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

Ans. empty\_dict = {}

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

Ans. my\_dict = {'foo': 42}

value = my\_dict['foo']

print(value)

# Output: 42

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

Ans. The most significant distinction between a dictionary and a list is their underlying data structure and how they store and organize data:

Data Structure:

A list is an ordered collection of elements, where each element is identified by its position or index. Lists maintain the order of elements and allow for duplicates.

A dictionary is an unordered collection of key-value pairs, where each key is unique and maps to a corresponding value. Dictionaries do not maintain the order of elements, and they provide fast lookups based on keys.

Accessing Elements:

In a list, elements are accessed by their index. You can retrieve elements by specifying the index position, and the elements are ordered based on their positions.

In a dictionary, elements (values) are accessed by their keys. You specify the key to retrieve the corresponding value, and dictionaries provide fast lookups based on keys.

Indexing:

Lists support integer-based indexing, where each element has an integer index starting from 0.

Dictionaries support key-based indexing, where each value is associated with a unique key.

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

Ans. spam = {'bar': 100}

value = spam['foo'] # This will raise 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. The expressions 'cat' in spam and 'cat' in spam.keys() both check whether the key 'cat' exists in the dictionary spam, but they differ in their approach and efficiency:

'cat' in spam:

This expression directly checks whether the key 'cat' exists in the dictionary spam.

It searches for the key 'cat' within the keys of the dictionary.

If the key 'cat' is found in the dictionary spam, the expression evaluates to True; otherwise, it evaluates to False.

This approach is more efficient because it directly checks for the presence of the key without creating an intermediate list of keys.

'cat' in spam.keys():

This expression retrieves all the keys of the dictionary spam using the keys() method and then checks whether the key 'cat' exists in the list of keys.

It first creates a list of all keys in the dictionary and then performs a membership check to see if 'cat' is in that list.

If the key 'cat' is found in the list of keys, the expression evaluates to True; otherwise, it evaluates to False.

This approach is less efficient because it involves creating an intermediate list of keys, which consumes additional memory and processing time, especially for large dictionaries.

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

Ans. The expressions 'cat' in spam and 'cat' in spam.values() both check for the presence of the value 'cat' within the values of the dictionary spam, but they differ in their approach and functionality:

'cat' in spam:

This expression checks whether the key 'cat' exists in the dictionary spam.

It searches for the key 'cat' within the keys of the dictionary.

If the key 'cat' is found in the dictionary spam, regardless of its associated value, the expression evaluates to True; otherwise, it evaluates to False.

This expression is used to determine whether a specific key exists in the dictionary, irrespective of its associated value.

'cat' in spam.values():

This expression checks whether the value 'cat' exists within the values of the dictionary spam.

It searches for the value 'cat' within all the values of the dictionary.

If the value 'cat' is found in any of the values of the dictionary spam, the expression evaluates to True; otherwise, it evaluates to False.

This expression is used to determine whether a specific value exists in the dictionary, regardless of its associated key.

7. What is a shortcut for the following code?

if 'color' not in spam:

spam['color'] = 'black'

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

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

Ans.

import pprint

# Example dictionary

my\_dict = {'name': 'John', 'age': 30, 'city': 'New York'}

# Pretty print the dictionary

pprint.pprint(my\_dict)