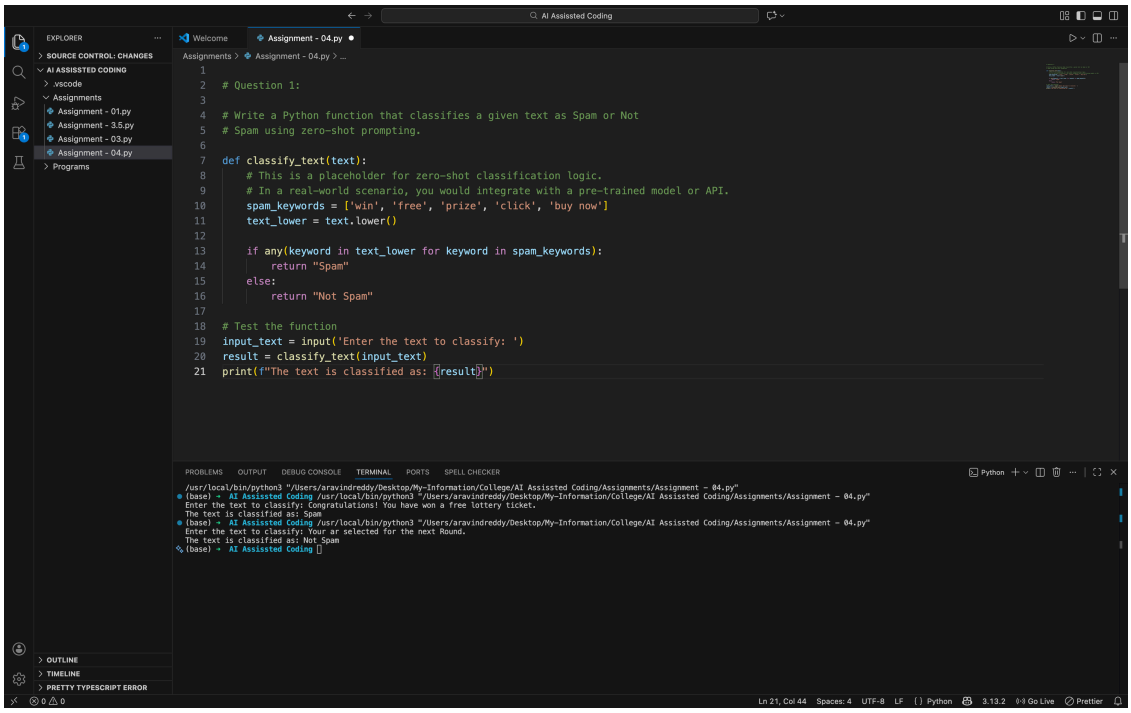
ASSIGNMENT - 04

Hall Ticket Number : 2303a51027 **Name** : Aravind Reddy **Batch NO :** 01

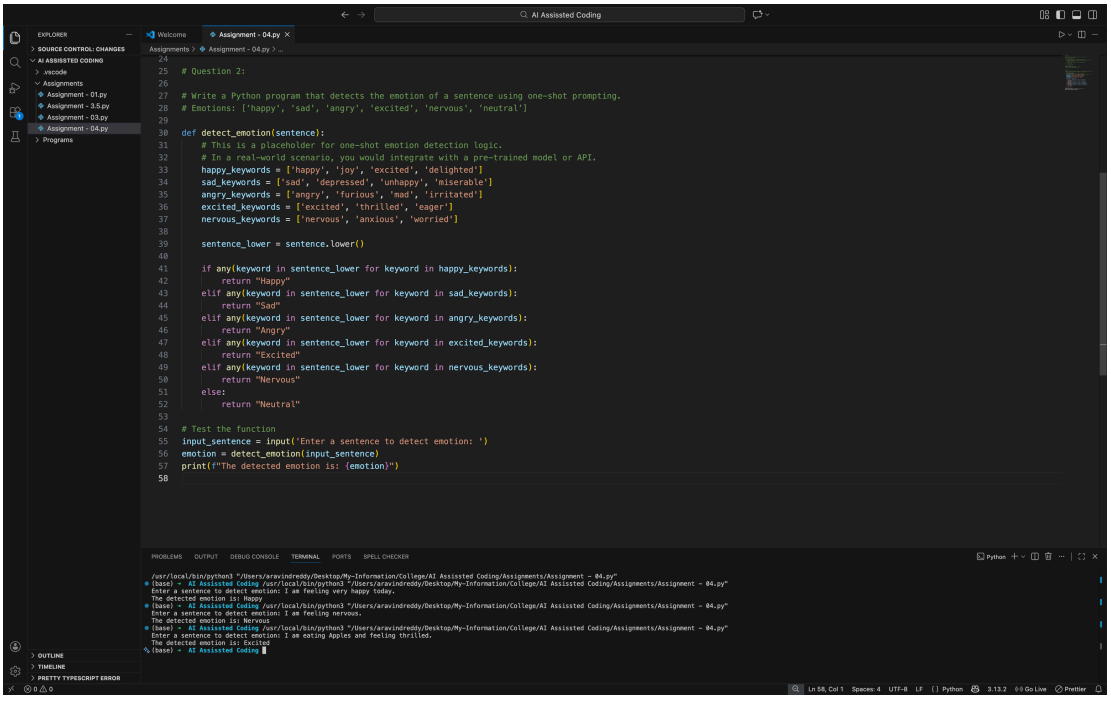
Q1. Zero-Shot Prompting (Basic Lab Task)



```python
# Question 1:

# Write a Python function that classifies a given text as Spam or Not
# Spam using zero-shot prompting.

def classify_text(text):
    # This is a placeholder for zero-shot classification logic.
    # In a real-world scenario, you would integrate with a pre-trained model or API.
    spam_keywords = ['win', 'free', 'prize', 'click', 'buy now']
    text_lower = text.lower()

    if any(keyword in text_lower for keyword in spam_keywords):
        return "Spam"
    else:
        return "Not Spam"

# Test the function
input_text = input('Enter the text to classify: ')
result = classify_text(input_text)
print(f"The text is classified as: {result}")
```

```
/usr/local/bin/python3 "/Users/aravindreddy/Desktop/My-Information/College/AI Assissted Coding/Assignments/Assignment - 04.py"
(base) → AI Assissted Coding /usr/local/bin/python3 "/Users/aravindreddy/Desktop/My-Information/College/AI Assissted Coding/Assignments/Assignment - 04.py"
Enter the text to classify: Congratulations! You have won a free lottery ticket.
The text is classified as: Spam
(base) → AI Assissted Coding /usr/local/bin/python3 "/Users/aravindreddy/Desktop/My-Information/College/AI Assissted Coding/Assignments/Assignment - 04.py"
Enter the text to classify: Your ar selected for the next Round.
The text is classified as: Not Spam
(base) → AI Assissted Coding
```

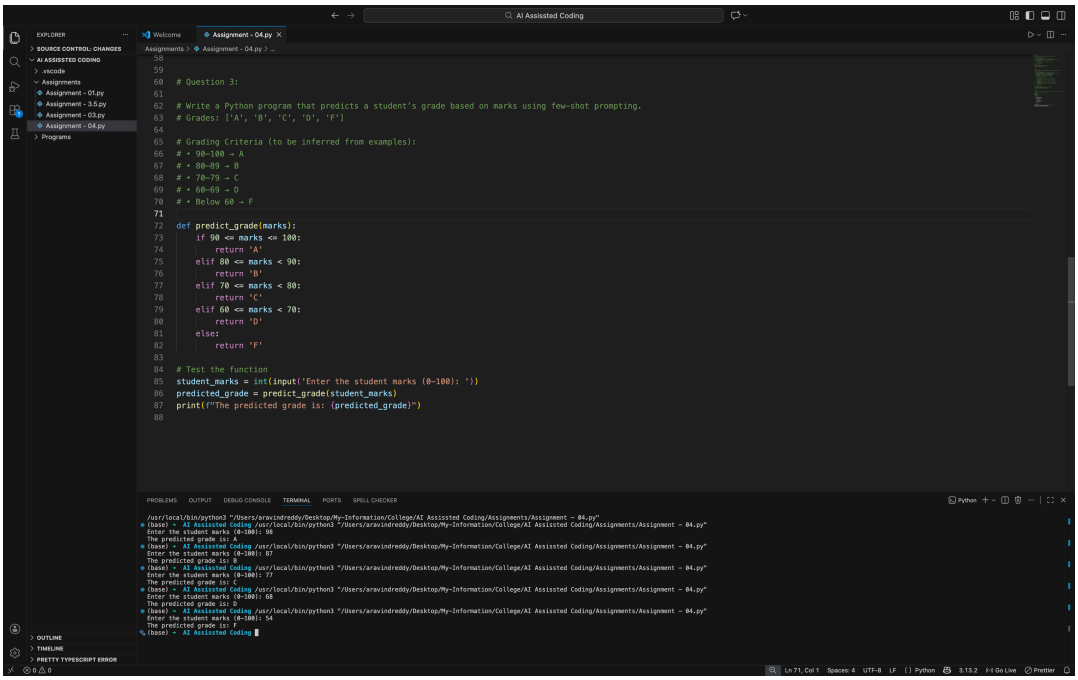
Q2. One-Shot Prompting (Emotion detection)



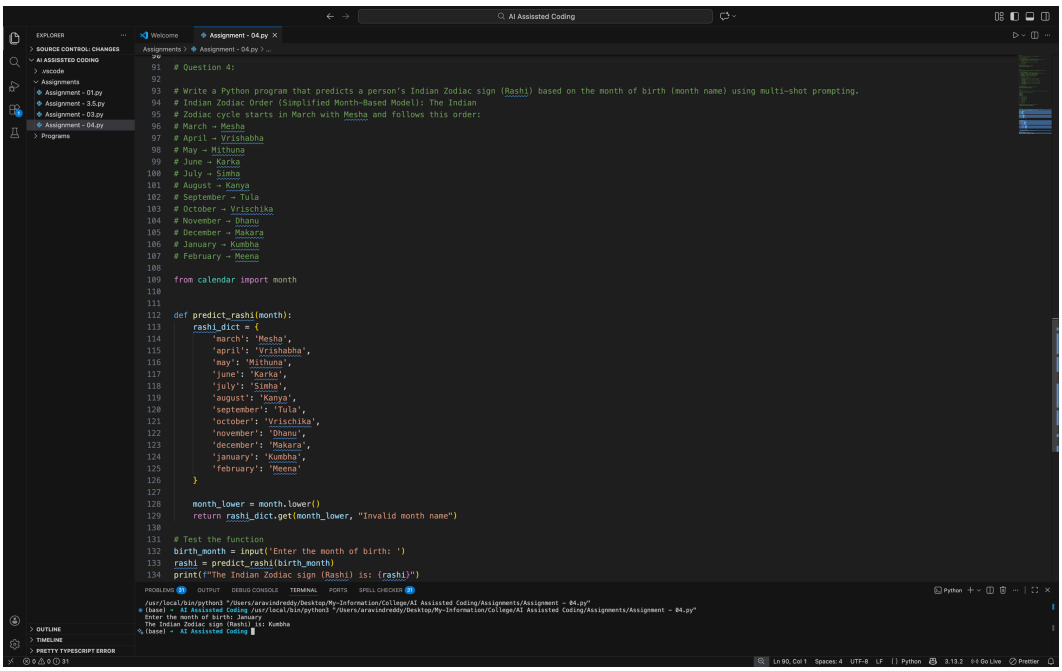
```python
# Question 2:

# Write a Python program that detects the emotion of a sentence using one-shot prompting.
# Emotions: ['happy', 'sad', 'angry', 'excited', 'nervous', 'neutral']

def detect_emotion(sentence):
    # This is a placeholder for one-shot emotion detection logic.
    # In a real-world scenario, you would integrate with a pre-trained model or API.
    happy_keywords = ['happy', 'joy', 'excited', 'delighted']
    sad_keywords = ['sad', 'depressed', 'unhappy', 'miserable']
    angry_keywords = ['angry', 'furious', 'mad', 'irritated']
    excited_keywords = ['excited', 'thrilled', 'eager']
    nervous_keywords = ['nervous', 'anxious', 'worried']

    sentence_lower = sentence.lower()

    if any(keyword in sentence_lower for keyword in happy_keywords):
        return "Happy"
    elif any(keyword in sentence_lower for keyword in sad_keywords):
        return "Sad"
    elif any(keyword in sentence_lower for keyword in angry_keywords):
        return "Angry"
    elif any(keyword in sentence_lower for keyword in excited_keywords):
        return "Excited"
    elif any(keyword in sentence_lower for keyword in nervous_keywords):
        return "Nervous"
    else:
        return "Neutral"

# Test the function
input_sentence = input('Enter a sentence to detect emotion: ')
emotion = detect_emotion(input_sentence)
print(f"The detected emotion is: {emotion}")
```

```
/usr/local/bin/python3 "/Users/aravindreddy/Desktop/My-Information/College/AI Assissted Coding/Assignments/Assignment - 04.py"
(base) → AI Assissted Coding /usr/local/bin/python3 "/Users/aravindreddy/Desktop/My-Information/College/AI Assissted Coding/Assignments/Assignment - 04.py"
Enter a sentence to detect emotion: I am feeling very happy today.
The detected emotion is: Happy
(base) → AI Assissted Coding /usr/local/bin/python3 "/Users/aravindreddy/Desktop/My-Information/College/AI Assissted Coding/Assignments/Assignment - 04.py"
Enter a sentence to detect emotion: I am feeling nervous.
The detected emotion is: Nervous
(base) → AI Assissted Coding /usr/local/bin/python3 "/Users/aravindreddy/Desktop/My-Information/College/AI Assissted Coding/Assignments/Assignment - 04.py"
Enter a sentence to detect emotion: I am eating Apples and feeling thrilled.
The detected emotion is: Excited
(base) → AI Assissted Coding
```

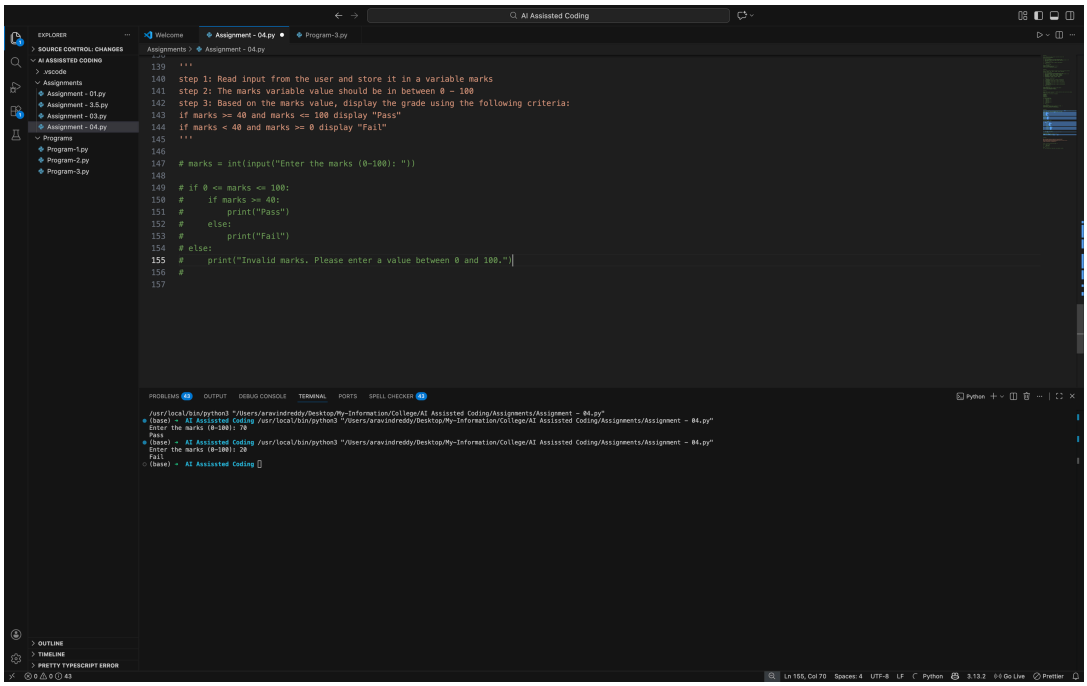
Q3. Few-Shot Prompting (Student Grading Based on Marks)



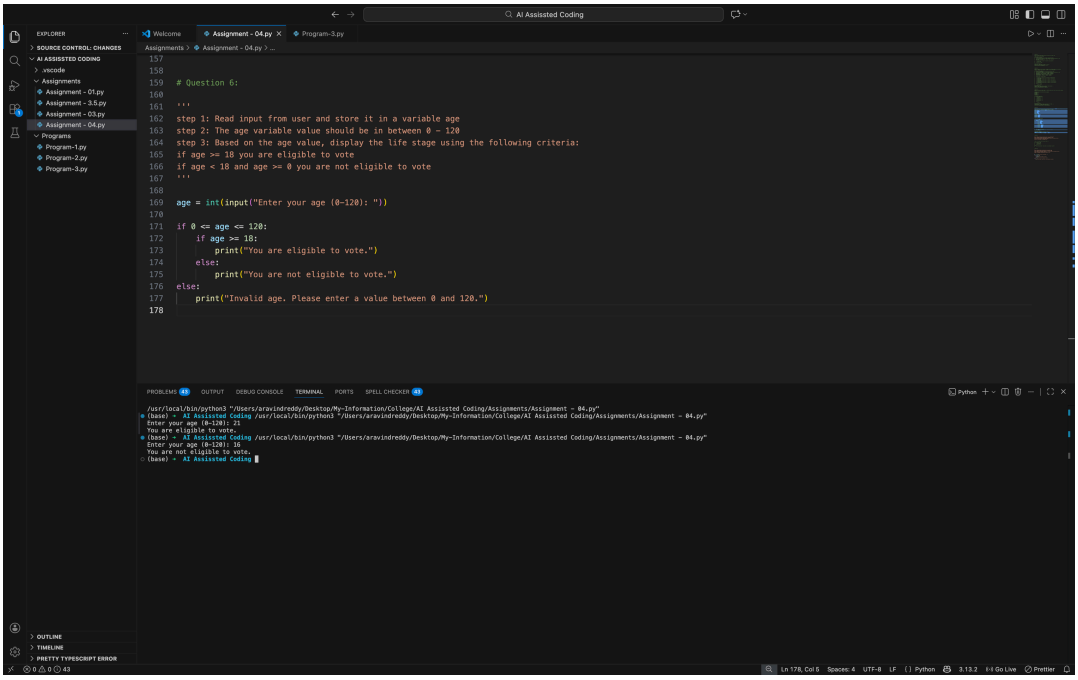
Q4. Multi-Shot Prompting (Indian Zodiac Sign Prediction using Month Name)



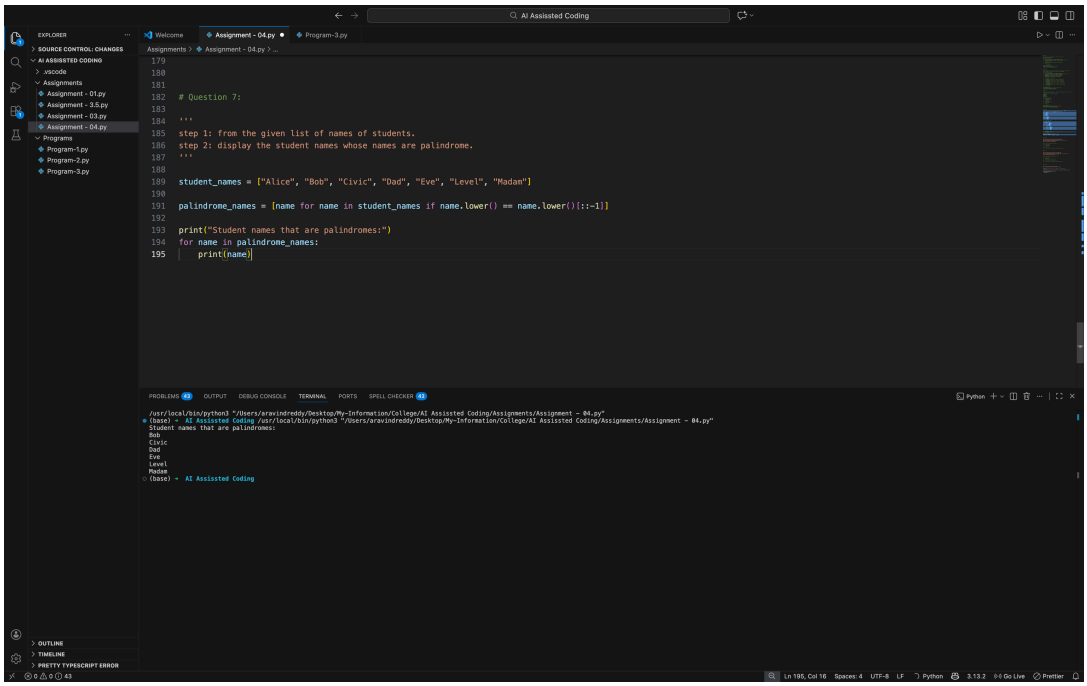
Q5. Result Analysis Based on Marks



Q6 Voting Eligibility Check (Chain-of-Thought Prompting)



Q7 Prompt Chaining (String Processing – Palindrome Names)



Q8 Prompt Chaining (String Processing – Word Length Analysis)