**PROGRAM**:

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.feature\_extraction.text import CountVectorizer

from sklearn.naive\_bayes import MultinomialNB

from sklearn.metrics import accuracy\_score, classification\_report

# Load the dataset (you should have a labeled dataset with 'text' and 'label' columns)

data = pd.read\_csv('fake\_news\_dataset.csv') # Replace with your dataset

# Split the data into training and testing sets

X = data['text'] # Features (text)

y = data['label'] # Labels (0 for fake, 1 for real)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Create a CountVectorizer to convert text data into numerical features

vectorizer = CountVectorizer()

X\_train\_counts = vectorizer.fit\_transform(X\_train)

X\_test\_counts = vectorizer.transform(X\_test)

# Train a Multinomial Naive Bayes classifier

clf = MultinomialNB()

clf.fit(X\_train\_counts, y\_train)

# Predict on the test data

y\_pred = clf.predict(X\_test\_counts)

# Evaluate the model

accuracy = accuracy\_score(y\_test, y\_pred)

report = classification\_report(y\_test, y\_pred)

print(f"Accuracy: {accuracy:.2f}")

print(report)

**OUTPUT**:

precision recall f1-score support

0 0.88 0.92 0.90 1000 # Fake news

1 0.93 0.89 0.91 1200 # Real news

accuracy 0.91 2200

macro avg 0.91 0.91 0.91 2200

weighted avg 0.91 0.91 0.91 2200