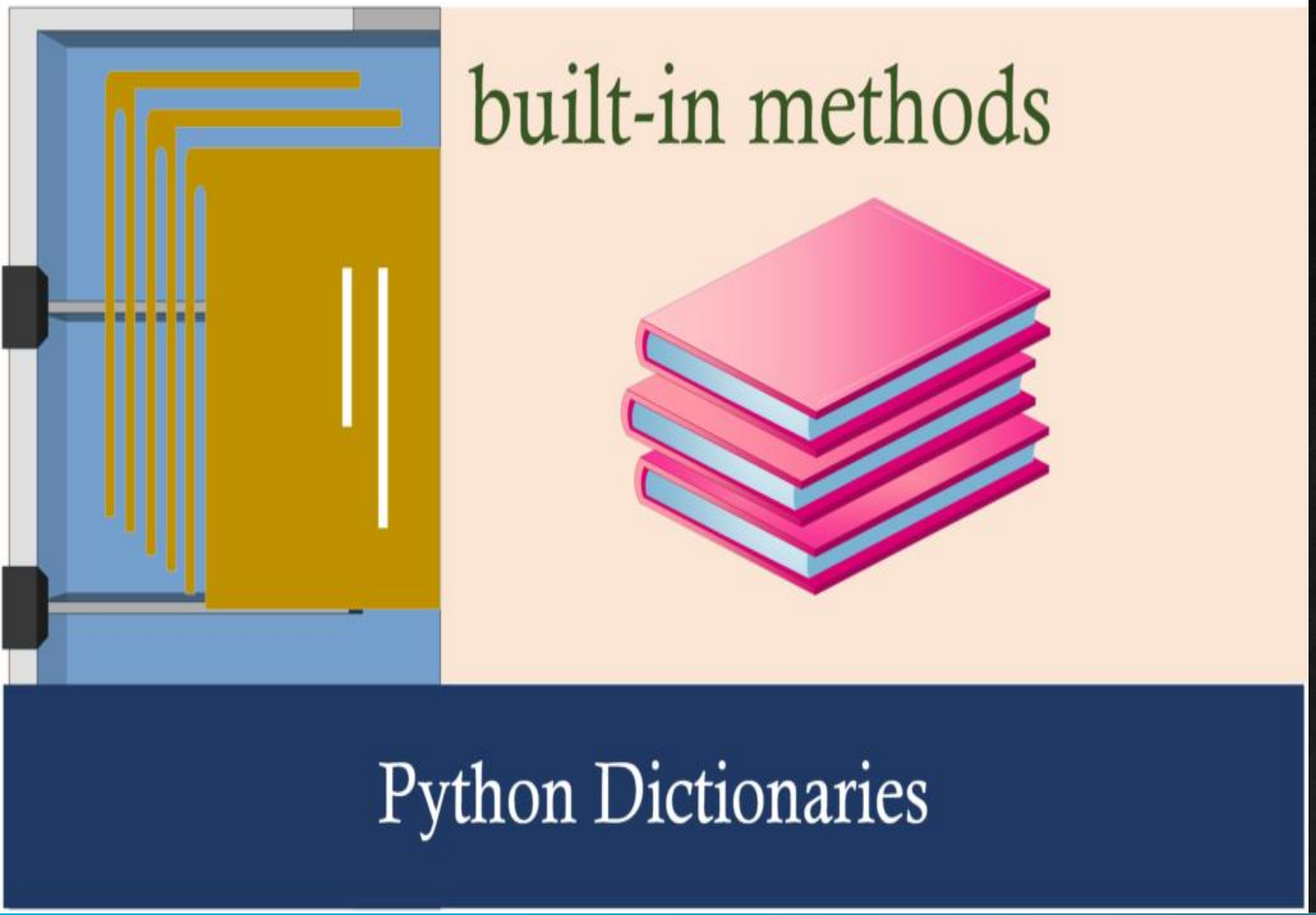


Dictionary

Dictionaries in Python

Agenda

What is a Dictionary in Python?

What are the key-value pairs?

Operations on Dictionaries

Important things about Dictionaries

Dictionary Methods

Problems for Practice

What is a Dictionary in python?

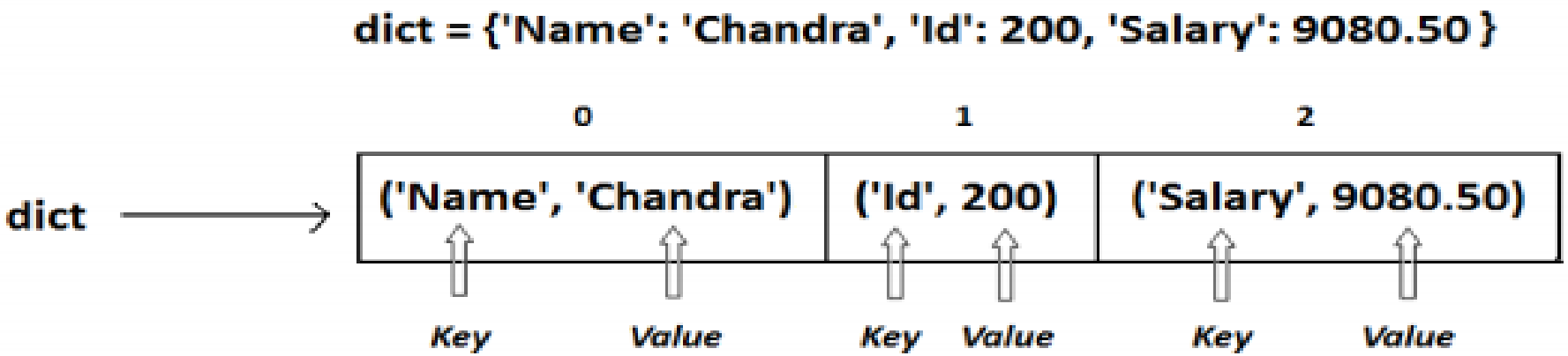
- dictionary represents a group of elements arranged in the form of key-value pairs. In the dictionary, the first element is considered as 'key' and the immediate next element is taken as its 'value'. The key and its value are separated by a colon (:). All the key-value pairs in a dictionary are inserted in curly braces {}.

  eg:

  ```
  dict = {"name" : "Alice", "age" : 23, "gender" : "female", "height" : 5.5}
  print(dict)

  Output : {'name': 'Alice', 'age': 23, 'gender': 'female', 'height': 5.5}
  ```

What are the key-value pairs?



When the 'key' is provided, we can get back its 'value'. This is how we search for the values in a dictionary. For example, 'Name' is the key. To get its value, i.e. 'Chandra', we should mention the key as an index to the dictionary, as: dict['Name']. This will return the value 'Chandra'. Similarly, dict['Id'] returns its value, i.e. 200.

Operations on Dictionaries

- To access the elements of a dictionary, we should not use indexing or slicing. For example, dict[0] or dict[1:3] etc. expressions will give error. To access the value associated with a key, we can mention the key name inside the square braces, as: dict['Name']. This will return the value associated with 'Name'. This is nothing but 'Chandra'.
- If you want to know how many key-value pairs are in the dictionary, you can use len() function as shown in the following statements.

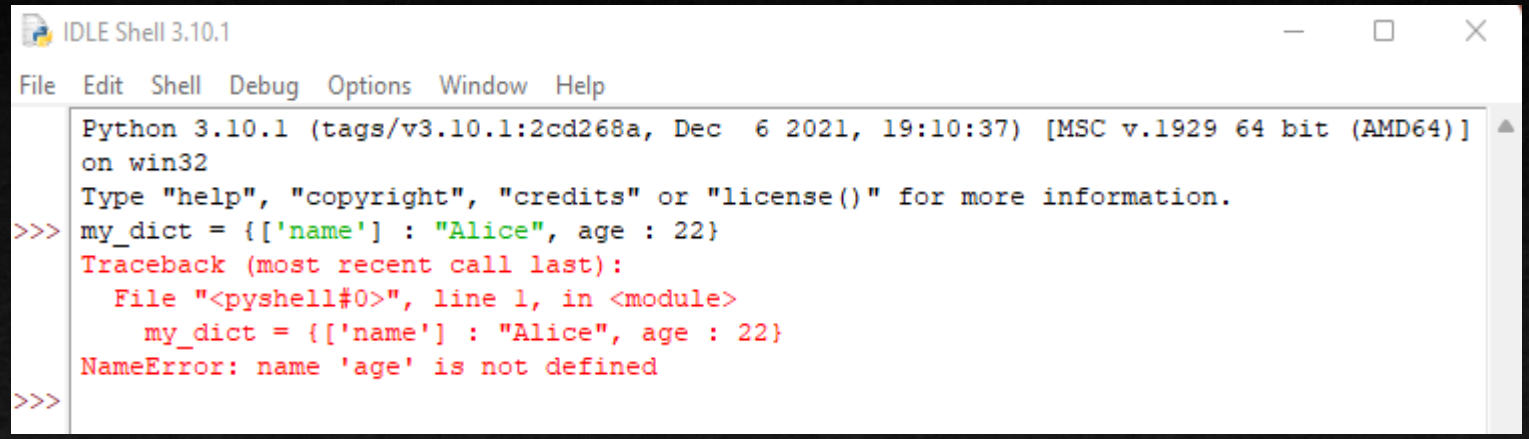
  dict = {"name" : "Alice", "age" : 25, "gender" : "female"}

  print(len(dict))

  Output : 3
- We can use del keyword to delete a dictionary completely.
- To check if a key is present in the dictionary, we can use 'in' and 'not in'.
- Question : What if a dictionary has duplicate keys? Find it out yourself and observe the output carefully.

Important things to know about Dictionaries

- Keys should be immutable type. For example, we can use a number, string or tuples as keys since they are immutable. We cannot use lists or dictionaries as keys. If they are used as keys, we will get 'TypeError'. Consider the following example :



- Indexing and slicing are not useful to access the elements of a dictionary.
- The keys of a dictionary should be unique and belong to immutable datatype. The value can be immutable or mutable.

Dictionary Methods

Method	Example	Description
clear()	d.clear()	Removes all key-value pairs from dictionary ‘d’.
copy()	d1 = d.copy()	Copies all elements from ‘d’ into a new dictionary ‘d1’.
fromkeys()	d.fromkeys(s, [v])	Create a new dictionary with keys from sequence ‘s’ and values all set to ‘v’.
get()	d.get(k, [v])	Returns the value associated with key ‘k’. If key is not found, it returns ‘v’.
item()	d.items()	Returns an object that contains key-value pairs of ‘d’. The pairs are stored as tuples in the object.
keys()	d.key()	Returns a sequence of keys from the dictionary ‘d’.
values()	d.values()	Returns a sequence of values from the dictionary ‘d’.

Dictionary Methods

Method	Example	Description
update()	d.update(x)	Adds all elements from dictionary ‘x’ to ‘d’.
pop()	d.pop(k, [v])	Removes the key ‘k’ and its value from ‘d’ and returns the value. If key is not found, then the value ‘v’ is returned. If key is not found and ‘v’ is not mentioned then ‘KeyError’ is raised.
setdefault()	d.setdefault(k, [v])	[,v]) If key ‘k’ is found, its value is returned. If key is not found, then the k, v pair is stored into the dictionary ‘d’.

Problems for Practice

1) A Python program to create a dictionary with employee details and retrieve the values upon giving the keys.
2) A Python program to retrieve keys, values and key-value pairs from a dictionary.
3) A Python program to create a dictionary and find the sum of values.
4) A Python program to create a dictionary from keyboard and display the elements.
5) A Python program to create a dictionary with cricket players names and scores in a match. Also we are retrieving runs by entering the player's name.
6) A Python program to show the usage of for loop to retrieve elements of dictionaries.
7) A Python program to find the number of occurrences of each letter in a string using dictionary.

Problems for Practice

8) A Python program to sort the elements of a dictionary based on a key or value.
9) A Python program to delete a specific item from a dictionary.
10) A Python program to convert the elements of two lists into key-value pairs of a dictionary.
11) A Python program to convert a string into key-value pairs and store them into a dictionary.
12) A Python function to accept a dictionary and display its elements.
13) A Python program to create a dictionary that does not change the order of elements.

Points to remember about Dictionaries

- A dictionary represents a group of elements arranged in the form of key-value pairs. In the dictionary, the first element is considered as ‘key’ and the immediate next element is taken as its ‘value’.
- The key and value pairs should be written inside a dictionary by separating them with a colon (:). Each pair should be separated by a comma. All the key-value pairs of the dictionary should be written inside curly braces {}.
- While inserting a new element or modifying the existing element, we can use the format: dict[key] = value.
- The get(k, v) method returns the value upon taking the key ‘k’. If the key is not found in the dictionary, then it will return a default value ‘v’.
- The update({k:v}) method stores the key ‘k’ and its value ‘v’ pair into an existing dictionary.

Points to remember about Dictionaries

- The dict() method converts a list or tuple or zip object into a dictionary.
- The zip() method is useful to convert the sequences like lists into a zip class object.
- An ordered dictionary is a dictionary but it will keep the order of the elements.
- Ordered dictionaries are created using the OrderedDict() method of collections module.

References

- Dr. R. Nageswara Rao. Core Python Programming, 2ed (p. 332). Wiley India.
- https://www.w3schools.com/python/python_dictionaries.asp
- https://www.tutorialspoint.com/python3/python_dictionary.htm

Thank you