Dictionary

1. Create an empty dictionary:

my\_dict = {}

1. Add a key-value pair to a dictionary:

|  |
| --- |
| d={100:"kabin", 200: "pragyan", 300:"nikita",400:"dipesh"}  d[500]= 'bhaskar'  print(d)  output:  {100: 'kabin', 200: 'pragyan', 300: 'nikita', 400: 'dipesh', 500: 'bhaskar'} |

1. Access a value in a dictionary using its key:

|  |
| --- |
| d={100:"kabin", 200: "pragyan", 300:"nikita",400:"dipesh"}  d[500]= 'bhaskar'  value= d[100]  print(value)  output:  kabin |

1. Update the value associated with a specific key in a dictionary:

d[100] =’ayush’

1. How do you remove a key-value pair from a dictionary?

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

del my\_dict['b'] # Removes the key 'b'

1. How do you check if a key exists in a dictionary?

if 'a' in my\_dict:

print("Key exists")

1. How do you get the number of key-value pairs in a dictionary?

The len() function is used to get the number of key-value pairs.

num\_pairs = len(my\_dict)

1. How do you get all the keys from a dictionary?

keys = my\_dict.keys()

1. How do you get all the values from a dictionary?

values = my\_dict.values()

1. How do you convert two lists (keys and values) into a dictionary?

By using the zip() function with dict() to combine two lists into a dictionary.

|  |
| --- |
| keys = ['a', 'b', 'c']  values = [1, 2, 3]  my\_dict = dict(zip(keys, values)) |

1. How do you iterate over the key-value pairs in a dictionary?

|  |
| --- |
| my\_dict= {100:"kabin", 200: "pragyan", 300:"nikita",400:"dipesh"}  for key, value in my\_dict.items():      print(key, value)  output:  100 kabin  200 pragyan  300 nikita  400 dipesh |

1. How do you merge two dictionaries?

We can merge two dictionaries using the update().

|  |
| --- |
| dict1 = {'a': 1, 'b': 2}  dict2 = {'c': 3, 'b': 4}  dict1.update(dict2) |

1. How do you remove a key from a dictionary and return its value?

Using pop() method we can remove the key and return its value.

value = my\_dict.pop('a') # Removes 'a' and returns its value

1. How do you sort a dictionary by its keys?

sorted\_dict = dict(sorted(my\_dict.items()))

1. How do you sort a dictionary by its values?

sorted\_dict= dict(sorted(my\_dict.items(), key=lambda item: item[1]))

1. How do you get the value for a key without raising an error if the key doesn't exist?

We can use get() method, which returns None or a default value if the key is not found.

value = my\_dict.get('key', 'default\_value')

1. How do you reverse the key-value pairs in a dictionary?

reversed\_dict = {value: key for key, value in my\_dict.items()}

1. How do you find the key with the maximum value in a dictionary?

We can use the max() function with the key parameter to get the key with the highest value.

max\_key = max(my\_dict, key=my\_dict.get)

1. How do you check if two dictionaries are equal?

if dict1 == dict2:

print("Dictionaries are equal")

1. How do you get a dictionary’s items as a list of tuples?

By using the items() method we can get the dictionary items as a list of tuples

items = list(my\_dict.items())

1. How do you create a dictionary with default values for missing keys?

|  |
| --- |
| d={100:"Bhaskar",200:"Dipesh",300:"Isha",400:"Kabin"}  (d.setdefault(100,"Angry"))  print(d)  d={100:"Bhaskar",200:"Dipesh",300:"Isha",400:"Kabin",}  (d.setdefault(600,"Angry"))  print(d)  output:  {100: 'Bhaskar', 200: 'Dipesh', 300: 'Isha', 400: 'Kabin'}  {100: 'Bhaskar', 200: 'Dipesh', 300: 'Isha', 400: 'Kabin', 600: 'Angry'} |

1. How do you count the occurrences of each element in a list using a dictionary?

|  |
| --- |
| elements = [1, 2, 2, 3, 3, 3,3]  count\_dict = {}  for elem in elements:      count\_dict[elem] = count\_dict.get(elem, 0) + 1  print(count\_dict)  output:  {1: 1, 2: 2, 3: 4} |

1. How do you group a list of items by a certain property using a dictionary?

|  |
| --- |
| list=["Bhaskar", "Biplov","Ankit","Ankrit","Bhuwan","Ayush"]  d={}  for words in list:      first\_letter=words[0]      if first\_letter in d:        d[first\_letter].append(words)      else:        d[first\_letter]=[]        d[first\_letter].append(words)  print(d)  output:  {'B': ['Bhaskar', 'Biplov', 'Bhuwan'], 'A': ['Ankit', 'Ankrit', 'Ayush']} |

1. How do you remove multiple keys from a dictionary at once?

keys\_to\_remove = ['a', 'b']

my\_dict = {k: v for k, v in my\_dict.items() if k not in keys\_to\_remove}

1. How do you combine two dictionaries, summing values for common keys?

|  |
| --- |
| d1={100:"Bhaskar",200:"Dipesh",300:"Isha",400:"Kabin"}  d2={500: 'Prakash', 900: 'Hira', 300: 'Pant', 200: 'buddha'}  common\_dict={}  for k,v in d1.items():     common\_dict[k]=v  for k,v in d2.items():      if k in common\_dict:         common\_dict[k]+=v      else:         common\_dict[k]=v  print(common\_dict)  output:  {100: 'Bhaskar', 200: 'Dipeshbuddha', 300: 'IshaPant', 400: 'Kabin', 500: 'Prakash', 900: 'Hira'} |

1. How do you use a dictionary to simulate a switch-case statement?

|  |
| --- |
| def case1(): print("Case 1")  def case2(): print("Case 2")  switch = {      'a': case1,      'b': case2  }  switch.get('a', lambda: None)()  ouput:  Case 1 |

1. How do you nest dictionaries and access deeply nested values?

|  |
| --- |
| nested\_dict = {'a': {'b': {'c': 1}}}  value = nested\_dict['a']['b']['c']  print(value)  output:  1 |

1. How do you invert a dictionary, i.e., swap keys and values?

inverted\_dict = {v: k for k, v in my\_dict.items()}

1. How do you find the intersection of two dictionaries (keys that exist in both)?

common\_keys = dict1.keys() & dict2.keys()

1. How do you handle mutable objects as dictionary keys, and what problems can it cause?

|  |
| --- |
| #converting it into tuples  l1=[2,4]  l2=[1,3]  l3=[6,8]  l4=[5,7]  # d={l1:"First two even",l2:"First two odd",l3:"Second two even",l4:"Second two odd"}  # print(d)    d={tuple(l1):"First two even",tuple(l2):"First two odd",tuple(l3):"Second two even",tuple(l4):"Second two odd"}  print(d)  output:  {(2, 4): 'First two even', (1, 3): 'First two odd', (6, 8): 'Second two even', (5, 7): 'Second two odd'} |