Java HashMap Interview Questions

1. **How does put() method of HashMap work in Java?**
   On Hashing Principle
   Put() will use Key to generate HashCode by HashCode() method. Then, the hashcode will be used to generate an index to store the Key+HashCode+Value+NextCode into HashMap segment.
2. **What is the requirement for an object to be used as Key or Value in HashMap?**
   Both key and value should implement hashCode() and equals() methods.
   When put(), key will be used to generate hashcode by hashCode() and generate index later to store the key+value pair.
   When get(), key will also be used to get hashcode and index. equal() will be used to find matched key by index and get value eventually.
3. **What will happen if you store a key which is already present in the HashMap?**
   No duplicate key can exist. So the old value will be overridden by the new value.
4. **Can you store a null key in Java HashMap?**
   Yes, but only one null key is allowed.
5. **Can you store a null value in Java HashMap?**
   Yes.
6. **How does HashMap handle Collision in Java?**
   Refer to the diagram.
7. **Which data structure HashMap represents?**

8. **What data Structure is used to implement HashMap?**

9. **Can you store a duplicate key in HashMap?**
   No
10. **Can you store a duplicate value in HashMap?**
    Yes

Java HashMap Interview Questions (Continued)

11. **Is HashMap thread-safe in Java?**
    No.
12. **What will happen if you use a multi-threaded Java application?**
    Since HashMap is not thread-safe in Java, several thread could modify same HashMap at the same time. Some thread could not get the up-to-date data. Data cannot be synchronized.
13. **What are different ways to iterate hashmap?**
    keySet and Iterator
    entrySet and Iterator
14. **How do you remove a mapping when iterating over HashMap?**
    Use remove() method
15. **In what kind of order is mappings stored in HashMap?**
    Random Order.
16. **Can you sort HashMap in Java?**
    No.
    You can convert it into LinkedHashMap or SortedHashMap
17. **What is the load factor in HashMap?**
    Load factor is a number which controls when to resize the HashMap. HashMap's default size is 16. If the load factor is 0.75, that means when the HashMap is got 75% full, the Resizing will be triggered to involve array copy.
18. **How does resizing happen in HashMap?**
    Same as above
19. **How many entries can you store in HashMap? What's the maximum limit?**
    No limit.
20. **What's the difference between capacity and size of the map?**
    Capacity means how many entries HashMap can store. The size of map represents how many key & value pair has been stored.

```
public static void main(String[] args) {
    Map<String, Integer> testMap = new HashMap<String, Integer>();
    testMap.put("A", 100);
    testMap.put("B", 200);
    testMap.put("C", 300);
    testMap.put("D", 400);


    Iterator<String> keyIt = testMap.keySet().iterator();
    while (keyIt.hasNext()) {
        String key = keyIt.next();
        System.out.println(testMap.get(key));
    }

    Iterator<Entry<String, Integer>> entryIt = testMap.entrySet().iterator();
    while (entryIt.hasNext()) {
        Entry<String, Integer> mapEntry = entryIt.next();
        System.out.println(mapEntry.getKey() + " " + mapEntry.getValue());
    }

    for (String key: testMap.keySet()) {
        System.out.println(testMap.get(key));
    }

    for (Entry<String, Integer> mapEntry: testMap.entrySet() ) {
        System.out.println(mapEntry.getKey() + " " + mapEntry.getValue());
    }

}
```

Java HashMap Interview Questions (Continued)

21. **What will happen if two different keys return same HashCode?**
    Collision occurs. They will get same index and these two entries will be stored under same segment one after another, with next node connected, like a linkedList.
    When the threshold_value reaches 8, after JDK 1.8, if there is the 9th one, the whole linkedList will be converted to a Balanced Binary Tree.