Python frozenset()

- The frozenset() function returns an immutable frozenset object initialized with elements from the given iterable.
- Frozen set is just an immutable version of a Python set object. While elements of a set can be modified at any time, elements of the frozen set remain the same after creation.
- Due to this, frozen sets can be used as keys in Dictionary or as elements of another set. But like sets, it is not ordered (the elements can be set at any index).

The syntax of `frozenset()` function is:

```
frozenset([iterable])
```

frozenset() Parameters

The `frozenset()` function takes a single parameter:

- **iterable (Optional)** - the iterable which contains elements to initialize the frozenset with.
  Iterable can be set, dictionary, tuple, etc.

Example 1: Working of Python frozenset()

```
# tuple of vowels
vowels = ('a', 'e', 'i', 'o', 'u')

fSet = frozenset(vowels)
print('The frozen set is:', fSet)

# frozensets are immutable
fSet.add('v')
```

Output

```
The frozen set is: frozenset({'a', 'o', 'u', 'i', 'e'})
Traceback (most recent call last):
  File "<string>, line 8, in <module>
    fSet.add('v')
AttributeError: 'frozenset' object has no attribute 'add'
```

When you use a dictionary as an iterable for a frozen set, it only takes keys of the dictionary to create the set.

```
# random dictionary
person = {"name": "John", "age": 23, "sex": "male"}

fSet = frozenset(person)
print('The frozen set is:', fSet)
```

Output

```
The frozen set is: frozenset({'name', 'sex', 'age'})
```

frozenset operations

Like normal sets, frozenset can also perform different operations like `copy`, `difference`, `intersection`, `symmetric_difference`, and `union`.

```
# Frozensets
# initialize A and B
A = frozenset([1, 2, 3, 4])
B = frozenset([3, 4, 5, 6])

# copying a frozenset
C = A.copy()  # Output: frozenset({1, 2, 3, 4})
print(C)

# union
print(A.union(B))  # Output: frozenset({1, 2, 3, 4, 5, 6})

# intersection
print(A.intersection(B))  # Output: frozenset({3, 4})

# symmetric_difference
print(A.symmetric_difference(B))  # Output: frozenset({1, 2, 5, 6})
```