Unnamed: 4
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...			

In [23]:

```
# checking the number of rows and columns in the dataframe
mail_data.shape
```

Out[23]:

```
(5572, 5)
```

In [24]:

```
# Rename the columns "v1 and "v2" to new names
new_column_names = {"v1":"Category","v2":"Message"}
mail_data.rename(columns = new_column_names,inplace = True)
```

In [25]:

```
mail_data.head()
```

Out[25]:

	Category	Message	Unnamed: 2	Unnamed: 3	Unnamed: 4
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...			

In [26]:

```
# label spam mail as 0;  ham mail as 1;

mail_data.loc[mail_data['Category'] == 'spam', 'Category',] = 0
mail_data.loc[mail_data['Category'] == 'ham', 'Category',] = 1
```

In [27]:

```
# separating the data as texts and label

X = mail_data['Message']

Y = mail_data['Category']
```

In [28]:

```python
print(X)
```

```
0       Go until jurong point, crazy.. Available only ...
1                           Ok lar... Joking wif u oni...
2       Free entry in 2 a wkly comp to win FA Cup fina...
3       U dun say so early hor... U c already then say...
4       Nah I don't think he goes to usf, he lives aro...
                              ...                        
5567    This is the 2nd time we have tried 2 contact u...
5568                 Will Ì_ b going to esplanade fr home?
5569    Pity, * was in mood for that. So...any other s...
5570    The guy did some bitching but I acted like i'd...
5571                           Rofl. Its true to its name
Name: Message, Length: 5572, dtype: object
```

In [29]:

```python
print(Y)
```

```
0       1
1       1
2       0
3       1
4       1
       ..
5567    0
5568    1
5569    1
5570    1
5571    1
Name: Category, Length: 5572, dtype: object
```

In [30]:

```python
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, ra
```

In [31]:

```python
print(X.shape)
print(X_train.shape)
print(X_test.shape)
```

```
(5572,)
(4457,)
(1115,)
```

In [34]:

```python
from sklearn.feature_extraction.text import TfidfVectorizer

# Initialize the vectorizer with lowercase set to True (boolean, not string
feature_extraction = TfidfVectorizer(min_df=1, stop_words='english', lowerc

# Fit on the training data and transform it to TF-IDF features
X_train_features = feature_extraction.fit_transform(X_train)

# Transform the test data to TF-IDF features using the same vectorizer
X_test_features = feature_extraction.transform(X_test)

# Ensure the labels are integers
Y_train = Y_train.astype(int)
Y_test = Y_test.astype(int)
```

```
In [35]: print(X_train)
```

```
3075    Mum, hope you are having a great day. Hoping t...
1787                           Yes:)sura in sun tv.:)lol.
1614    Me sef dey laugh you. Meanwhile how's my darli...
4304                Yo come over carlos will be here soon
3266                    Ok then i come n pick u at engin?
                              ...                        
789                          Gud mrng dear hav a nice day
968             Are you willing to go for aptitude class.
1667    So now my dad is gonna call after he gets out ...
3321    Ok darlin i supose it was ok i just worry too ...
1688                     Nan sonathaya soladha. Why boss?
Name: Message, Length: 4457, dtype: object
```

In [36]:
```
print(X_train_features)
```

```
  (0, 741)      0.3219352588930141
  (0, 3979)     0.2410582143632299
  (0, 4296)     0.3891385935794867
  (0, 6599)     0.20296878731699391
  (0, 3386)     0.3219352588930141
  (0, 2122)     0.38613577623520473
  (0, 3136)     0.440116181574609
  (0, 3262)     0.25877035357606315
  (0, 3380)     0.21807195185332803
  (0, 4513)     0.2909649098524696
  (1, 4061)     0.380431198316959
  (1, 6872)     0.4306015894277422
  (1, 6417)     0.4769136859540388
  (1, 6442)     0.5652509076654626
  (1, 7443)     0.35056971070320353
  (2, 933)      0.4917598465723273
  (2, 2109)     0.42972812260098503
  (2, 3917)     0.40088501350982736
  (2, 2226)     0.413484525934624
  (2, 5825)     0.4917598465723273
  (3, 6140)     0.4903863168693604
  (3, 1599)     0.5927091854194291
  (3, 1842)     0.3708680641487708
  (3, 7453)     0.5202633571003087
  (4, 2531)     0.7419319091456392
  :     :
  (4452, 2122)  0.31002103760284144
  (4453, 999)   0.6760129013031282
  (4453, 7273)  0.5787739591782677
  (4453, 1762)  0.45610005640082985
  (4454, 3029)  0.42618909997886
  (4454, 2086)  0.3809693742808703
  (4454, 3088)  0.34475593009514444
  (4454, 2001)  0.4166919007849217
  (4454, 1049)  0.31932060116006045
  (4454, 7346)  0.31166263834107377
  (4454, 5370)  0.42618909997886
  (4455, 1148)  0.38998123077430413
  (4455, 6433)  0.38998123077430413
  (4455, 6361)  0.25697343671652706
  (4455, 2764)  0.3226323745940581
  (4455, 7358)  0.2915949626395065
  (4455, 7407)  0.3028481995557642
  (4455, 2108)  0.3136468384526087
  (4455, 4251)  0.30616657078392584
  (4455, 3763)  0.16807158405536876
  (4455, 4773)  0.35860460546223444
  (4456, 6117)  0.5304350313291551
  (4456, 6133)  0.5304350313291551
  (4456, 1386)  0.4460036316446079
  (4456, 4557)  0.48821933148688146
```

In [37]:
```
model = LogisticRegression()
```

In [38]:

```
# training the Logistic Regression model with the training data
model.fit(X_train_features, Y_train)
```

Out[38]:

```
LogisticRegression()
```

In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with nbviewer.org.

In [ ]:

In [39]:

```
# prediction on training data

prediction_on_training_data = model.predict(X_train_features)
accuracy_on_training_data = accuracy_score(Y_train, prediction_on_training_
```

In [43]:

```
print('Accuracy on training data : ', accuracy_on_training_data)
```

```
Accuracy on training data :  0.9661207089970832
```

In [44]:

```
# prediction on test data

prediction_on_test_data = model.predict(X_test_features)
accuracy_on_test_data = accuracy_score(Y_test, prediction_on_test_data)
```

In [45]:

```
print('Accuracy on test data : ', accuracy_on_test_data)
```

```
Accuracy on test data :  0.9623318385650225
```

In [46]:

```
input_mail = ["I've been searching for the right words to thank you for thi

# convert text to feature vectors
input_data_features = feature_extraction.transform(input_mail)

# making prediction

prediction = model.predict(input_data_features)
print(prediction)


if (prediction[0]==1):
  print('Ham mail')

else:
  print('Spam mail')
```

```
[1]
Ham mail
```

In [ ]: