

## Collection (Blog)

Date: \_\_\_\_\_

- The collection is a set of 4 basic interfaces Collection, List, Set and Map.
- Set can contain unique objects where List can contain duplicate objects.

Q. Why does map contains key-value pairs but value and key cannot be null.

A. Write code to reverse an array list in Java.

Ans. Collections.reverse(List) method is used to reverse a Collection in java.

Q. How can you synchronize a map ?

A. Using Collections.synchronizedMap(Map).

Q. How can you synchronize a List ?

A. Using Collections.synchronizedList(List).

Q. What is the internal working of Hash Collection.

A. Hash collection uses a hash code to uniquely identify an object in the hash collection. Hashing is a process of converting an object into

Q. What are differences between HashSet And TreeSet ?

Ans : → The collection framework provides two general purpose implementation often Set interface.  
1) HashSet and ② TreeSet  
Both store unique elements but TreeSet guarantees that elements of the set will be in sorted order where as HashSet does not.

Q. What is hashCode ?

Ans : → A hashCode is a unique number, it is used by hash collections to identify and search an object. Method hashCode() returns the hashCode of an object.

Q. Why do you override equals() and hashCode() both methods in collection classes ?

Ans : → hashCode is used to uniquely identify an object in Hash collections like HashMap and HashSet.

Method hashCode() returns unique code for an object. The hashCode generation algorithm can be changed by overriding the hashCode() method.

In order to maintain consistent comparison in the system we need to override hashCode() method.

IMPORTANT NOTES

Q. What is the internal working of Hash Collection.

Ans : → Hash collection uses a hash code to uniquely identify an object in the hash collection.

Integer from by using the method `highCode()`.

Q. What are the differences between Comparable and Comparator interfaces?  
Ans: → Both interfaces are used to compare two objects and sort the collections.

Comparable compares the current object with another object of the same class does not compare classes with same object of different classes.

For example, Employee class does implement Comparable interface and Santa object of employee class will compare it self with Bunker object of Employee class.

Comparable interface exists in java.util package

Comparator interface compares the two different objects of single class: for example, Math class will compare two objects of Name class → we can create multiple comparator for single class.

The Comparator interface exists in `java.util` package

IMPORTANT NOTES

Q. In this is natural ordering?  
Ans: → Objects are compared and sorted on the basis of a primary key.

Q. Which key do you choose to do natural ordering?  
Ans: → We use the primary key attributes.

Q. How many Comparators you can create for a single class say "Employee"?  
Ans: → If the Employee object need to sort on Name and Id attributes then we will create two comparators one for Name and the second for Id .that means if employee class needs to sort on 4 attributes the we will create 4 comparators, one for each attribute.

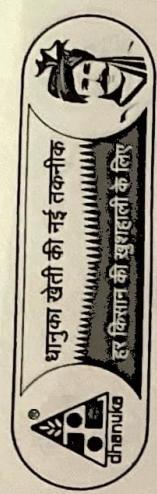
Q. How do you sort a collection using a Comparator?  
Ans: → Collections.sort(Collection<T>)

Q. Where are differences between Iterator and Enumeration interfaces?  
Ans: → Both are the same , Iterator has one additional method `remove()` to remove an element from a collection.

IMPORTANT NOTES

Q. What is iterator?  
Ans: → Iterator is used to read data sequentially from a collection . Collection interface contains Iterator() method that returns iterator object

Date :



Q. Which key do you choose to do natural ordering?

Ans: → We use the primary key attributes.

Q. How many Comparators you can create for a single class say "Employee"?  
Ans: → If the Employee object need to sort on Name and Id attributes then we will create two comparators one for Name and the second for Id .that means if employee class needs to sort on 4 attributes the we will create 4 comparators, one for each attribute.

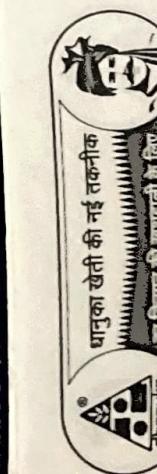
Q. How do you sort a collection using a Comparator?  
Ans: → Collections.sort(Collection<T>)

Q. Where are differences between Iterator and Enumeration interfaces?  
Ans: → Both are the same , Iterator has one additional method `remove()` to remove an element from a collection.

IMPORTANT NOTES

Q. What is iterator?  
Ans: → Iterator is used to read data sequentially from a collection . Collection interface contains Iterator() method that returns iterator object

Date :



Q. Which key do you choose to do natural ordering?  
Ans: → We use the primary key attributes.

Q. How many Comparators you can create for a single class say "Employee"?  
Ans: → If the Employee object need to sort on Name and Id attributes then we will create two comparators one for Name and the second for Id .that means if employee class needs to sort on 4 attributes the we will create 4 comparators, one for each attribute.

Q. How do you sort a collection using a Comparator?  
Ans: → Collections.sort(Collection<T>)

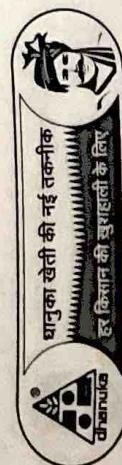
Q. Where are differences between Iterator and Enumeration interfaces?  
Ans: → Both are the same , Iterator has one additional method `remove()` to remove an element from a collection.

IMPORTANT NOTES

Q. What is iterator?  
Ans: → Iterator is used to read data sequentially from a collection . Collection interface contains Iterator() method that returns iterator object

Java Collection Classes :-

- ① ArrayList
- ② LinkedList
- ③ PriorityQueue



Date : \_\_\_\_\_

Q. What is Enumerator ?

Ans: → Just like Iterator, it reads data sequentially from collection objects, it is used with List and ArrayList classes.

Q. What are Generics ?  
Ans: → When you define data types of collection, it is called Generics.

ArrayList & String > l = new ArrayList<String>;

Q. What is Auto boxing and Auto Unboxing ?  
Ans. Auto boxing : → It is convert primitive data to wrapper class.

Auto Unboxing : → It is convert wrapper class to primitive data.

Q. What is Concurrent Collection ?  
Ans: → It is introduced in JDK 1.5 in java.util.concurrent package.

Regular collections require Concurrent modification exception when a collection is structurally modified after creating an iterator.  
It is called failFast condition

Concurrent Collection allows the addition and deletion of an element of an iterator if reader mode modification exception.

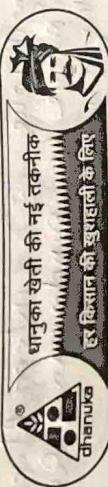
\* Internal Working of hash collection :-

- Internally Hash Collection uses a hashcode to uniquely identify an object while saving or searching an element.
- Method hashCode() is used to calculate and get the hashcode of an object. The process of converting an object into hashcode is called hashing.
- Method equals() is used to compare two objects in the Hash collection.
- If you override hashCode() method then you must override equals() method for consistent comparison of objects.
- Hash Collection uses an array to store its elements.
- If two elements have the same hashcode then multiple elements are stored at the same array index locations as per Net.

IMPORTANT NOTES

- The bucket is a linked list that contains elements of the same hash code.

Date : \_\_\_\_\_



**Q. What is the stream API ?**  
**Ans:-** The Stream API is used to manipulate collections of objects. A stream is a sequence of objects that supports various methods which can be pipelined to produce the desired result.

**Q. What are the benefits of Stream API ?**

**Ans :-** A Stream can be created from Arrays, Collections, files and other IO Sources.  
 Does stream store data / can you store data in the stream ?  
**Ans :-** No → A Stream does not store data , it does only intermediate operation.

**Q. What are the key intermediate operations ?**

**Ans** Intermediate operations are :-

- 1) map () :- It maps a function to Stream and changes elements of the stream.
- 2) filter () :- It filters elements of the stream.
- 3) distinct () :- It removes duplicate elements from the stream.
- 4) sorted () :- It sorts elements of the stream.
- 5) limit () :- It limits an stream to the given maxsize , according to the given parameter.
- 6) skip () :- It skips the given N elements of the stream.

Date : \_\_\_\_\_

**Q. What are the terminal operations ?**

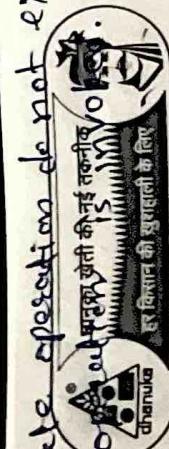
**Ans :-** Terminal operations are :-

- 1) collect () :- It converts an stream into collection .
- 2) to Array () :- It converts a stream into array .
- 3) count () :- It returns the count of the elements in the stream .
- 4) reduce () :- It reduces the elements of the stream according to the given Identity value .
- 5) min () :- It returns the minimum elements of the stream according to the given comparator .
- 6) max () :- It returns the maximum element of the stream according to the given comparator .
- 7) anyMatch () :- It will return any matched element according to the matched predicate . If the stream is empty then return false .
- 8) allMatch () :- It will return all the matched elements from a stream . return true if the stream is empty .
- 9) noneMatch () :- It returns the matched elements of the predicate . return true if the stream is empty .

**IMPORTANT NOTES**

- 10) findAny () :- It returns an optional describing some elements of the stream .
- 11) findFirst () :- It returns the first elements of the stream .

Intermediate operations do not execute until a terminal operation is executed.



Q. When intermediate operations are executed?

Ans:- An intermediate operation transforms one stream into another stream.

Q. What is the function of the filter() method?

Ans:- It filters elements of the stream.

Q. What is the function of the . map() method?

Ans:- It maps a function to a stream and changes elements of the stream.

Q. How do you remove duplicate elements from the stream?

Ans:- By using distinct() method of Stream.

Q. How do you sort elements of the stream?

Ans:- By using sorted() method of Stream

Q. How do you get distinct (remove duplicate elements) of a list in Java?

Ans:- By using the removeDuplicates() method which contains methods to get distinct elements.

IMPORTANT NOTES

① If duplicates are present in the list, then it will remove all the duplicates.

② If duplicates are present in the list, then it will remove all the duplicates.

Q. What are the diff b/w Stream And Collection?

Ans:- Stream and Collection look similar, but there are major differences b/w both.

Stream Collection

- |   |   |
|---|---|
| 1) It does not store data.                | It stores data.   |
| 2) It is read only.                       | It is read or write both.                               |
| 3) It can only be read once.              | It can be read multiple times.                          |
| 4) Elements can not be directly accessed. | Elements can be directly accessed.<br>i.e. list.get(0); |

Q. What is the collection framework?

Ans:- It is a set of interfaces like Collection, List, Set and Map. The Collection is the root interface of the collection.

Q. What are the diff. b/w an Iterator and an Enumeration?

Ans:- Both are the same, read sequentially for the collection, but Iterator also provides the ability to remove element from collection.

- IMPORTANT NOTES
- ① Iterator is a first-in-first-out mechanism.
  - ② Iterator has ability to remove an element from a collection whereas Enumeration cannot.
  - ③ Enumeration available only in standard classes.



Date: \_\_\_\_\_

## Collection

- Q. What is collection framework ?  
Ans:- The Collection framework is a unified architecture for representing and manipulating set of ~~collection~~ elements called Collection.
- Q. What is Collection ?  
Ans:- Collection is a interface of all interface in Collection framework package Java.util. Contains Collections framework classes.

## Key Interface of Collection framework

ArrayList	Queue
Set	Map
SortSet	SortedMap
NavigableSet	NavigableMap

- Q. What is Iterator ?  
Ans:- Iterator is used to read data sequentially from a collection. Collection interface contains iterator() method that return iterator object.

- Q. What is an Enumeration ?  
Ans:- It is just like iterator, it reads sequentially data from collection classes, it is used with Iterator.

IMPORTANT NOTES

- Ans :- If is first data from collection classes, it is used with Iterator.

## What are difference b/w HashTable & HashMap

- | HashTable                            | HashMap                              |
|--------------------------------------|--------------------------------------|
| 1) Synchronized                      | - Asynchronous                       |
| 2) Thread Safe                       | - Not Thread Safe                    |
| 3) Doesn't permit null               | - permits null value as key - value. |
| 4) Slow performance                  | - high performance                   |
| 5) Used in multithreaded application | -                                    |
- Q. In what are diff. b/w ArrayList & Vector

IMPORTANT NOTES



Date :

Difference b/w List &amp; Set

&lt;ArrayList&gt;

&lt;LinkedList&gt;

- 1) ArrayList internally creates a linked list to store elements.
- 2) In ArrayList by default - new element is added at the end of arraylist.
- 3) When we want to insert a new element in ArrayList then new array will be created and the element is copied in new array.
- 4) ArrayList is best in retrieval operation choice in insertion or deletion in middle.
- 5) ArrayList implements Random Access interface.

ArrayList is better than linked list because it does not implement Random Access interface.

ArrayList is best in retrieval operation choice in insertion or deletion in middle.

ArrayList is better than linked list because it does not implement Random Access interface.

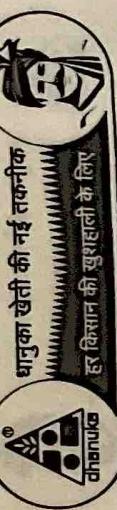
IMPORTANT NOTES

Date :	8. difference b/w List & Set ?
Ans	<List>
1)	List allows duplicates
2)	list preserve insertion order
3)	Set doesn't preserve order.

Date :	Q. What is concurrent Collection ? Ans:- Adding or deleting of elements after creating it will not cause Concurrent modification exception.
Ans	1) Blocking Queue
2)	Concurrent Map
3)	Concurrent Navigable Map
4)	Concurrent HashMap
5)	Concurrent SkipList Map.

Date :	Q. What is a list interface ? Ans:- List interface is a child interface of a collection interface. The list interface extends the collection interface. The list interface declares a method for managing an ordered collection of objects.
Ans	Duplicate elements and null values can be inserted into a list.
1)	add(i, o)
2)	get(i)
3)	remove(i)
4)	set(i, o)
5)	indexof(i)
6)	lastIndexOf(i)

IMPORTANT NOTES



- Date : \_\_\_\_\_
- Q. What is a Deque? Interface?
- A. It is the deque interface of Queue interface. It represents a queue where you can insert and remove elements from both end of the queue. Thus Deque stores the double ended queue!
- Q. What is a map interface?
- A. → The map interface does not extends the collection interface that is why it is not called true collection. It is an object that maps key with values.
- A map contains unique keys. Each key is mapped with its value. Two different keys may have same values.
- Q. What is Set interface?
- A. The Set interface is child of set interface. Its elements are automatically ordered. Additional method in this interface are :-
- ① `add()`
  - ② `size()`
  - ③ `headSet()`
  - ④ `tailSet()`
- Q. What is a Queue interface?
- A. The Queue interface contains queue of elements. Besides basic collection operation it provides addition operation like insertion, extraction and inspection. Queue can implement Fifo sub class of Queue and -
- ① Priority Queue
  - ② Linked List
- IMPORTANT NOTES :-  
Methods of Queue interface are :-
- ① `add()`
  - ② `remove()`
  - ③ `poll()`
  - ④ `offer()`
- Q. What is Sorted Map Interface?
- A. → It facilitates a map to be ordered according to natural ordering of keys.

- Date : \_\_\_\_\_
- Q. What is Set interface?
- A. The Set interface is child of set interface. Its elements are automatically ordered. Additional method in this interface are :-
- ① `add()`
  - ② `size()`
  - ③ `headSet()`
  - ④ `tailSet()`
- Q. What is a Queue interface?
- A. The Queue interface contains queue of elements. Besides basic collection operation it provides addition operation like insertion, extraction and inspection. Queue can implement Fifo sub class of Queue and -
- ① Priority Queue
  - ② Linked List
- IMPORTANT NOTES :-  
Methods of Queue interface are :-
- ① `add()`
  - ② `remove()`
  - ③ `poll()`
  - ④ `offer()`
- Q. What is Hashmap?
- A. Hashmap is a map interface which stores key-value pairs. It uses HashTable to store data. HashTable is a collection of Hashmap.



Date :

a list design pattern is followed by the

Iterator interface?  
Ans: Iterator design pattern.

Q. What are differences b/w Vector and  
ArrayList?  
Ans:

<ArrayList> <Vector>

- |   |                                  |   |
|---|----------------------------------|---|
| ① | Vector is a synchronized         | ArrayList is a non-synchronized.                    |
| ② | Vector grows double of its size. | ArrayList grows half of its size.                   |
| ③ | Vector is a Thread safe.         | Not Thread safe.                                    |
| ④ | Vector is a synchronized         | ArrayList high performance.<br>So less performance. |
| ⑤ | Used for multithread system.     | Used for single thread system.                      |

Q. What type of object compared by Comparable & compareTo()?  
Ans: Object & Comparable.

IMPORTANT NOTES

- ① No. to → Plain. Old Java Object
- ② DTD → Data Transfer Object
- ③ VO → Value Object.

Date :

Q. What is java.util package?  
Ans: It is package that contains all utility classes and interfaces like Vector,  
ArrayList, Hashmap, HashTable,  
HashSet, List & Set, map etc.

Q. Differences between HashSet & TreeSet?

- |   |  |
|---|--|
| Ans: → <HashSet> -                      | <TreeSet>  |
| 1) HashSet does not maintain any order. | TreeSet is sorts the elements in the asc. order. |
| 2) Contains only one null value.        | not contain null value.                          |
| 3)                                      |  |

Q. Difference between Comparable & Comparator?  
Ans: → <Comparable> -

- |   |                    |
|---|--------------------|
| 1) Comparable is used in Java programming package - | java.util package. |
| 2) Comparable method use -                          | compare() -        |

IMPORTANT NOTES

- ③ Comparable compare -
- Comparable compare -
- Comparable to same -
- for different object -
- in a & no other class -



Date :

Q. What is HashSet & TreeSet?  
Ans: HashSet & TreeSet are two collection classes which implement Set interface.

Q. Differences between HashSet & TreeSet?

- |   |  |
|---|--|
| Ans: → <HashSet> -                      | <TreeSet>  |
| 1) HashSet does not maintain any order. | TreeSet is sorts the elements in the asc. order. |
| 2) Contains only one null value.        | not contain null value.                          |
| 3)                                      |  |

Q. Difference between Comparable & Comparator?  
Ans: → <Comparable> -

- |   |                    |
|---|--------------------|
| 1) Comparable is used in Java programming package - | java.util package. |
| 2) Comparable method use -                          | compare() -        |

IMPORTANT NOTES

- ③ Comparable compare -
- Comparable compare -
- Comparable to same -
- for different object -
- in a & no other class -