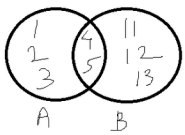
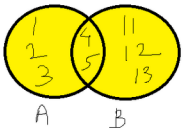
Sets

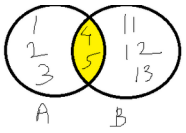
1. It stores unique values.
2. It is mutable.
3. It is unordered.
4. It is represented using {} or converter function set()
5. To create an empty set use s=set()

Set.add(value)	It adds a single value in the set only if it is immutable and unique
Set.update(iterable)	It adds all the values from iterable to set one by one only if it is immutable and unique
Set.pop()	It will delete the data randomly from the set
Set.remove(value)	It will delete the given value if present otherwise throws exception
Set.discard(value)	It will delete the given value if present otherwise it will ignore
Set.clear()	It removes all the values from the set, but it will keep empty set
Set.copy()	It creates a shallow copy of the set
S1.union(s2) S1.|s2	Will display all values of s1 and s2
S1.intersection(s2) S1&s2	It will display only common values
S1.difference(s2) S1-s2	It will display values only in s1
S1.difference_update(s2) S1=s1-s2	It will display values only in s1 and also overwrite s1
S1.symmetric_difference(s2) S1^s2	Only values in either s1 or s2
S1.symmetric_difference_update(s2) S2=s1^s2	Only values in either s1 or s2, and will overwrite s1

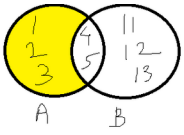




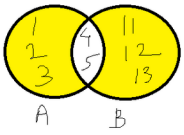
A.union(B) A|B



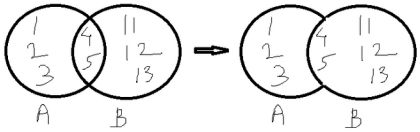
A.intersection(B)
A&B



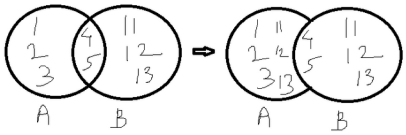
A.difference(B)



A.Symmetric_differen
ce(B)



A.difference_update(
B)



A.symmetric_differen
ce_update(B)