Everything about Python Sets

Creating

```
# Empty set
my_set = set()

# Set with elements
fruits = {"apple", "banana"}

# Using set() function
numbers = set([1, 2, 3, 4, 5])
```

Accessing Characters

- Sets are unordered, so elements cannot be accessed by index.
- Use a loop to iterate over elements.

```
for fruit in fruits:
    print(fruit)
```

Editing a sets

```
# Adding elements
fruits.add("orange")

# Updating with multiple elements
fruits.update(["grape", "mango"])
```

Frozenset

- A frozenset is an immutable version of a set.
- you can perform set operations like union, intersection, and difference.
- Used as dictionary keys.

```
# Creating a frozenset
fset = frozenset([1, 2, 3, 4, 5])
print(fset)  # frozenset({1, 2, 3, 4, 5})
```

Sets Operations

```
A = {1, 2, 3}; B = {3, 4, 5}

# Union
print(A | B)  # {1, 2, 3, 4, 5}
print(A.union(B))

# Intersection
print(A & B)  # {3}
print(A.intersection(B))

# Difference
print(A - B)  # {1, 2}
print(A.difference(B))

# Symmetric Difference
print(A ^ B)  # {1, 2, 4, 5}
print(A.symmetric_difference(B))
```

Set Unpacking

Basic

```
fruits = {"apple", "banana", "cherry"}

a, b, c = fruits  # Unpacks all elements into variables
print(a, b, c)
# Output: apple banana cherry (order may vary due to set's unordered nature)
```

use *

```
numbers = {1, 2, 3, 4, 5}

first, *middle, last = numbers
print(first)
print(middle)  # Remaining elements as a list
print(last)
```

Deleting a sets

```
# Remove an element (error if not found)
fruits.remove("banana")

# Remove an element (no error if not found)
fruits.discard("apple")

# Pop a random element
fruits.pop()

# Clear all elements
fruits.clear()

# Delete the set
del fruits
```

Set Functions

When/ Why	Function	Input	Output
Copy Set	copy()	new_s = s.**copy()**	{1,2,3} (copy)
Length	len()	**len(**{1,2,3}**)**	3
Check Subset	issubset()	{1,2}.**issubset(**{1,2,3}**)**	True
Check Superset	issuperset()	{1,2,3}.**issuperset(**{2,3}**)**	True
No common element	isdisjoint()	{1, 2, 3}.**isdisjoint(**{7, 8}**)**	True
Sorted	sorted()	{1, 3, 2}.**sorted()**	[1, 2, 3]

Set Operations