Assingnement-5

1-What does an empty dictionaries code look like?

In Python, an empty dictionary is represented by curly braces {}

This creates a variable named empty\_dict that refers to an empty dictionary. You can later add key-value pairs to this dictionary using the syntax (empty\_dict[key] = value)

Code –empty\_dict[key] = value

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

a-my\_number={‘foo’, : 42}

value\_of\_foo=my\_number[‘foo’]

print(value\_of\_foo)

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

a- A list is an ordered collection of items, where as a dictionary is an unordered data collection in a key. Value pair. Elements from list can be accessed using keys. Defined using square brackets **[]**

4-What happens if you try to access spam foo if spam is {‘bar’ :100}?

a-spam = {'bar': 100}

value = spam['foo']

if 'foo' in spam:

value = spam['foo']

else:

value = None #Or any default value you prefer

value = spam.get('foo', None) # None is the default value if 'foo' is not present

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

1. The expressions 'cat' in spam and 'cat' in spam.values**()** both check for the presence of the value 'cat' in the values of the dictionary spam, but they operate on different parts of the dictionary:

spam = {'a': 'cat', 'b': 'dog', 'c': 'fish'}

# The first expression checks for the key 'cat' among the keys

print('cat' in spam) # False

print('cat' in spam.values() # True

7- A shortcut for the given code can be achieved using the **setdefault** method of dictionaries. The setdefault method sets the value of a key in the dictionary if the key is not already present, and it returns the value associated with the key.

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

8- How do you “quot;pretty print&quot” dictionary values using which module and function?

from pprint import pprint

# Your dictionary

my\_dict = {'name': 'John', 'age': 25, 'city': 'New York'}

# Pretty print the dictionary

pprint(my\_dict)

from pprint import pprint

my\_dict = {'name': 'John', 'age': 25, 'city': 'New York'}

# Pretty print the dictionary to a file

with open('output.txt', 'w') as file:

pprint(my\_dict, file=file)