Dictionaries are unordered collections of unique values stored in (Key-Value) pairs.

Once stored in a dictionary, you can later obtain the value using just the key.

For example, consider the Phone lookup where it is very easy and fast to find the phone number(value) when we know the name(Key) associated with it.

Characteristics of Python :

1. Unordered
2. Unique
3. Mutable

Creating a Dictionary

1. Using Curly{} bracket
2. Using dict() constructor

In [6]:

```
# create a dictionary using {}
person = {"name": "Sanvee", "country": "India", "telephone": 1178}
print(person)

# create a dictionary using dict()
person1 = dict({"name": "Sanvee", "country": "India", "telephone": 1178})
print(person)

# create a dictionary from sequence having each item as a pair
person2 = dict([("name", "Sanvee"), ("country", "India"), ("telephone", 1178)])
print(person)

# create dictionary with mixed keys keys
# first key is string and second is an integer
sample_dict = {"name": "Sanvee", 10: "Mobile"}
print(sample_dict)
```

```
{'name': 'Sanvee', 'country': 'India', 'telephone': 1178}
{'name': 'Sanvee', 'country': 'India', 'telephone': 1178}
{'name': 'Sanvee', 'country': 'India', 'telephone': 1178}
{'name': 'Sanvee', 10: 'Mobile'}
```

In [8]:

```
emptydict = {}
print(type(emptydict))
```

```
<class 'dict'>
```

In [9]:

```
#for empty set give set constructor
#empty_set = set()
#print(type(empty_set))
```

Unordered means that the items does not have a defined order, you cannot refer to an item by using an index.

In [14]:

```
#Dictionaries cannot have two items with the same key:
person = {"name": "Sanvee", "country": "India", "telephone": 1178, "telephone": 1234}
print(person)
```

```
{'name': 'Sanvee', 'country': 'India', 'telephone': 1234}
```

In [16]:

```
person = {"name": "Sanvee", "country": "India", "telephone": [1178, 1234]}
print(person)
```

```
{'name': 'Sanvee', 'country': 'India', 'telephone': [1178, 1234]}
```

Accessing elements of Dictionary

In [10]:

```
person = {"name": "Sanvee", "country": "India", "telephone": 1178}

# access value using key name in []
print(person['name'])

#  get key value using key name in get()
print(person.get('telephone'))
```

```
Sanvee
1178
```

In [11]:

```
#Get all keys and values
person = {"name": "Sanvee", "country": "India", "telephone": 1178}

# Get all keys
print(person.keys())
print(type(person.keys()))

# Get all values
print(person.values())
print(type(person.values()))

# Get all key-value pair
print(person.items())
print(type(person.items()))
```

```
dict_keys(['name', 'country', 'telephone'])
<class 'dict_keys'>
dict_values(['Sanvee', 'India', 1178])
<class 'dict_values'>
dict_items([('name', 'Sanvee'), ('country', 'India'), ('telephone', 1178)])
<class 'dict_items'>
```

In [12]:

```
#Length of dictionary
print(len(person))
```

```
3
```

Adding items to the dictionary

In [17]:

```
#Let's see how to add two new keys to the dictionary.
person = {"name": "Sanvee", 'country': "India", "telephone": 1178}

# update dictionary by adding 2 new keys
person["weight"] = 50
person.update({"height": 5.4})

# print the updated dictionary
print(person)
```

```
{'name': 'Sanvee', 'country': 'India', 'telephone': 1178, 'weight': 50, 'hei
ght': 5.4}
```

In [18]:

```
#We can also add more than one key using the update() method.
person = {"name": "Sanvee", 'country': "India"}

# Adding 2 new keys at once
# pass new keys as dict
person.update({"weight": 50, "height": 5.4})
# print the updated dictionary
print(person)

# pass new keys as as list of tuple
person.update([("city", "Mumbai"), ("company", "NetTech India",)])
# print the updated dictionary
print(person)
```

```
{'name': 'Sanvee', 'country': 'India', 'weight': 50, 'height': 5.4}
{'name': 'Sanvee', 'country': 'India', 'weight': 50, 'height': 5.4, 'city':
'Mumbai', 'company': 'NetTech India'}
```

In [28]:

```
dict1 = {'Sanvee': 70, 'Komal': 80, 'Shreya': 55}
dict2 = {'Pratham': 68, 'Suraj': 50, 'Siddhesh': 66}

# copy second dictionary into first dictionary
dict1.update(dict2)
# printing the updated dictionary
print(dict1)
```

```
{'Sanvee': 70, 'Komal': 80, 'Shreya': 55, 'Pratham': 68, 'Suraj': 50, 'Siddh
esh': 66}
```

In [35]:

```
dict1 = {'Sanvee': 70, 'Komal': 80, 'Shreya': 75}
dict2 = {'Pratham': 68, 'Suraj': 50, 'Shreya': 75}

# copy second dictionary into first dictionary
dict1.update(dict2)
# printing the updated dictionary
print(dict1)
```

```
{'Sanvee': 70, 'Komal': 80, 'Shreya': 75, 'Pratham': 68, 'Suraj': 50}
```

In [29]:

```
#We can unpack any number of dictionary and add their contents to another dictionary using
student_dict1 = {'Arnav': 1, 'Aashi': 2, }
student_dict2 = {'Yash': 5, 'Rudra': 6}
student_dict3 = {'Vini': 7, 'Simi': 9}

# join three dictionaries
student_dict = {**student_dict1, **student_dict2, **student_dict3}
# printing the final Merged dictionary
print(student_dict)
```

```
{'Arnav': 1, 'Aashi': 2, 'Yash': 5, 'Rudra': 6, 'Vini': 7, 'Simi': 9}
```

Modify the values of the dictionary keys

In [20]:

```
person = {"name": "Sanvee", "country": "India"}
print(person)
# updating the country name
person["country"] = "USA"
# print the updated country
print(person['country'])
print(person)

# updating the country name using update() method
person.update({"country": "Canada"})
# print the updated country
print(person['country'])
print(person)
```

```
{'name': 'Sanvee', 'country': 'India'}
USA
{'name': 'Sanvee', 'country': 'USA'}
Canada
{'name': 'Sanvee', 'country': 'Canada'}
```

Removing items from the dictionary

In [21]:

```
person = {'name': 'Sanvee', 'country': 'India', 'telephone': 1178, 'weight': 50, 'height':

# Remove last inserted item from the dictionary
deleted_item = person.popitem()
print(deleted_item)
# display updated dictionary
print(person)
```

```
('height', 5.4)
{'name': 'Sanvee', 'country': 'India', 'telephone': 1178, 'weight': 50}
```

In [23]:

```
person = {'name': 'Sanvee', 'country': 'India', 'telephone': 1178, 'weight': 50, 'height':
# Remove key 'telephone' from the dictionary
deleted_item = person.pop('telephone')
print(deleted_item)
# display updated dictionary
print(person)
```

```
1178
{'name': 'Sanvee', 'country': 'India', 'weight': 50, 'height': 5.4}
```

In [24]:

```
# delete key 'weight'
del person['weight']
# display updated dictionary
print(person)
```

```
{'name': 'Sanvee', 'country': 'India', 'height': 5.4}
```

In [25]:

```
# remove all item (key-values) from dict
person.clear()
# display updated dictionary
print(person)  # {}
```

```
{}
```

In [26]:

```
person = {'name': 'Sanvee', 'country': 'India', 'telephone': 1178, 'weight': 50, 'height':
# Delete the entire dictionary
del person
```

In [27]:

```
print(person)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-27-c3b2e0a07fd4> in <module>()
----> 1 print(person)

NameError: name 'person' is not defined
```

In [38]:

```
#Copy a Dictionary
dict1 = {'Sanvee': 75, 'Siddhesh': 55}

# Copy dictionary using copy() method
dict2 = dict1.copy()
# printing the new  dictionary
print(dict2)
```

```
{'Sanvee': 75, 'Siddhesh': 55}
```

In [40]:

```
dict1 = {'Sanvee': 75, 'Siddhesh': 55}
# Copy dictionary using assignment = operator
dict2 = dict1
print(dict2)
```

```
{'Sanvee': 75, 'Siddhesh': 55}
```

In [41]:

```
dict2.update({'Sanvee': 90})
print(dict2)

print(dict1)
```

```
{'Sanvee': 90, 'Siddhesh': 55}
{'Sanvee': 90, 'Siddhesh': 55}
```

Sort dictionary

In [42]:

```
dict1 = {'c': 45, 'b': 95, 'a': 35}

# sorting dictionary by keys
print(sorted(dict1.items()))

# sort dict eys
print(sorted(dict1))

# sort dictionary values
print(sorted(dict1.values()))
```

```
[('a', 35), ('b', 95), ('c', 45)]
['a', 'b', 'c']
[35, 45, 95]
```

Python Built-in functions with dictionary

max() and min()

In [43]:

```
dict = {1:'aaa',2:'bbb',3:'AAA'}
print('Maximum Key',max(dict)) #
print('Minimum Key',min(dict)) #
```

```
Maximum Key 3
Minimum Key 1
```

In [ ]:

```

```