>>> def calculate(numbers):

... if len(numbers) != 9:

... raise ValueError("List must contain nine numbers.")

...

... # Calculate statistics

... mean = sum(numbers) / 9

... variance = sum((x - mean) \*\* 2 for x in numbers) / 9

... std\_deviation = variance \*\* 0.5

... maximum = max(numbers)

... minimum = min(numbers)

... total\_sum = sum(numbers)

...

... # Reshape for axis-wise calculations

... axis1 = [numbers[i:i+3] for i in range(0, 9, 3)]

... axis2 = [[numbers[j] for j in range(i, 9, 3)] for i in range(3)]

...

... # Calculate statistics for axis1 and axis2

... mean\_axis1 = [sum(row) / 3 for row in axis1]

... mean\_axis2 = [sum(col) / 3 for col in axis2]

... variance\_axis1 = [sum((x - m) \*\* 2 for x in row) / 3 for row, m in zip(axis1, mean\_axis1)]

... variance\_axis2 = [sum((x - m) \*\* 2 for x in col) / 3 for col, m in zip(axis2, mean\_axis2)]

... std\_deviation\_axis1 = [v \*\* 0.5 for v in variance\_axis1]

... std\_deviation\_axis2 = [v \*\* 0.5 for v in variance\_axis2]

... max\_axis1 = [max(row) for row in axis1]

... max\_axis2 = [max(col) for col in axis2]

... min\_axis1 = [min(row) for row in axis1]

... min\_axis2 = [min(col) for col in axis2]

... sum\_axis1 = [sum(row) for row in axis1]

... sum\_axis2 = [sum(col) for col in axis2]

...

... return {

... 'mean': [mean\_axis1, mean\_axis2, mean],

... 'variance': [variance\_axis1, variance\_axis2, variance],

... 'standard deviation': [std\_deviation\_axis1, std\_deviation\_axis2, std\_deviation],

... 'max': [max\_axis1, max\_axis2, maximum],

... 'min': [min\_axis1, min\_axis2, minimum],

... 'sum': [sum\_axis1, sum\_axis2, total\_sum]

... }

...

>>> # Example usage:

>>> result = calculate([0, 1, 2, 3, 4, 5, 6, 7, 8])

>>> print(result)

{'mean': [[1.0, 4.0, 7.0], [3.0, 4.0, 5.0], 4.0], 'variance': [[0.6666666666666666, 0.6666666666666666, 0.6666666666666666], [6.0, 6.0, 6.0], 6.666666666666667], 'standard deviation': [[0.816496580927726, 0.816496580927726, 0.816496580927726], [2.449489742783178, 2.449489742783178, 2.449489742783178], 2.581988897471611], 'max': [[2, 5, 8], [6, 7, 8], 8], 'min': [[0, 3, 6], [0, 1, 2], 0], 'sum': [[3, 12, 21], [9, 12, 15], 36]}

>>>