

## Question 1

### Step 1: Question

Create a Python function that converts an amount from one currency to another using exchange rates stored in a dictionary. Use GitHub Copilot along with VS Code. Use few shots prompting.

### Step 2: Prompt

"Write a Python function that converts an amount from one currency to another using a dictionary of exchange rates."

### Step 3: Code

Python

```
def convert_currency(amount, from_currency, to_currency, rates):
    if from_currency not in rates or to_currency not in rates:
        return None
    usd_amount = amount / rates[from_currency]
    return usd_amount * rates[to_currency]

# Example usage:
rates = {'USD': 1, 'EUR': 0.85, 'INR': 74}
result = convert_currency(100, 'USD', 'INR', rates)
print(result)
```

### Step 4: Output

Code

7400.0

## Question 2

### Step 1: Question

Write a Python program to extract all email addresses from a block of text using regular expressions. GitHub Copilot along with VS Code. Use zero shot prompting.

### Step 2: Prompt

"Extract all email addresses from a given text using Python regular expressions."

### Step 3: Code

● Python

```
import re

text = "Please contact us at support@example.com or sales@example.org for more information."
emails = re.findall(r'[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}', text)
print(emails)
```

### Step 4: Output

○ Code

```
['support@example.com', 'sales@example.org']
```

### Question 3

Step 1: Question Given a list of movies with their genres, write a Python function that recommends movies based on a user's preferred genre. Use the Cursor AI tool. Use few-shot prompting.

Step 2: Prompt" Few-shot examples:

- Input: movies= [('Inception', 'Sci-Fi'), ('Titanic', 'Romance')], preferred='Sci-Fi' → Output: ['Inception']
- Input: movies = [('Dune', 'Sci-Fi'), ('Up', 'Animation')], preferred='Animation' → Output: ['Up']

Task: Write a Python function that takes a list of (movie, genre) tuples and a preferred genre and returns a list of movies matching that genre.

"Step 3: Code

+

```
python

def recommend_movies(movies, preferred_genre):
    return [movie for movie, genre in movies if genre == preferred_genre]

# Example usage:
movies = [('Inception', 'Sci-Fi'), ('Titanic', 'Romance'), ('Interstellar', 'Sci-Fi'), ('The Notebook', 'Romance')]
recommended = recommend_movies(movies, 'Sci-Fi')
print(recommended)
```

Step 4: Output

≡ text

```
['Inception', 'Interstellar']
```

#### Question 4

Step 1: Question Write Python code that reads a CSV file containing student names and marks in 3 subjects. Calculate the total and average marks for each student. Use the Cursor AI tool.

Step 2: Prompt" Write Python code to read a CSV file with student names and marks for 3 subjects, then print each student's total and average marks.

Step 3: Code

```
import csv

with open('students.csv', 'r') as file:
    reader = csv.reader(file)
    header = next(reader)
    for row in reader:
        name = row[0]
        marks = list(map(int, row[1:]))
        total = sum(marks)
        average = total / len(marks)
        print(f"{name}: Total = {total}, Average = {average}")

# Example students.csv file:
# Name,Subject1,Subject2,Subject3
# Alice,85,90,95
```

Step 4: Output

```
Alice: Total = 270, Average = 90.0
Bob: Total = 248, Average = 82.66666666666667
```