1. Make necessary imports:

```
1.  import numpy as np
2.  import pandas as pd
3.  import itertools
4.  from sklearn.model_selection
import train_test_split
5.  from
sklearn.feature_extraction.tex
import TfidfVectorizer
6.  from sklearn.linear_model
import
PassiveAggressiveClassifier
7.  from sklearn.metrics import
accuracy_score, confusion_matr
```

2. Now, let's read the data into a DataFrame, and get the shape of the data and the first 5 records.

```
#Read the data

df=pd.read_csv('D:\\DataFlair\

#Get shape and head
df.shape
df.head()
```

3. And get the labels from the DataFrame.

```
#DataFlair - Get the labels
labels=df.label
labels.head()
```

4. Split the dataset into training and testing sets.

5. Let's initialize a TfidfVectorizer with stop words from the English language and a maximum document frequency of 0.7 (terms with a higher document frequency will be discarded). Stop words are the most common words in a language that are to be filtered out before processing the natural language data. And a TfidfVectorizer turns a collection of raw documents into a matrix of TF-IDF features.

Now, fit and transform the vectorizer on the train set, and transform the vectorizer on the test set.

6. Next, we'll initialize a PassiveAggressiveClassifier. This is. We'll fit this on tfidf_train and y_train.

Then, we'll predict on the test set from the TfidfVectorizer and calculate the accuracy with accuracy_score() from sklearn.metrics.

7. We got an accuracy of 92.82% with this model. Finally, let's print out a confusion matrix to gain insight into the number of false and true negatives and positives.