**HashMap vs ConcurrentHashMap**

**1.  Thread -Safe :**

|  |  |
| --- | --- |
| **HashMap** | **Concurrent HashMap** |
| while HashMap is **not thread-safe** . | ConcurrentHashMap **is thread-safe** that is the code can be accessed by single thread at a time . |
| HashMap can be synchronized by using synchronizedMap(HashMap)  method .  By using this method we get a HashMap object which is equivalent to the HashTable object . So every modification  is performed on  Map is locked on Map object. | ConcurrentHashMap synchronizes or locks on the certain portion of the Map . To optimize the performance of ConcurrentHashMap , Map is divided into different partitions depending upon the Concurrency level . So that we do not need to synchronize the whole Map Object. |
| In HashMap there can only be one null key . | ConcurrentHashMap does not allow NULL values . So the key can not be null in ConcurrentHashMap |
| While in HashMap any number of threads can access the code at the same time . | In multiple threaded environment HashMap is usually faster than ConcurrentHashMap . As only single thread can access the certain portion of the Map and thus reducing the performance . |