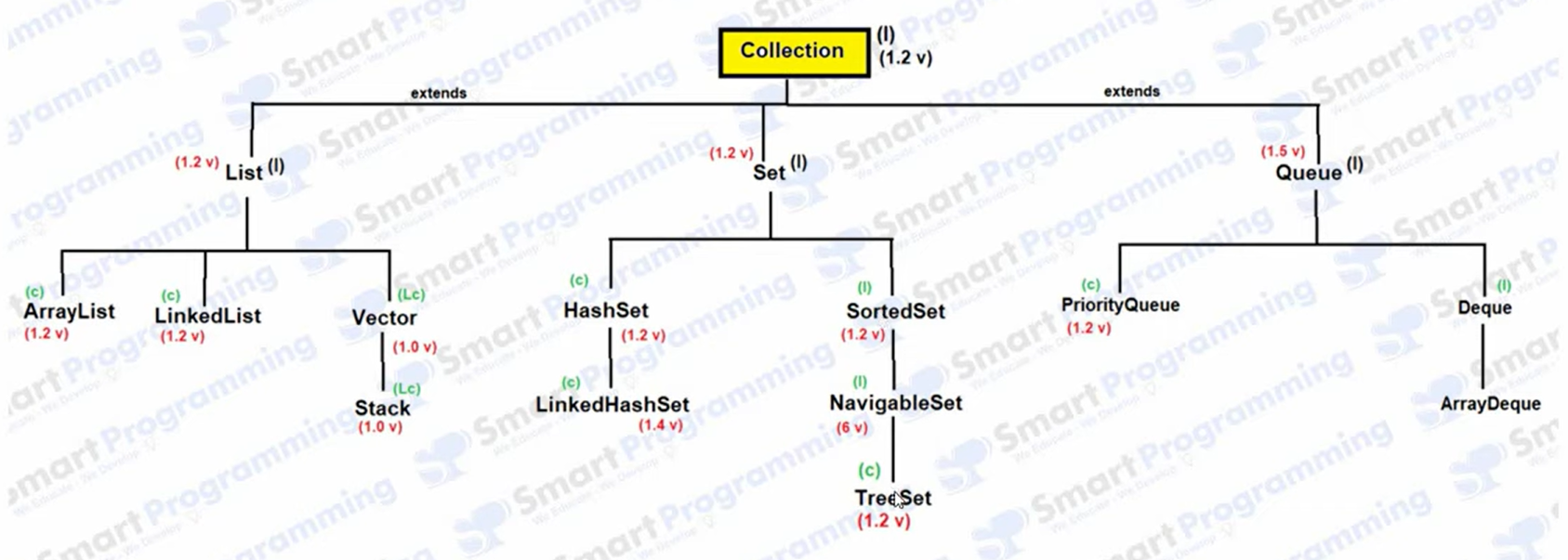
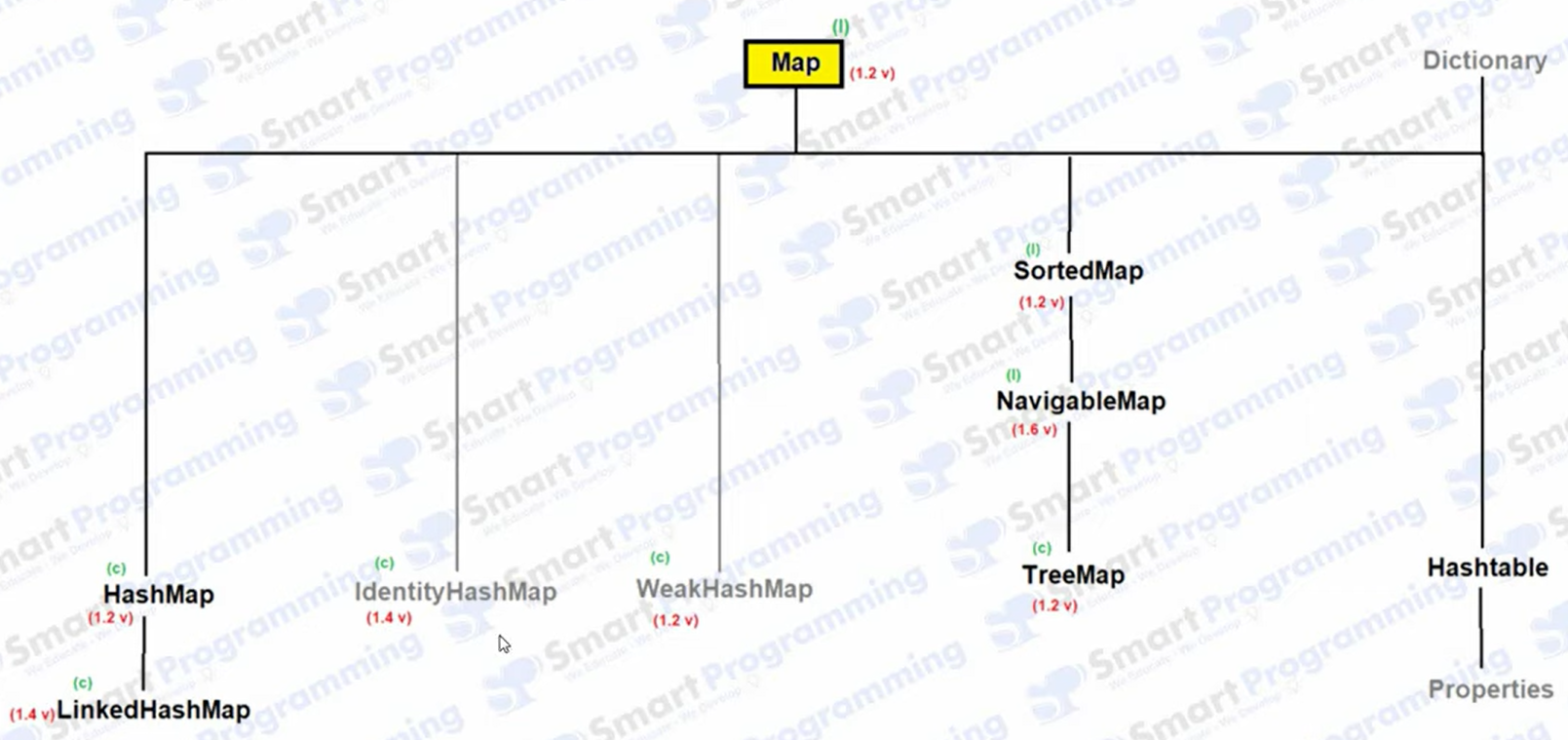
**@ Collection Framework**

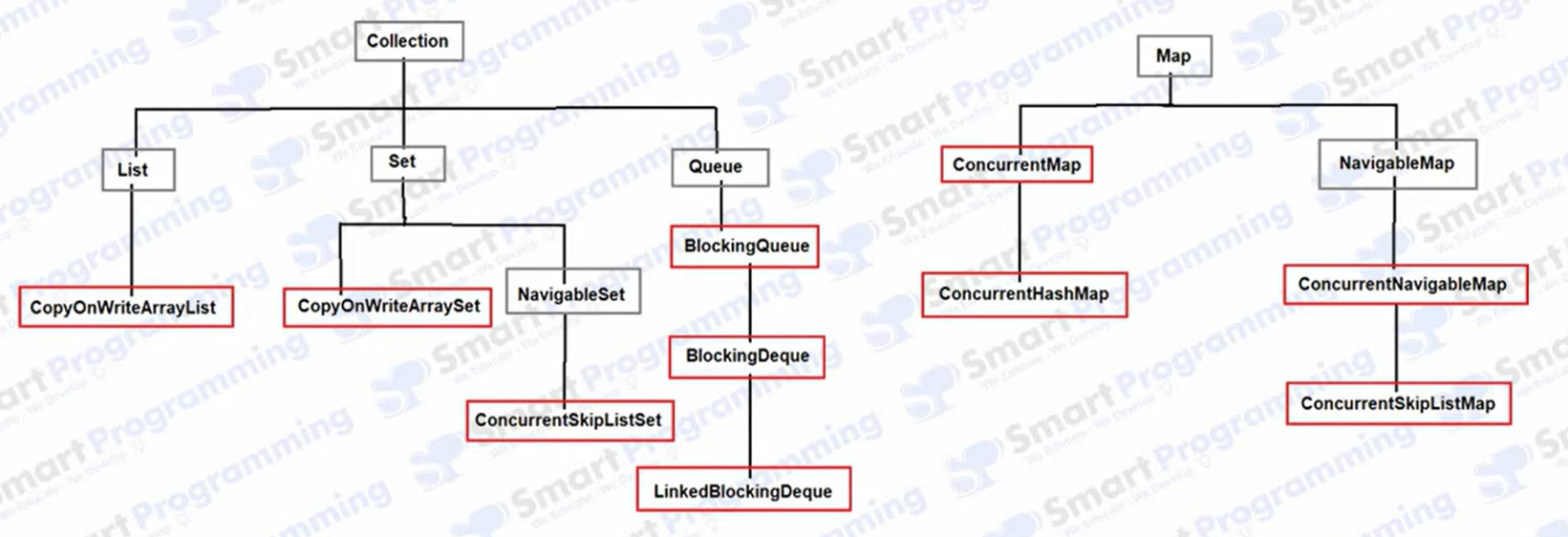
* **‘Collection’ and ‘Map’ together form collection framework**

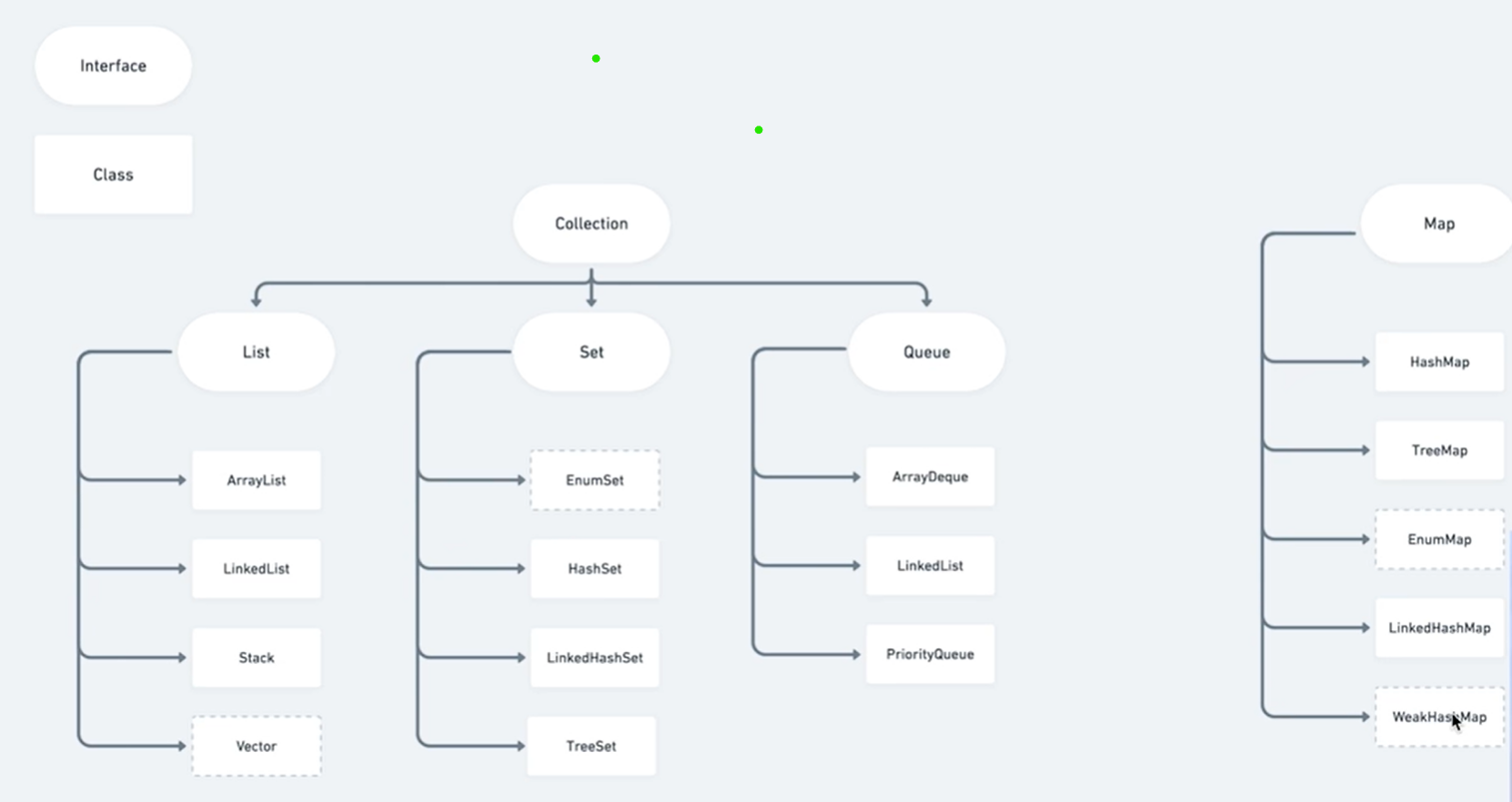
# Collection Hierarchy

[ in fig, ‘I’ = Interface || ‘C’ = class || ‘LC’ = Legacy Class || ‘v’ = version ]

# Map Hierarchy

[ in fig, ‘I’ = Interface || ‘C’ = class || ‘LC’ = Legacy Class || ‘v’ = version ]

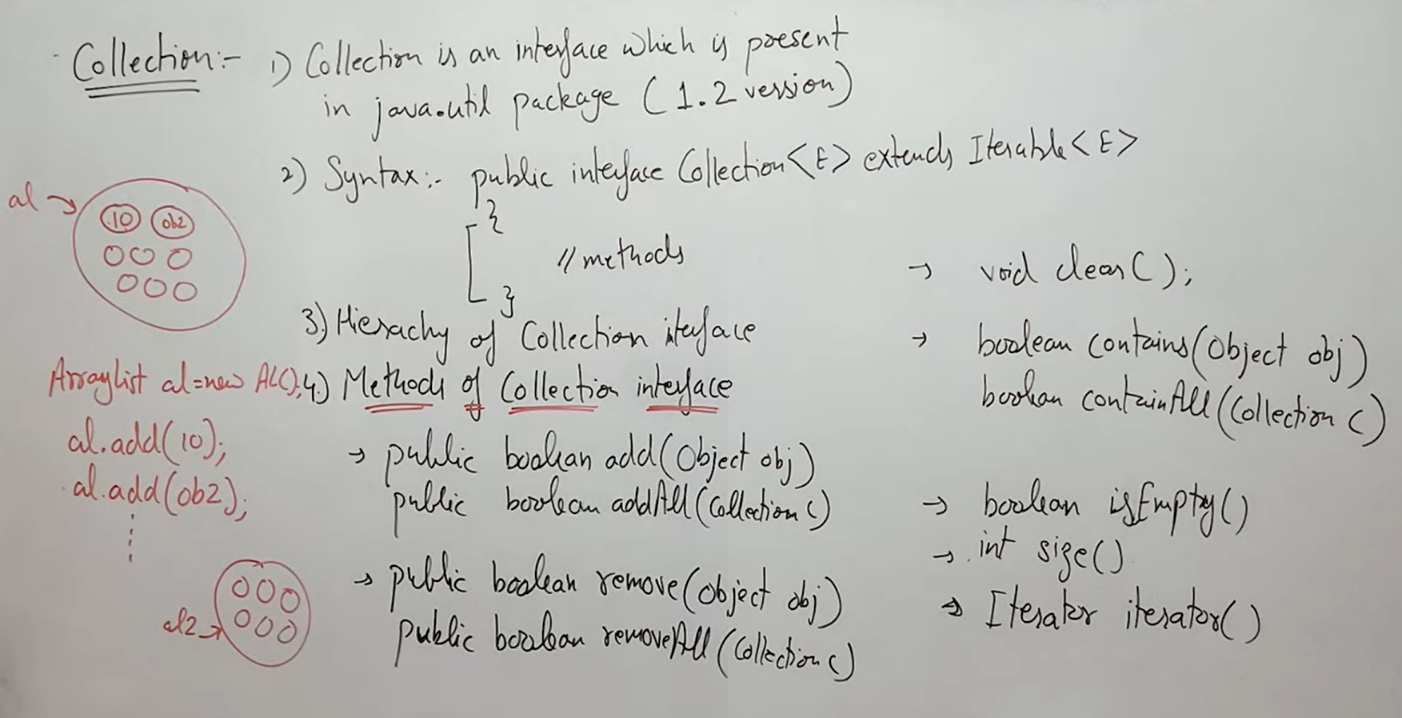
# Concurrent classes ( added latter )



**Note:** ‘List’, ‘set’ etc. are Interface, So we can’t create object of ‘Interface’ and ‘Abstract class’. Object can be created for class which implement them. i.e. ArrayList, Stack.

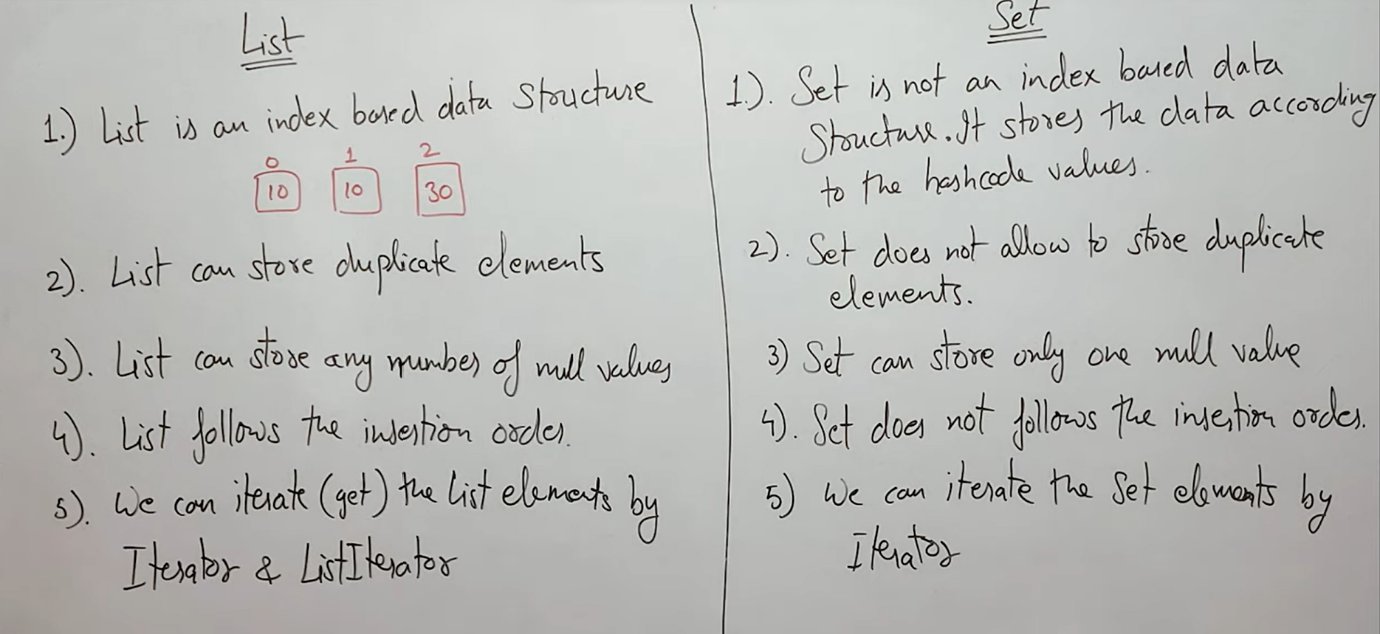
/////////////////////////////////////////////////////////////////////////////////////////////////////////

**# Collection Interface**



[Note: collection contains all common methods for child classes, that would be helpful.]

# Difference between ‘List’ and ‘Set’



* ‘Set’ uses Hashcode(?) that’s why we can’t store duplicate in it.
* ‘Iterator’ can only iterate in forward direction. But ‘List-Iterator’ can iterate in both direction

////////////////////////////////////////////////////////////////////////////////////////////////////////

***#### Abstract Data Type***

* Doesn,t dictate how the data is organized
* Dictates the operations you can perform
* Concrete data structure is usually a concrete class
* Abstract data type is usually an interface

# List ( <https://docs.oracle.com/javase/8/docs/api/java/util/List.html> )

* Import java.util.list; /or/ java.util.\*;
* Abstract data type

1) List (Interface) > List is a child of Collection(Interface).

\* When to use:

a) when order is necessary, the insertion order should be preserved.

b) Duplicated elements are allowed.

\* Classes which has implemented List:

a) ArrayList

b) LinkedList

c) Vector,Stack(Legacy Classes)

2) Set (Interface) > Set is a child of Collection

\* When to use:

a) Insertion Order is not preserved.

b) Duplicates are not allowed.

\* Classes which has implemented List:

a) LinkedHashSet

b) bHashSet

3) Queue (Interface) > Set is a child of Collection

\* When to use:

FIFO Concept is used

\* Classes which has implemented Queue

a) priorityQueue

4) Map > Independent Interface called MAP (not a child interface)

Present in key value pair.

Key and value are object.

the keys cannot be duplicated.

the values can be duplicated.

Classes:

HashMap

LinkedHashMap