Java Interview Question and Answers

➢ Topic – Set :

Set Interview Questions with Answers

---

1. What is a Set in Java?
Answer: A set in Java is an unordered collection of unique elements or objects. It does not store elements in a particular order.

2. Is Set interface or class in Java?
Answer: Set is a generic interface in Java. This interface does not provide any additional methods of its own. It uses methods defined by the collection interface, but places additional restrictions on those methods.

3. Name three general-purpose set implementation classes in Java.
Answer: Three general purpose set implementation classes are HashSet, LinkedHashSet, and TreeSet.

4. Which java.util.Set implementation is the fastest one: Hashset, LinkedHashSet, and TreeSet?
Answer: HashSet is faster as compared to LinkedHashSet and TreeSet.

---

5. Which java.util.Set implementation does not guarantee the order of elements?
Answer: HashSet

6. What are the important features of Set in Java?
Answer: The following are the important features of Set in Java.

(a) Set is an unordered collection of elements. It does not maintain a particular order while storing elements. While retrieving, we may not get the same order as that we put elements.

(b) It contains only unique elements. It does not store duplicate elements.

(c) Java Set uses a map based structure for its implementation.

(d) It can be iterated by using Iterator but cannot be iterated by using ListIterator.

(e) Most of the set implementations allow adding only one null element. Tree set does not allow to add null element.

(f) Set is not an indexed-based structure like a list in Java. Therefore, we cannot access elements by their index position.

(g) It does not provide any get method like a list.

---

7. How to create a generic Set object in Java?

Answer: We can create a generic Set object by using one of its three concrete classes: HashSet, LinkedHashSet, and TreeSet. The general syntax to create set objects is as:

```
Set<T> set = new HashSet<T>(); where T is a type of generic.

Set<T> set = new LinkedHashSet<T>();

Set<T> set = new TreeSet<T>();



For example:

Set<Integer> set = new HashSet<Integer>(); // Creates an empty set of Integer objects.

Set<String> set2 = new LinkedHashSet<String>(); // Creates an empty set of String
objects.
```

8. What is the difference between List and Set in Java?

Answer: Major differences between a List and a Set are:

(a) List is an ordered collection of elements whereas a Set is just a distinct collection of elements that is unordered.

(b) List can have duplicate elements while Set contains only unique elements.

9. How will you convert a List to a Set in Java?

Answer: There are two ways to convert a List to a Set in Java.

Method 1: Use HashSet

```
Set<Integer> mySet = new HashSet<Integer>(myList);
```

Here, we are putting a list into a HashSet. Internally hashCode() method is used to identify duplicate objects or elements.

Method 2: Use TreeSet

```
Set<Integer> mySet = new TreeSet<Integer>(myComparator);

mySet.addAll(myList);
```

Here, we are using the comparator to find duplicate objects.

---

10. How will you remove duplicate elements from an ArrayList?

Answer: To remove duplicate elements from an ArrayList, we will a Set. There are two options for this purpose.

Option 1:

Use Set if ordering of elements is not important. We will just put the elements of ArrayList in a HashSet and then add them back to the ArrayList. The following code is an example.

```
List<Integer> list = new ArrayList<Integer>();

list.add(5);

list.add(10);



Set<Integer> s = new HashSet<Integer>(list);

list.clear();

list.addAll(s);
```

Option 2:

Use LinkedHashSet if ordering of elements is important. Then we will put the elements of ArrayList in a LinkedHashSet and then add them back to the ArrayList. The sample code is as:

```
List<Integer> list = new ArrayList<Integer>();

list.add(5);

list.add(15);



Set<Integer> s = new LinkedHashSet<Integer>(list);

list.clear();

list.addAll(s);
```

11. What are the Collection classes that implement Set interface in Java?

Answer: Following are the Collection classes that implement Set interface:

- AbstractSet
- ConcurrentSkipListSet
- CopyOnWriteArraySet
- EnumSet
- HashSet
- JobStateReasons
- LinkedHashSet
- TreeSet

12. Suppose you have a Collection called "c" that contains duplicate elements. How will you remove those elements using Set?

Answer: Assume that "c" is a collection of type String and contains duplicate elements.

```
Collection<String> noDuplicates = new HashSet<String>(c);
```

13. Suppose you have a Collection called "c" that contains duplicate elements. How will you remove those elements using Set but must be in the order?

Answer: Assume that “c” is a collection of type String and contains duplicate elements.

```
Collection<String> noDuplicates = new LinkedHashSet<String>(c);
```

14. How can you get the number of elements present in a Set?
Answer: The java.util.Set.size() method returns the number of elements in a Set. It is used to get the size of the Set. The sample code is as:

```
// Creating an empty Set.

  Set<String> set = new HashSet<String>();

// Call add() method to add elements into the Set.

  set.add("Banana");

  set.add("Orange");

  set.add("Apple");

// Printing the size of the Set.

  System.out.println("The size of set is: " + set.size());
```

15. What happens if we attempt to add an element that already exists in the Set?
Answer: As the duplicate elements are not allowed in the Set, adding an element that already exists in Set will ignore.

16. What happens if we try to remove an element from a Set that does not exist in the Set?
Answer: The remove() method of Set will return a false value. It will not throw any exception.

17. Name the java.util.Set implementation class that maintains the order of elements based on their insertion order.
Answer: LinkedHashSet

18. How will you check whether an element is present in the Set?
Or **How to search an element in a Set?**
Answer: The contains() method checks whether an element is present in the set. If it is present in the set, the method returns true, otherwise false.

19. Suppose you have two sets named set1 and set2. You want to perform a union operation in order to add set2 into set1. Which bulk operation would you to prefer?

Answer: addAll operation will serve for this purpose. For example:
set1.addAll(set2);

20. Suppose you have two sets named set1 and set2. You want to remove all elements of set2 from set1. Which bulk operation would you to prefer?

Answer: removeAll operation will serve for this purpose. For example:
set1.removeAll(set2);

21. What will be the output of the following code?

```
public class SetEx {

public static void main(String[] args)

{

 List list = new ArrayList();

 list.add(null);

 list.add("AA");

 list.add("BB");

 list.add("bb");

 list.add(null);

Set set = new HashSet<>(list);

System.out.println(set);

  }

}
```

Output: [null, AA, BB, bb]

22. What will the output of the following program if there is no error?

```
public class SetEx {

public static void main(String[] args)

{

 Set set = new HashSet<>();

 set.add("C");

 set.add("B");

 set.add("A");

 set.add(null);

  set = new LinkedHashSet(set);

  System.out.println(set);

  }

}
```

Output: [null, A, B, C]

23. When to use Set?

Answer: To understand the best answer with an example, go to this tutorial: *Set in Java*.

24. What are the ways to iterate a Set in Java?

Answer: There are mainly three ways to iterate a set in Java. They are as:

- Using Iterator
- Using Enhanced for loop
- Using forEach()

Go to this tutorial for example code: *How to iterate Set in Java*

25. How can you synchronize the elements of a Set in Java?

Answer: Use synchronizedSet() method that returns the synchronized (thread safe) set backed by the specified Set.

HashSet Interview Questions Answers

---

26. What is a HashSet in Java?
Answer: HashSet is an unordered collection of elements (objects) that contains only unique elements.

27. What are the important features of HashSet?
Answer: The important features of HashSet are as:

(a) The underlying data structure of HashSet is Hashtable. A hash table stores data by using a mechanism called hashing.

(b) HashSet does not allow duplicate elements.

(c) It allows only one null element.

(d) It does not maintain any order in which elements are added.

(e) It is much faster because of the use of hashing technique and gives constant-time performance for adding (insertion), retrieval, removal, contains, and size operations.

(f) HashSet class is not synchronized, which means it is not thread-safe. If you want to synchronize HashSet, use Collections.synchronizeSet() method.

(g) The iterator returned by HashSet class is fail-fast which means any modification that happened in the HashSet during iteration will throw ConcurrentModificationException.

---

28. How java.util.HashSet stores its elements?
Answer: HashSet internally uses a HashMap to store the elements and to maintain the uniqueness of elements. When we create a HashSet object, a corresponding HashMap object is also created.

When we add an element in HashSet, it adds it into the corresponding HashMap.

29. What is the initial default capacity of HashSet?
Answer: The initial default capacity for HashSet is 16.

30. What is the HashSet default load factor?

Answer: The default load factor for HashSet is 0.75.

31. How to create a user-defined object of HashSet?
Answer: Go to this tutorial to get the best answer: *HashSet in Java*.

32. When to use HashSet in Java?
Answer: We can use HashSet in Java when

- we don't want to store duplicate elements.
- we want to remove duplicate elements from the list.
- HashSet is more preferred when add and remove operations are more as compared to get operations.
- we are not working in a multi-threading environment.

LinkedHashSet Interview Questions

33. What is LinkedHashSet in Java?
Answer: LinkedHashSet is an ordered collection of elements that contains only unique elements. It internally uses a linked list to store the elements in the set.

34. What is the difference between HashSet and LinkedHashSet in Java?
Answer: HashSet does not preserve the insertion order of elements, while LinkedHashSet preserves the insertion order of elements. LinkedHashSet is slightly slower than HashSet.

35. When to use LinkedHashSet?
Answer: You can use LinkedHashSet when you do not want duplicate elements (i.e. want to remove duplicate elements) and want to maintain order in which elements are inserted.

36. Which is better to use: HashSet or LinkedHashSet?
Answer: If you do not require maintaining order in which elements are inserted, then use HashSet that is more fast and efficient than LinkedHashSet.

In this tutorial, we have listed the top 25 **Set interview questions in Java** with the best possible answers. We have also covered some coding questions with output from which the interviewer can ask from freshers and experienced.

Hope that you will have understood all answers of these set questions in Java and practiced them.

All the best!!!