Collection :-

- If we want to represent a group of individual objects as a single entity then we should go for collection.

Collection Framework :- It contains several classes and interfaces which can be used to represent a group of individual objects as a single entity.

9 Key Interfaces of collection Framework :-

(1) collection (2) List (3) set (4) Sorted set
(5) Navigable Set (6) Queue (7) Map (8) Sorted Map
(9) Navigable Map.

(1) Collection (I) :-

- If we want to represent a group of individual objects as a single entity then we should go for collection.

- Collection interface defines the most common methods which are applicable for any collection object.
- In genral, collection interface is concidered as Root interface of collection framework.
- There is no concrete class which implements collection interface directly.

Diff b/w collection and collections :-

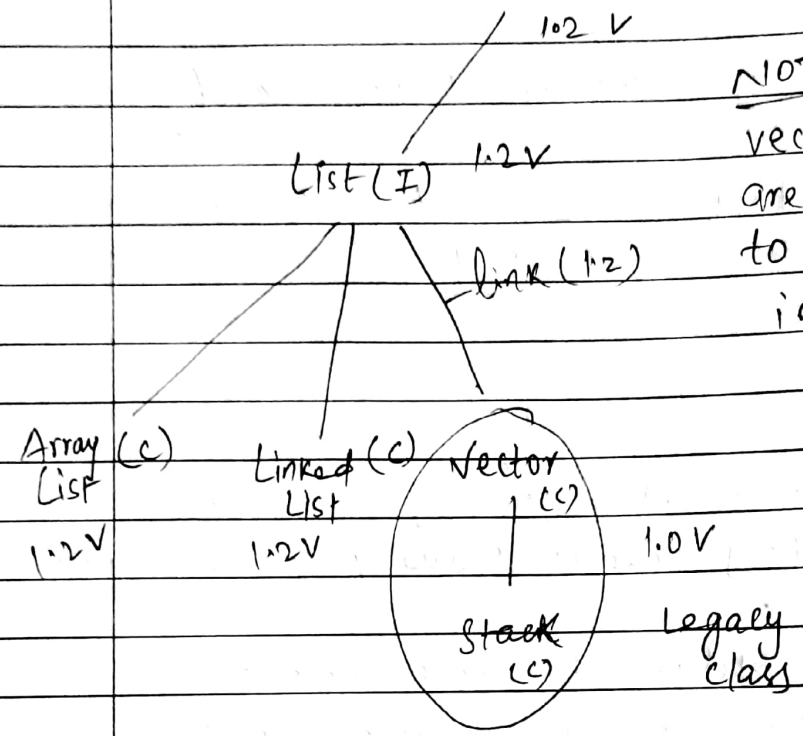
collection	collections
Interface	(1) Class. (Utility class)
If we want to represent a group of individual objects as a single entity then we should go for collection.	(2) Collections is an utility class present in java.util package to define several utility methods for collection objects (like Sorting, Searching etc)

⇒ List (I) :→

→ It is the child interface of Collection.

→ If we want to represent a group of individual Objects as a single entity where duplicates are allowed and insertion order must be preserved then we should go for List(I) interface.

→



NOTE :- In 1.2 version, vector and stack classes are re-engineered are to implement in List interface.

⇒ Set (I) :→

→ It is the child interface of Collection

→ If we want to represent a group of individual Objects as a single entity where duplicates are not allowed and insertion order not required then we should go for set interface.

⇒