**Q1**

def add\_matrices(matrix1, matrix2):

result = []

for i in range(len(matrix1)):

row = []

for j in range(len(matrix1[i])):

row.append(matrix1[i][j] + matrix2[i][j])

result.append(row)

return result

**Q2**

def multiply\_matrices(matrix1, matrix2):

result = []

for i in range(len(matrix1)):

row = []

for j in range(len(matrix2[0])):

cell\_value = 0

for k in range(len(matrix2)):

cell\_value += matrix1[i][k] \* matrix2[k][j]

row.append(cell\_value)

result.append(row)

return result

**Q3**

def transpose\_matrix(matrix):

rows = len(matrix)

cols = len(matrix[0])

# Create a new matrix with swapped rows and columns

transposed\_matrix = [[0 for \_ in range(rows)] for \_ in range(cols)]

for i in range(rows):

for j in range(cols):

transposed\_matrix[j][i] = matrix[i][j]

return transposed\_matrix

# Example matrix

matrix = [[1, 2, 3],

[4, 5, 6],

[7, 8, 9]]

# Transpose the matrix

transposed\_matrix = transpose\_matrix(matrix)

for row in transposed\_matrix:

print(row)

**Q4**

def sort\_words\_alph(words):

return sorted(words)

**Q5**

import string

def remove\_punctuation(input\_string):

# Create a translation table to remove punctuation

translator = str.maketrans("", "", string.punctuation)

# Use the translation table to remove punctuation

clean\_string = input\_string.translate(translator)

return clean\_string