1. Python – Sort Dictionary key and values List

def sort\_dict\_keys\_and\_values(dictionary):

sorted\_keys = sorted(dictionary.keys())

sorted\_values = [dictionary[key] for key in sorted\_keys]

return sorted\_keys, sorted\_values

# Example usage

my\_dict = {'b': 2, 'a': 1, 'c': 3}

sorted\_keys, sorted\_values = sort\_dict\_keys\_and\_values(my\_dict)

print(sorted\_keys) # Output: ['a', 'b', 'c']

print(sorted\_values) # Output: [1, 2, 3]

1. Handling missing keys in Python dictionaries

from collections import defaultdict

def handle\_missing\_keys(dictionary, default\_value):

handled\_dict = defaultdict(lambda: default\_value, dictionary)

return handled\_dict

# Example usage

my\_dict = {'a': 1, 'b': 2}

handled\_dict = handle\_missing\_keys(my\_dict, 'Not Found')

print(handled\_dict['a']) # Output: 1

print(handled\_dict['c']) # Output: 'Not Found'

1. Python dictionary with keys having multiple inputs

from collections import defaultdict

def create\_dict\_with\_multiple\_inputs(inputs):

multi\_dict = defaultdict(list)

for key, value in inputs:

multi\_dict[key].append(value)

return multi\_dict

# Example usage

my\_inputs = [('a', 1), ('b', 2), ('a', 3)]

multi\_dict = create\_dict\_with\_multiple\_inputs(my\_inputs)

print(multi\_dict) # Output: {'a': [1, 3], 'b': [2]}

1. Print anagrams together in Python using List and Dictionary

def group\_anagrams(words):

anagram\_dict = {}

for word in words:

sorted\_word = ''.join(sorted(word))

if sorted\_word in anagram\_dict:

anagram\_dict[sorted\_word].append(word)

else:

anagram\_dict[sorted\_word] = [word]

return list(anagram\_dict.values())

# Example usage

my\_words = ['eat', 'tea', 'tan', 'ate', 'nat', 'bat']

anagram\_groups = group\_anagrams(my\_words)

print(anagram\_groups) # Output: [['eat', 'tea', 'ate'], ['tan', 'nat'], ['bat']]

1. K’th Non-repeating Character in Python using List Comprehension and OrderedDict
2. Check if binary representations of two numbers are anagram
3. Python Counter to find the size of largest subset of anagram words
4. Python | Remove all duplicates words from a given sentence
5. Python Dictionary to find mirror characters in a string
6. Counting the frequencies in a list using dictionary in Python