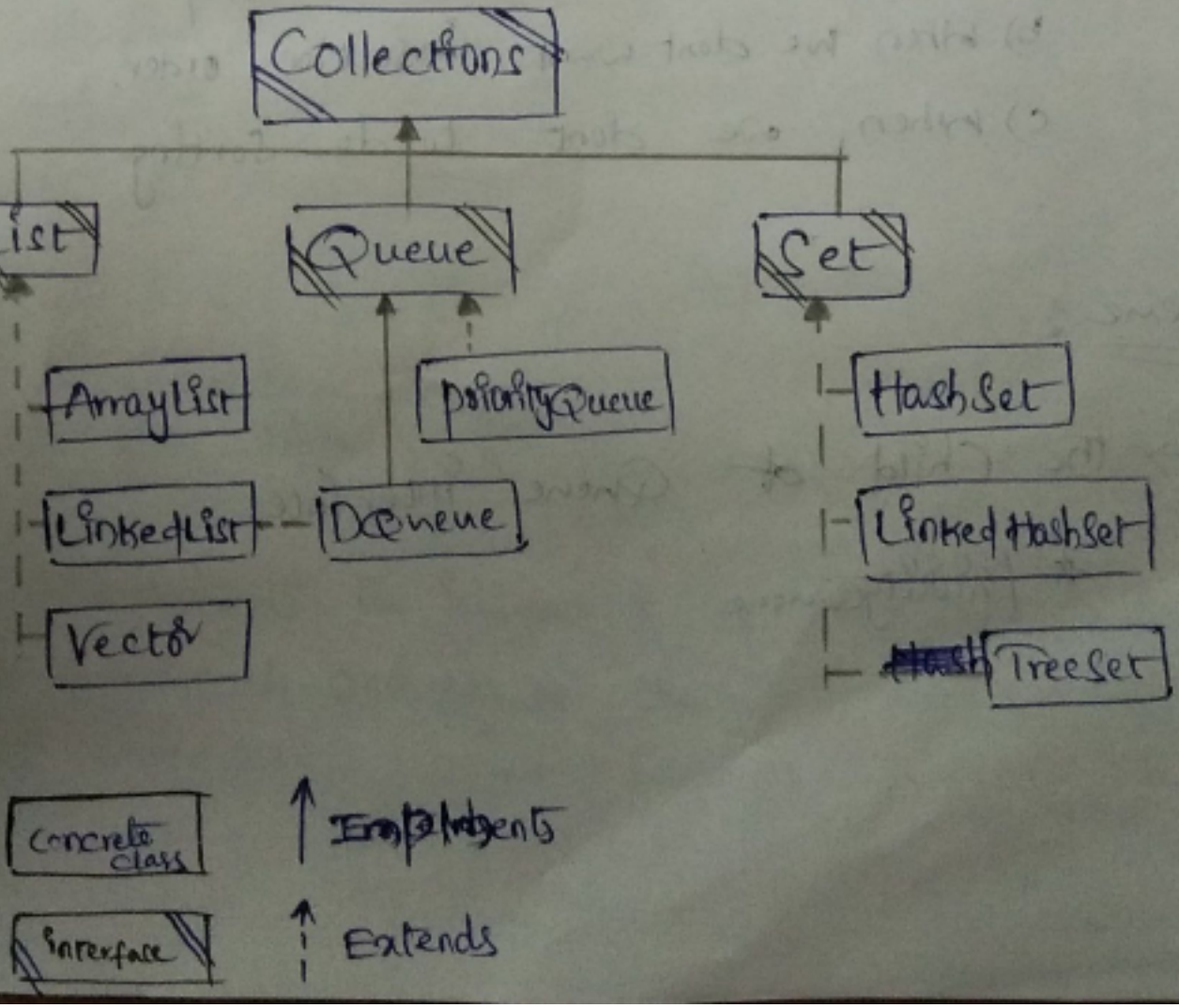
1. What are the main differences B/W array and Collection?

(A)

Array	Collection
1. Array itself a data structure and has some restrictions for entering values.	1. Collection has various data structure available providing freedom to users for manipulation of objects.
2. Arrays are not growable	2. Collections are growable
3. Array elements cannot be removed	3. In collections elements canbe removed and modified
4. Arrays doesn't allow null values	4. Collections allow NULL values

2. Explain various interfaces used in Collection frameWorks?

A



(2) List:

→ The childs of List interface are

* ArrayList
* Vector
* LinkedList

→ List is adopted when,

a. insertion order is of higher priority
b. When inserting or deleting the elements wanted to be feasible.
c. When sorting is of last priority.

Set:

→ The child of Set are

* HashSet
* LinkedHashSet
* TreeSet

→ Set is adopted when

a) Duplicate value is to be removed
b) When we dont want insertion order.
c) When we dont to do sorting.

Queue:

→ The child of Queue interface

* PriorityQueue

3. What is the difference B/W ArrayList & Vector?

(A)

ArrayList	Vector
1. ArrayList comes from non-legacy class i.e., from version 1.2	1. Vector comes from legacy class i.e., from version 1.0.
2. Arraylist is not Synchronized	2. Vector is Synchronized
3. It is thread Safe	3. It is thread Safe, but not fast
4. Initially the new capasity is meant to be → (3/2 * oldcapasity)+1) later changed to 50% (i.e., 0.5% of load-factor)	4. The capasity is (2 * old capasity) i.e., 100%

4. What is the difference B/H ArrayList & LinkedList?

(A)

ArrayList	LinkedList
1. Underline data Structure is Growable (or) resizable array	1. Underline data structure is doubly linked list
2. Best suited operation is for data retrival and insertion order given higher priority.	2. Best suited operation is for insersion and deletion feasibility
3. It implements the interfaces of Serializable and Cloneable and random access	3. The secondary interface of LinkedList Serializable, Cloneable

5. What is the difference B/W Iterator & ListIterator?

(A)

Iterator	ListIterator
1. Iterator is the universal class that helps to fetch the data (among the other things)	1. ListIterator is the interface that extends interface
2. Iterator can only point forward	2. ListIterator can print backward & forword
3. It prints from first to last	3. List Iterator can print for any specified location.

⑥ What is the difference B/W List & Set?

Ⓐ

List	Set
1. The List implementation allows us to add the same or duplicate elements	1. The Set implements doesn't allow us to add the same or duplicate elements.
2. The insersion Order is maintained by the List	2. It doesn't maintain the insertion Order of elements
3. List allows us to add any number of null values	3. The Set allows us to add at least one null value in it.
4. The List implementation class are LinkedList and Array List	4. The Set implementation classes are TreeSet, HashSet and LinkedHashSet
5. We can get the element of a specified index from the List using the get() method	5. We cannot find the element from the Set based on the index because it doesn't provide any get method()

7. What is the difference b/w HashSet & TreeSet?

(A)

HashSet	TreeSet
1. It does not provide a guarantee to sort the data	1. It provide a guarantee to sort the data. Sorting depends on the supplied Comparator.
2. In HashSet only an element can be null	2. It does not allow null elements.
3. It uses hashcode() or, equals() method for Compasion.	3. It uses compare() (or) CompareTo() methods for Comparision.
4. It is fasten than treeset	4. It is slowen in Comparision to Hashset
5. Underline datastructure is hashtable	5. Underline datastructure is Binary tree.
6. It allows only hetrogeneous value	6. It allows only homogeneous value.

8. What is the diffrence B/w HashSet & HashMap?

(A)

HashSet	HashMap
1. HashSet is a set. It creates a collection that uses a has table for storage.	1. HashMap is a hash table based implementation of map interface.
2. Secondary interface is Serializable, Cloneable, Iterable, Collections.	2. Secondary interface is Serializable and Cloneable.
3. It doesn't allow duplicates	3. It does not allow duplicate keys, but duplicate values are allowed.
4. It can contain a single null value.	4. It allows a single null value and multiple null values
5. It uses add() method to add elements in the HashSet	5. Hash map uses the put() method to add the elements in the hash map

9. Difference B/W HashMap & HashTable?

HashMap	HashTable
1. HashMap is non-Synchronized. It is not-thread safe and can't be shared between many threads without proper Synchronization Code.	1. HashTable is Synchronized. it is thread-safe and can be shared with many threads.
2. HashMap allows one null key and multiple null values	2. HashTable doesn't allow any null key or value
3. Hashmap is a new class introduced in JDK 1.2	3. HashTable is Legacy class.
4. Hashmap is fast.	4. Hashtable is slow
5. Iterator in hashmap is fail-fast	5. Enumerator in HashTable is not fail-fast.

10. What is difference B/W Comparable & Comparator?

(A)

Comparable	Comparator
1. Comparable provides a single sorting sequence. In other words, we can sort the collection on the basis of single element such as id, name and price.	1. The Comparator provides multiple sorting sequences. In other words. we can sort the collection on the basis of multiple element such as id, name and price etc.
2. Comparable affects the original class. i.e., the actual class is modified	2. Comparator doesn't affect the original class. i.e., the actual class is not modified.
3. Comparable provides compareTo() method to sort elements.	3. Comparator provides compare() method to sort elements.
4. Comparable is present in java.lang package	4. A Comparator is present in the java.util package.

11. How to Synchronize List, Set and Map elements?

(A) 1. In order to get a Synchronized list from ArrayList we use Synchronized List (List<T>) method.

2. Collections SynchronizedSet() method is used to Synchronize Set ~~Collections~~.elements.

3. Collections Synchronized map (Hashmap) method is used to Synchronize map elements.

12. What do you understand by fail fast & fail safe?

(A) In Java, Collections Supports two types of iterators they are.

1. fail-fast
2. fail safe

} these are very useful in exception handling.

1. fail-fast:

The fail-fast iterator aborts the operation as soon it exposes failure and stops the entire operation comparitevely.

2. fail safe:

The fail safe iterator doesn't aborts the operation in case of failure. Instead it tries to avoid failures as much as possible.

(13) Difference B/W Array & Arraylist?

(A)

Array	Arraylist
1. Array is dynamically created obj	1. The Arraylist is a class of java collections framework.
2. Array is static in size	2. Arraylist is dynamic in size
3. Array is fixed length data structure	3. Arraylist is variable length data structure
4. It's mandatory to provide size of an array why initializers it directly or indirectly	4. we can create an instance of array list without specifiers its size.