Here are some Python programming practice questions specifically focused on dictionaries:

**Basic Dictionary Operations**

1. **Create a Dictionary**: Write a Python program to create a dictionary with keys as numbers from 1 to 5 and values as their squares.
   * Example: {1: 1, 2: 4, 3: 9, 4: 16, 5: 25}
2. **Accessing Dictionary Elements**: Write a Python program to access the value associated with a specific key in a dictionary. If the key doesn’t exist, print an appropriate message.
3. **Updating a Dictionary**: Write a Python program to add a new key-value pair to an existing dictionary and update the value of an existing key.
4. **Delete a Key from a Dictionary**: Write a Python program to remove a key from a dictionary and handle the case where the key doesn’t exist.
5. **Check for a Key in a Dictionary**: Write a Python program to check if a given key exists in a dictionary. Return True if it exists, otherwise False.

**Dictionary Methods and Functions**

1. **Iterating Through a Dictionary**: Write a Python program to iterate through a dictionary, printing each key-value pair.
2. **Dictionary Length**: Write a Python program to find the number of key-value pairs in a dictionary.
3. **Merge Two Dictionaries**: Write a Python program to merge two dictionaries into one.
4. **Maximum and Minimum Values in a Dictionary**: Write a Python program to find the key with the maximum value and the key with the minimum value in a dictionary.
5. **Sorting a Dictionary by Keys**: Write a Python program to sort a dictionary by its keys.

**Advanced Dictionary Operations**

1. **Dictionary Comprehension**: Write a Python program to create a dictionary from two lists where one list provides the keys and the other provides the values.
   * Example: Given keys = ['a', 'b', 'c'] and values = [1, 2, 3], create { 'a': 1, 'b': 2, 'c': 3 }.
2. **Nested Dictionaries**: Write a Python program to create a nested dictionary where the outer keys represent categories and the inner dictionaries represent items and their prices.
   * Example: { "Fruits": {"Apple": 1.2, "Banana": 0.5}, "Vegetables": {"Carrot": 0.7, "Spinach": 1.0} }
3. **Dictionary from Two Lists Using Zip**: Write a Python program to create a dictionary from two lists using the zip() function.
   * Example: Given keys = ['name', 'age'] and values = ['Alice', 25], create { 'name': 'Alice', 'age': 25 }.
4. **Counting Frequency of Elements in a List Using a Dictionary**: Write a Python program that takes a list of items and counts the frequency of each item using a dictionary.
   * Example: For the list ['apple', 'banana', 'apple', 'orange', 'banana', 'apple'], output should be { 'apple': 3, 'banana': 2, 'orange': 1 }.
5. **Invert a Dictionary**: Write a Python program to invert a dictionary, i.e., swap the keys and values.
   * Example: For { 'a': 1, 'b': 2, 'c': 3 }, output should be { 1: 'a', 2: 'b', 3: 'c' }.
6. **Sum of Dictionary Values**: Write a Python program to calculate the sum of all the values in a dictionary.
7. **Filtering Dictionary Based on Values**: Write a Python program to filter a dictionary based on values, keeping only those key-value pairs where the value is greater than a specified threshold.
8. **Group Elements by Frequency**: Write a Python program to group elements of a list based on their frequency in a dictionary.
   * Example: For the list [1, 2, 2, 3, 3, 3], output should be { 1: [1], 2: [2, 2], 3: [3, 3, 3] }.
9. **Find Common Keys in Two Dictionaries**: Write a Python program to find the keys that are common in two dictionaries.
10. **Dictionary with Default Values**: Write a Python program to create a dictionary with keys from a list and default values for all keys.

These questions will help you practice and solidify your understanding of working with dictionaries in Python.