Medha Nikalesh Bapu

24030331245030

1) Create a set and display its elements

In [1]:
```python
my_set = {1, 2, 3, 4, 5}
print("Elements of the set:")
for element in my_set:
    print(element)
```

Elements of the set: 1 2
3 4 5

2) Add an element to a set

In [4]:
```python
my_set = {"apple", "banana", "cherry"}
my_set.add("orange")
print(my_set)
```

{'apple', 'orange', 'cherry', 'banana'}

3) Remove an element from a set

In [6]:
```python
fruits = {"apple", "banana", "cherry"}
fruits.remove("banana")
print(fruits)
```

{'apple', 'cherry'}

4) Clear all elements from a set

In [11]:
```python
my_set = {1, 2, 3, 4}
my_set.clear()        print(my_set)
```

set()

5) Copy a set to another set

```
In [12]:    original = {1, 2, 3}
            duplicate = original.copy()
            print(duplicate)
```

{1, 2, 3}

6) Check if an element exists in a set

```
In [15]:    my_set = {"apple", "banana", "cherry"}

            if "banana" in my_set:
                print("'banana' is in the set.")
            else:
                print("Not found.")
```

'banana' is in the set.

7) Find the length of a set

```
In [2]:    my_set = {"apple", "banana", "cherry"}
           print(len(my_set))
```

3

8) Iterate through a set using a loop

```
In [3]:    my_set = {"apple", "banana", "cherry"}
           iterator = iter(my_set)
           for element in iterator:
               print(element)
```

apple
banan
a
cherry

Set Operations

9) Find the union of two sets

```
In [2]:    set1 = {1, 2, 3} set2
           = {3, 4, 5}

           union_set = set1.union(set2)
           print("Union of two sets:", union_set)
```

Union of two sets: {1, 2, 3, 4, 5}

10) Find the intersection of two sets

In [4]:
```python
set1 = {1, 2, 3} set2 = {2, 3, 4}

intersection_set = set1.intersection(set2)
print("Intersection of two sets:", intersection_set)
```

Intersection of two sets: {2, 3}

11) Find the difference between two sets

In [6]:
```python
set1 = {1, 2, 3, 4} set2 = {3, 4, 5, 6}

difference_set = set1.difference(set2)
print("Difference between two sets:", difference_set)
```

Difference between two sets: {1, 2}

12) Find the symmetric difference between two sets

```python
set1 = {1, 2, 3, 4} set2 = {3, 4, 5, 6} sym_diff = set1.symmetric_difference(set2) print("Symmetric Difference:", sym_diff)
```

13) Check if one set is a subset of another

In [16]:
```python
A = {1, 2, 3}
B = {1, 2, 3, 4, 5}
if A.issubset(B):
    print("A is a subset of B")
elif B.issubset(A):
    print("B is a subset of A ")
else:
    print("A is not a subset of B and also B is not a subset of A")
```

A is a subset of B

14) Check if one set is a superset of another

In [20]:
```python
A = {1, 2, 3, 4, 5}
B = {1, 2, 3}
if A.issuperset(B):
    print("A is a superset of B")
elif B.issuperset(A):
    print("B is a superset of A ")
else:
    print("A is not a superset of B and also B is not a superset of A")
```

A is a superset of B

15) Check if two sets are disjoint

In [21]:
```python
A = {1, 2, 3} B = {4, 5,
6} if A.isdisjoint(B):

    print("A and B are disjoint sets")
else:
    print("A and B are not disjoint sets")
```

A and B are disjoint sets

Set Applications

16) Remove duplicates from a list using a set

In [22]:
```python
my_list = [1, 2, 2, 3, 4, 4, 5]
unique_set = set(my_list)
unique_list = list(unique_set)
print("Original list:", my_list)
print("List after removing duplicates:", unique_list)
```

Original list: [1, 2, 2, 3, 4, 4, 5]
List after removing duplicates: [1, 2, 3, 4, 5]

17) Convert a list to a set and back to a list

In [23]:
```python
my_list = [1, 2, 2, 3, 4, 4, 5]
my_set = set(my_list)
new_list = list(my_set)
print("Original list:", my_list)
print("After converting to set and back to list:", new_list)
```

Original list: [1, 2, 2, 3, 4, 4, 5]
After converting to set and back to list: [1, 2, 3, 4, 5]

18) Find common elements in two lists using sets

In [24]:
```python
list1 = [1, 2, 3, 4, 5] list2 = [4, 5, 6, 7, 8] set1 =
set(list1) set2 = set(list2) common_elements =
set1.intersection(set2)        common_list        =
list(common_elements)              print("Common
elements:", common_list)
```

Common elements: [4, 5]

19) Get elements in one list but not in another using sets

In [25]:

```python
list1 = [1, 2, 3, 4, 5] list2 = [4, 5, 6, 7] set1 = set(list1)
set2 = set(list2)
difference = set1.difference(set2)
result = list(difference)
print("Elements in list1 but not in list2:", result)
```

Elements in list1 but not in list2: [1, 2, 3]

20) Use set comprehension to generate a set (e.g., squares or primes)

In [26]:

```python
squares = {x**2 for x in range(1, 11)}
print("Squares from 1 to 10:", squares)
```

Squares from 1 to 10: {64, 1, 4, 36, 100, 9, 16, 49, 81, 25}

Frozen Set Programs

21) Create and print a frozenset

In [27]:

```python
fset = frozenset([1, 2, 3, 4, 5])
print("Frozenset:", fset)
```

Frozenset: frozenset({1, 2, 3, 4, 5})