List Comprehension:

1. Find all of the numbers from 1-1000 that are divisible by 7
2. Find all of the numbers from 1-1000 that have a 3 in them
3. Count the number of spaces in a string
4. Create a list of all the consonants in the string “Yellow Yaks like yelling and yawning and yesturday they yodled while eating yuky yams”
5. Get the index and the value as a tuple for items in the list “hi”, 4, 8.99, ‘apple’, (‘t,b’,’n’). Result would look like (index, value), (index, value)
6. Find the common numbers in two lists (without using a tuple or set) list\_a = 1, 2, 3, 4, list\_b = 2, 3, 4, 5
7. Get only the numbers in a sentence like ‘In 1984 there were 13 instances of a protest with over 1000 people attending’
8. Given numbers = range(20), produce a list containing the word ‘even’ if a number in the numbers is even, and the word ‘odd’ if the number is odd. Result would look like ‘odd’,’odd’, ‘even’
9. Produce a list of tuples consisting of only the matching numbers in these lists list\_a = 1, 2, 3,4,5,6,7,8,9, list\_b = 2, 7, 1, 12. Result would look like (4,4), (12,12)
10. Find all of the words in a string that are less than 4 letters
11. Use a nested list comprehension to find all of the numbers from 1-1000 that are divisible by any single digit besides 1 (2-9)
    1. Turn every item of a list into its square
    2. Concatenate two lists index-wise list1 = ["M", "na", "i", "Ke"]

list2 = ["y", "me", "s", "lly"]

Expected output: ['My', 'name', 'is', 'Kelly']

* 1. ['My', 'name', 'is', 'Kelly'] list1 = ["Hello ", "take "]

list2 = ["Dear", "Sir"]

**Expected output:** **['Hello Dear', 'Hello Sir', 'take Dear', 'take Sir']**

1. Extend nested list by adding the sublist

list1 = ["a", "b", ["c", ["d", "e", ["f", "g"], "k"], "l"], "m", "n"]

# sub list to add

sub\_list = ["h", "i", "j"]

Expected Output:

['a', 'b', ['c', ['d', 'e', ['f', 'g', 'h', 'i', 'j'], 'k'], 'l'], 'm', 'n']

1. Finding Transpose of a Matrix using List Comprehension matrix = [[1, 2], [3,4], [5,6], [7,8]] o/p: [[1, 3, 5, 7], [2, 4, 6, 8]]
2. Reverse each String in a Tuple