|  |  |  |
| --- | --- | --- |
| **ConcurrentHashMap** | **synchronizedMap** | **Hashtable** |
| We will get ThreadSafety without locking TotalMap object just with Bucket level lock. | We will get Threadsafety by locking whole Map object. | We will get Threadsafety by locking whole Map object. |
| At a time multiple threads are allowed to operate on Map object is safe manner.6 | At a time only one thread is allowed to perform any operation on Map object. | At a time only one thread is allowed to perform any operation on Map object. |
| Read operation can be performed without lock but write operation can be performed with bucket level lock. | Every read and write operations require total map object lock. | Every read and write operations require total map object lock. |
| While one thread iterating Map object, the other threads are allowed to modify map and we won’t get ConcurrentModificationException. | While one thread iterating map object, the other threads are not allowed to modify map. Otherwise we will get ConcurrentModificationException | While one thread iterating map object, the other threads are not allowed to modify map. Otherwise we will get ConcurrentModificationException |
| Iterator of ConcurrentHashMap is Fail-safe and won’t raise ConcurrentModificationException | Iterator of synchronizedMap is Fail-Fast and it will raise ConcurrentModificationException. | Iterator of synchronizedMap is Fail-Fast and it will raise ConcurrentModificationException. |
| Null is not allowed for both keys and values. | Null is allowed for both keys and values. | Null is not allowed for both keys and values. |
| Introduced in 1.5 version | Introduced in 1.2 version | Introduced in 1.0 version |