**1. What does an empty dictionary's code look like?**

**Ans.** empty dictionary will look like as below the variable empty\_dict is assigned an empty dictionary. We can also create an empty dictionary using the dict() constructor:

a. empty\_dict = {}

b. empty\_dict=dict()

**2. What is the value of a dictionary value with the key 'foo' and the value 42?**

Ans. In a dictionary, values are associated with keys. Keys provide a way to uniquely identify and access their corresponding values. In this case, the key 'foo' is associated with the value 42. So, when we can access the value using the key 'foo', you will get the value 42.

**3. What is the most significant distinction between a dictionary and a list?**

Ans. A dictionary, also known as a hash table or associative array, is an unordered collection of key-value pairs. Each key in a dictionary is unique and is used to access its corresponding value. The keys in a dictionary are typically strings or numbers, but they can be of any immutable type. Dictionaries provide fast access to values based on their keys, making it efficient to retrieve values by specifying the associated key.

On the other hand, a list is an ordered collection of items. It is an ordered sequence where each element has a specific position or index. Lists can store elements of different types, and duplicate values are allowed. The position of an element in a list determines its retrieval, modification, or removal. Lists are useful when you need to maintain the order of elements or when you want to access them by their position.

**4. What happens if you try to access spam['foo'] if spam is {'bar': 100}?**

Ans. If you try to access spam['foo'] where spam is {'bar': 100}, you would encounter a KeyError.

**5. If a dictionary is stored in spam, what is the difference between the expressions 'cat' in spam and 'cat' in spam.keys()?**

Ans**. 'cat' in spam**: This expression checks if the key 'cat' exists in the dictionary spam. It evaluates to a boolean value, True or False, depending on whether the key is present or not. If 'cat' is a key in spam, the expression will return True; otherwise, it will return False.

**'cat' in spam.keys():** This expression checks if the key 'cat' exists in the keys of the dictionary spam. The spam.keys() method returns a view object that represents the keys in the dictionary. By using 'cat' in spam.keys(), you are checking if 'cat' is present among the keys specifically. It also evaluates to a boolean value, True or False, depending on the existence of the key in the dictionary.

**6. If a dictionary is stored in spam, what is the difference between the expressions 'cat' in spam and 'cat' in spam.values() ?**

Ans. 'cat' in spam checks if 'cat' is a key in the dictionary spam.

'cat' in spam.values() checks if 'cat' is a value in the dictionary spam.

**7. What is a shortcut for the following code?**

**if 'color' not in spam:**

**spam['color'] = 'black'**

Ans. A shortcut for the given code can be achieved using the **dict.setdefault()** method. The setdefault() method allows you to set a default value for a key in a dictionary only if the key does not already exist. Here's the equivalent shortcut code

spam.setdefault('color', 'black')

In this case, if the key 'color' is not present in the spam dictionary, the setdefault() method will set the value 'black' for that key. However, if the key 'color' already exists in the dictionary, the method will not modify its value.

**8. How do you "pretty print" dictionary values using which module and function?**

Ans. To "pretty print" dictionary values in Python, we can use the pprint module and its pprint() function. The pprint module provides a way to format and display complex data structures, such as dictionaries, in a more readable and visually appealing manner. Lets see with code

import pprint

my\_dict = {'key1': 'value1', 'key2': 'value2', 'key3': 'value3'}

pprint.pprint(my\_dict)

**output:-** {'key1': 'value1',

'key2': 'value2',

'key3': 'value3'}

the pprint.pprint() function formats the dictionary by printing each key-value pair on a separate line with appropriate indentation.