**1. Basics of List Comprehension**

**Q1:** Write a list comprehension to generate a list of cubes from 1 to 10.  
**Q2:** Create a list of all characters in the string "DataScience" in lowercase using list comprehension.

**2. List Comprehension with Conditions**

**Q1:** Using list comprehension, create a list of numbers between 1 and 20 that are divisible by 3.  
**Q2:** For numbers 1 to 10, create a list that contains "Even" if the number is even, otherwise "Odd".

**3. Nested List Comprehension**

**Q1:** Given a 2D list matrix = [[2,4], [6,8], [10,12]], flatten it into a single list.  
**Q2:** Using nested list comprehension, create a 3x3 matrix of numbers from 1 to 9.

**4. Using Functions in List Comprehension**

**Q1:** Write a function double(x) that returns twice the number, and use it in a list comprehension for numbers 1–5.  
**Q2:** Write a list comprehension that uses a function is\_vowel(ch) to extract vowels from the string "education".

**5. Data Structures with Comprehensions**

**Q1:** Create a dictionary using dictionary comprehension where keys are numbers 1–5 and values are their cubes.  
**Q2:** From a list [1, 2, 2, 3, 4, 4, 5], create a set comprehension that keeps only unique even numbers.

**6. Default Parameters**

**Q1:** Define a function multiply(a, b=2) that multiplies two numbers. Call it with one and two arguments.  
**Q2:** Identify and fix the problem in this function:

def add\_value(value, data\_list=[]):

data\_list.append(value)

return data\_list

**7. Variable Arguments**

**Q1:** Write a function sum\_all(\*nums) that returns the sum of all numbers passed.  
**Q2:** Write a function display\_info(\*\*kwargs) that prints all key-value pairs passed as arguments.

**8. Specialized Sorts**

**Q1:** Given the list fruits = ['apple', 'kiwi', 'banana', 'cherry'], sort it based on string length.  
**Q2:** Given the list of tuples students = [('Amit', 85), ('Raj', 75), ('Anu', 85), ('Priya', 90)],  
sort it by marks and then by name.