PRACTICAL SUBMISSION RECORD- A.Y. 2025-26

Class: SYMCA Division : A **Semester: III**	**Course Code:** MCA01604 **Course Name:** Data Science Laboratory	**Batch:** S2
Name: Pranav Raju Malwatkar	**Roll No:** 52037	
CO No: CO605.1	**Assignment No:** 4	

Title : Use the Naive Bayes classifier to perform sentiment analysis on a dataset of movie reviews, classifying them as either positive or negative based on the review text.

Code:

```
# Install required packages
# !pip install nltk scikit-learn matplotlib

import pandas as pd
import matplotlib.pyplot as plt
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 confusion_matrix, ConfusionMatrixDisplay, accuracy_score

# Download and load NLTK movie review
dataset import nltk
nltk.download('movie_reviews')
from nltk.corpus import movie_reviews

# Load the data
docs = [(movie_reviews.raw(fileid),
    category) for category in
    movie_reviews.categories()
    for fileid in movie_reviews.fileids(category)]

# Create a DataFrame
df = pd.DataFrame(docs, columns=['review', 'sentiment'])

# Encode target labels
df['sentiment'] = df['sentiment'].map({'pos': 1, 'neg': 0})
```

```python
# Split the dataset
X_train, X_test, y_train, y_test = train_test_split(df['review'], df['sentiment'], test_size=0.2,
random_state=42)

# Text vectorization
vectorizer = CountVectorizer(stop_words='english')
X_train_vec = vectorizer.fit_transform(X_train)

X_test_vec = vectorizer.transform(X_test)

# Train Naive Bayes
classifier model =
MultinomialNB()
model.fit(X_train_vec,
y_train)

# Predict
y_pred = model.predict(X_test_vec)

# Confusion Matrix
cm = confusion_matrix(y_test, y_pred)
disp = ConfusionMatrixDisplay(confusion_matrix=cm, display_labels=["Negative", "Positive"])
disp.plot(cmap='Blues')
plt.title("Confusion Matrix - Naive Bayes Sentiment
Analysis") plt.show()
```

Output: