```
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.svm import LinearSVC
from sklearn.metrics import accuracy_score
```

```
raw_mail_data=pd.read_csv('spamham.csv')
mail_data=raw_mail_data.where((pd.notnull(raw_mail_data)),'')
```

```
mail_data.shape
```

```
(5171, 4)
```

```
mail_data.head()
```

	Unnamed: 0	label	text	label_num
0	605	ham	Subject: enron methanol ; meter # : 988291\r\n...	0
1	2349	ham	Subject: hpl nom for january 9 , 2001\r\n(see...	0
2	3624	ham	Subject: neon retreat\r\nho ho ho , we ' re ar...	0
3	4685	spam	Subject: photoshop , windows , office . cheap ...	1
4	2030	ham	Subject: re : indian springs\r\nthis deal is t...	0

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

```
0       False
1       False
2       False
4       False
5       False
        ...  
5165    False
5166    False
5167    False
5168    False
5169    False
Name: label, Length: 3672, dtype: bool
```

```
X= mail_data['text']
Y=mail_data['label_num']
```

```
print()
```

```
0       Subject: enron methanol ; meter # : 988291\r\n...
1       Subject: hpl nom for january 9 , 2001\r\n( see...
2       Subject: neon retreat\r\nho ho ho , we ' re ar...
3       Subject: photoshop , windows , office . cheap ...
4       Subject: re : indian springs\r\nthis deal is t...
                              ...
5166    Subject: put the 10 on the ft\r\nthe transport...
5167    Subject: 3 / 4 / 2000 and following noms\r\nhp...
5168    Subject: calpine daily gas nomination\r\n>\r\n...
5169    Subject: industrial worksheets for august 2000...
5170    Subject: important online banking alert\r\ndea...
Name: text, Length: 5171, dtype: object
0       0
1       0
2       0
3       1
4       0
       ..
5166    0
5167    0
5168    0
5169    0
5170    1
Name: label_num, Length: 5171, dtype: int64
```

```
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,train_size=0.8,test_size=0.2,random_sta
```

```
feature_extraction=TfidfVectorizer(min_df=1,stop_words='english',lowercase=1)
X_train_features=feature_extraction.fit_transform(X_train)
X_test_features=feature_extraction.transform(X_test)

Y_train=Y_train.astype(int)
Y_test=Y_test.astype(int)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/utils/_param_validation.py:558: Futu
  warnings.warn(
```

```
model=LinearSVC()
model.fit(X_train_features,Y_train)
```

```
▾ LinearSVC
LinearSVC()
```

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

```
print("accuracy on training data:",accuracy_on_training_data)
```

```
     accuracy on training data: 1.0
```

```
prediction_on_testing_data=model.predict(X_test_features)
accuracy_on_testing_data=accuracy_score(Y_test,prediction_on_testing_data)
print("accuracy on testing data:",accuracy_on_testing_data)
```

```
     accuracy on testing data: 0.9864734299516909
```

```
input_data=["nominations for oct . 21 - 23 , 2000 ( see attached file : hplnl 021 . xls )-
```

```
input_mail_features=feature_extraction.transform(input_data)
```

```
prediction=model.predict(input_mail_features)
print(prediction)
```

```
     [0]
```

```
if prediction[0]==0:
  print("the mail is a spam mail")
else:
  print("the mail is a ham mail")
```

```
     the mail is a spam mail
```