In [1]:
```
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt

import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords

from nltk.stem.porter import PorterStemmer

from sklearn.feature_extraction.text import CountVectorizer

from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import MultinomialNB

from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\KIIT\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

In [2]:
```
df = pd.read_csv('spam.csv', encoding='latin1')
df
```

Out[2]:

	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
...	...	...	...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...	NaN	NaN	NaN
5568	ham	Will ì_ b going to esplanade fr home?	NaN	NaN	NaN
5569	ham	Pity, * was in mood for that. So...any other s...	NaN	NaN	NaN
5570	ham	The guy did some bitching but I acted like i'd...	NaN	NaN	NaN
5571	ham	Rofl. Its true to its name	NaN	NaN	NaN

5572 rows × 5 columns

In [3]:
```
df.isnull().sum()
```

Out[3]:
```
v1              0
v2              0
Unnamed: 2    5522
Unnamed: 3    5560
Unnamed: 4    5566
dtype: int64
```

In [4]:
```
df = df[['v1', 'v2']]
df.columns = ['label', 'message']
df
```

Out[4]:

	label	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...
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...
...	...	...
5567	spam	This is the 2nd time we have tried 2 contact u...
5568	ham	Will ì_ b going to esplanade fr home?
5569	ham	Pity, * was in mood for that. So...any other s...
5570	ham	The guy did some bitching but I acted like i'd...
5571	ham	Rofl. Its true to its name

5572 rows × 2 columns

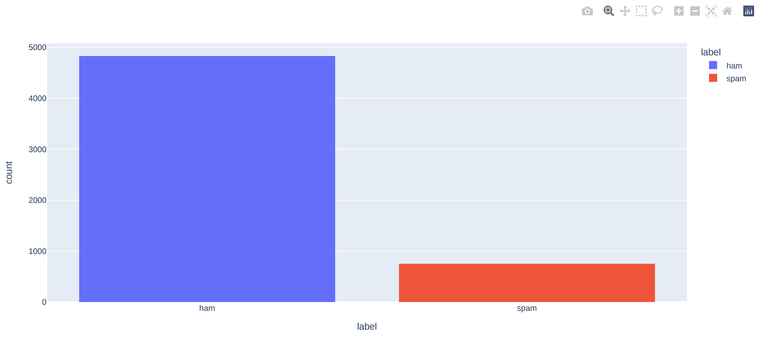
In [5]:
```
df.shape
```

Out[5]:
```
(5572, 2)
```

In [6]:
```
df.groupby('label').size()
```

Out[6]:
```
label
ham     4825
spam     747
dtype: int64
```

In [7]:
```
import plotly.express as px
px.histogram(df, x="label", color="label")
```



In [8]:
```
ps = PorterStemmer()
reg = []
for i in range(0, len(df)):
    review = re.sub('[^a-zA-Z]', ' ', df['message'][i])
    review = review.lower()
    review = review.split()
    review = [ps.stem(word) for word in review if not word in stopwords.words('english')]
    review = ' '.join(review)
    reg.append(review)

reg[1:10]
```

Out[8]:
```
['ok lar joke wif u oni',
 'free entri wkli comp win fa cup final tkt st may text fa receiv entri question std txt rate c appli',
 'u dun say earli hor u c alreadi say',
 'nah think goe usf live around though',
 'freemsg hey darl week word back like fun still tb ok xxx std chg send rcv',
 'even brother like speak treat like aid patent',
 'per request mell mell oru minnaminungint nurungu vettam set callertun caller press copi friend callertun',
 'winner valu network custom select receivea prize reward claim call claim code kl valid hour',
 'mobil month u r entitl updat latest colour mobil camera free call mobil updat co free']
```

In [9]:
```
cv = CountVectorizer(max_features = 4000)
X = cv.fit_transform(reg).toarray()
Y = pd.get_dummies(df['label'])
Y = Y.iloc[:, 1].values
```

In [10]:
```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.2, random_state=40)
```

In [11]:
```
model = MultinomialNB()
model.fit(X_train, Y_train)
```

Out[11]:
```
MultinomialNB()
```

In [12]:
```
pred = model.predict(X_test)
```

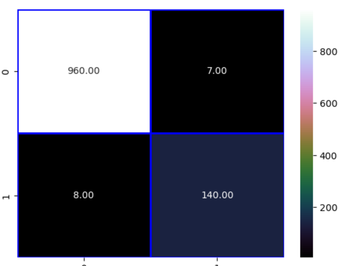
In [13]:
```
print("Multinomial Naive Bayes")
print("Confusion Matrix: ")
print(confusion_matrix(Y_test, pred))
print("Accuracy: ", accuracy_score(Y_test, pred))
```

```
Multinomial Naive Bayes
Confusion Matrix:
[[960   7]
 [  8 140]]
Accuracy:  0.9865470852017937
```

In [14]:
```
from sklearn.metrics import confusion_matrix
conf = confusion_matrix(Y_test, pred)

import seaborn as sns
sns.heatmap(conf, annot=True, cmap="cubehelix", linecolor="blue", linewidth=1.0,fmt="0.2f")
```

Out[14]:
```
<AxesSubplot:>
```



In [15]:
```
from sklearn.metrics import classification_report
report = classification_report(Y_test, pred)
print("Classification Report for MNB \n", report)
```

```
Classification Report for MNB
               precision    recall  f1-score   support

           0       0.99      0.99      0.99       967
           1       0.95      0.95      0.95       148

    accuracy                           0.99      1115
   macro avg       0.97      0.97      0.97      1115
weighted avg       0.99      0.99      0.99      1115
```

In [ ]: