Python Set Methods & Symbolic Operations

1. Python Set Methods

Assume: A = {1, 2, 3, 4}, B = {3, 4, 5, 6}

1. union(): A.union(B) => {1, 2, 3, 4, 5, 6}
2. intersection(): A.intersection(B) => {3, 4}
3. difference(): A.difference(B) => {1, 2}
4. symmetric_difference(): A.symmetric_difference(B) => {1, 2, 5, 6}
5. update(): A.update(B) => A becomes {1, 2, 3, 4, 5, 6}
6. intersection_update(): A.intersection_update(B) => A becomes {3, 4}
7. difference_update(): A.difference_update(B) => A becomes {1, 2}
8. symmetric_difference_update(): A.symmetric_difference_update(B) => A becomes {1, 2, 5, 6}
9. isdisjoint(): A.isdisjoint(B) => False
10. issubset(): A.issubset(B) => False
11. issuperset(): A.issuperset(B) => False
12. add(): A.add(7) => Adds 7 to A
13. remove(): A.remove(3) => Removes 3 from A (error if not found)
14. discard(): A.discard(3) => Removes 3 if present (no error)
15. pop(): A.pop() => Removes and returns random element
16. clear(): A.clear() => Empties the set

2. Set Operations using Symbols

A = {1, 2, 3, 4}

B = {3, 4, 5, 6}

1. Union (|): A | B => {1, 2, 3, 4, 5, 6}
2. Intersection (&): A & B => {3, 4}
3. Difference (-): A - B => {1, 2}
4. Symmetric Difference (^): A ^ B => {1, 2, 5, 6}

Python Set Methods & Symbolic Operations

5. Subset (<, <=): A < B => True if A is proper subset of B

6. Superset (>, >=): A > B => True if A is proper superset of B