TOPSTechnologies

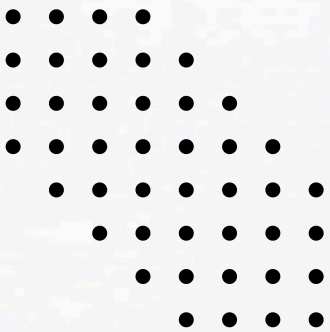
Dictionaries

Presented for :

TOPs Technologies

Presented by :

Sumit B Yadav



Python - Collections , Module, Functions

Que 1

Introduction to Dictionaries -

A Dictionary in python is a collection of key value pairs.
Key : A unique identifier used to access a value.
Value : The data associated with a key.

Key Features -
Unordered: The order of items is not guaranteed.
Mutable: You can change, add or remove key value pairs.
Unique Key: Key must be unique within a dictionary.
Syntax -
my_dic = {"key1": "value1", "key2": "value2"}

Example -
person = {
"name" : "Sumit",
"age" : 21,
"city" : "New York"
};
print(person["name"]
output: Sumit

Que 2

2. Accessing, adding, updating and deleting dictionary elements

Accessing dictionary elements -

1. Accessing elements using keys.

example-

person={"name" : "Sumit",
"age" : 21,
"city" : "New York"
};

print(person["city"])
output = New York

2. Adding New Elements:

add a new key-value pair by simply assigning a value to a new key.

person["email"] = "sy@gmail.com"
print(person)
output = {"name" : "Sumit" , "age": 20, "city": "New York", "email": "sy@gmail.com"}

Updating Elements:

Update the value of an existing key by assigning a new value.

person ["age"] = 32 #updating the age value.
print(person)
output = {"name" : "Sumit" , "age": 32, "city": "New York", "email": "sy@gmail.com"}

3. Deleting Elements:

Using del: Removes a key value pair.

```
del person["city"]
print(person)
```

output: {'name': 'Sumit', 'age': 32, 'email': 'sy@gmail.com'}

Using pop(): Removes the key-value pair and returns the value.

```
remove_email  person.pop("email")
print(removed_email)
print(person)
```

output: {'name': 'Sumit', 'age': 32}

Using clear(): Removes all key value pair from the dictionary.

```
person.clear()
print(person)
```

Que 3

3. Dictionary methods like keys(), value(), and items()

keys method-
returns a view object that displays all keys in the dictionary.

```
person = {
'name' : 'Sumit',
'age' : 30,
'city' : 'New York',
}

keys = person.keys()
print(keys)
output : dict_values(['name','age','city'])
```

values() method:
Returns a view object that displays all the values in the dictionary.

```
values = person.values()
print(values)
output : dict_values(['Sumit',30,'New York'])
```

items() method:
Returns a view object that displays all key-value pairs as tuples.

```
items = person.items()
print(items)
output = ([('name':'Sumit'), ('age':30), ('city':'New York')])
```