

Python Programming

Presented by

Mariena A. A.

**Asst. Professor and Head, Department of Computer Science
Little Flower College, Guruvayoor.**

Dictionary

- We can say that a dictionary is the collection of key-value pairs where the value can be any python object whereas the keys are the immutable python object, i.e., Numbers, string or tuple.
- The collections of the key-value pairs are enclosed within the curly braces {}.

```
Employee = {"Name": "John", "Age": 29, "salary":25000}
```

```
print(type(Employee))
```

```
print("printing Employee data .... ")
```

```
print(Employee)
```

Example

The values from the dictionary can be accessed by using the keys as keys are unique in the dictionary. `get()` method also used to access the dictionary values.

```
Employee = {"Name": "John", "Age": 29, "salary":25000}
```

```
print("printing Employee data .... ")
```

```
print("Name : %s" %Employee["Name"])
```

```
print("Age : %d" %Employee["Age"])
```

```
print("Salary : %d" %Employee["salary"])
```

Creation of Dictionary

Empty dictionary can be created as `dict={ }`

```
dict={key1:value1,key2:value2,key3:value3 }
```

To convert into dictionary

```
dict([(key1,value1),(key2,value2),(key3,value3)])
```

Dictionary operations

dictionary is a mutable data type, and its values can be updated by using the specific keys.

```
Employee = {"Name": "John", "Age": 29, "salary":25000}
```

```
print("printing Employee data .... ")
```

```
print(Employee)
```

Addition of elements

```
print("Enter the details of the new employee....")
```

```
Employee["Name"] = input("Name: ")
```

```
Employee["Age"] = int(input("Age: "))
```

```
Employee["salary"] = int(input("Salary: "))
```

```
Employee["dob"]=(input("Dob"))
```

```
print("printing the new data");
```

```
print(Employee)
```

Deleting element using del keyword

```
del dict[key] #deletes the given key value pair
del dict      # delete the dict from memory
dict.clear()  #clears all key value pairs from dict
```

```
Employee = {"Name": "John", "Age": 29, "salary":25000,"Company":
GOOGLE"}
```

```
print("Deleting some of the employee data")
del Employee["Name"]
del Employee["Company"]
del Employee
Employee.clear() # output: Employee= { }
```

Modification

```
Employee["salary"]=50000
```

ITERATION THROUGH FOR LOOP

A dictionary can be iterated using the for loop as given below.

To print keys only

```
Employee = {"Name": "John", "Age": 29, "salary":25000,"Company":"GOOGLE"}
}
```

```
for x in Employee:
```

```
    print(x);
```

o/p: name, company, age, salary

To print values

```
Employee = {"Name": "John", "Age": 29, "salary":25000,"Company":"GOOGLE"}
}
```

```
for x in Employee:
```

```
    print(Employee[x]);
```

Using values() method

```
Employee = {"Name": "John", "Age": 29, "salary":25000,"Company":"GOOGLE"}
}
```

```
for x in Employee.values():
```

```
    print(x);
```

Print values using items() method

```
Employee = {"Name": "John", "Age": 29, "salary":25000,"Company":"GOOGLE"}
}
```

```
for x in Employee.items():
```

```
    print(x);    Name: john Age: 29 Salary: 25000 Company: GOOGLE
```

Dictionary Rules

In the dictionary, we can not store multiple values for the same keys.

If we pass more than one values for a single key, then the value which is last assigned is considered as the value of the key.

```
Employee = {"Name": "John", "Age": 29, "Salary":25000,"Company":"GOOGLE",  
Name:"Johnn" }
```

```
for x,y in Employee.items():
```

```
    print(x,y)
```

In python, the key cannot be any mutable object. We can use numbers, strings, or tuple as the key but we can not use any mutable object like the list as the key in the dictionary.

```
Employee = {"Name": "John", "Age": 29, "salary":25000,"Company":"GOOGLE",  
100,201,301]:"Department ID" }
```

```
for x,y in Employee.items():
```

```
    print(x,y)
```

- o/p: ERROR

Dictionary built in function

<code>cmp(dict1, dict2)</code>	It compares the items of both the dictionary and returns true if the first dictionary values are greater than the second dictionary, otherwise it returns false.	
2	<code>len(dict)</code>	It is used to calculate the length of the dictionary.
3	<code>str(dict)</code>	It converts the dictionary into the printable string representation.
4	<code>type(dict)</code>	It is used to print the type of the passed variable.

Dictionary built in function

<code>a= d.copy()</code>		No memory allocated a will reference to the d dictionary
2. <code>d.clear()</code>		It is used to clear all items and produce empty dictionary.
3 <code>d.items()</code>		It displays key value pairs in the range.
4 <code>d.keys()</code>		It is used to display all keys
5. <code>d.values()</code>		It is used to display the values in the dictionary
6. <code>d.update()</code>		Add one or more items in the existing dictionary

Python

Functions, built-in functions, mathematical functions, date time functions, random numbers, writing user defined functions, composition of functions, parameter and arguments, default parameters, function calls, return statement, using global variables, recursion.

Hafna ---Function

Sreevidya----Built in function

Mathematical function—Sweaba

Safwana PM

9:37 AM

24.Safwana pm

Lena Sunil

9:37 AM

19 Lena Sunil P

josiya jose

9:37 AM

18 Josiya Jose

roniga antony

9:37 AM

Roniga Antony 5

Sweaba Seaban

Thank you!

