**HashMap (Not Thread-Safe)**

* Allows **one null key** and multiple null values.
* **Not synchronized**, meaning multiple threads modifying it **may cause race conditions**.
* Requires **explicit synchronization** for thread safety (e.g., using Collections.synchronizedMap()).
* Faster in a **single-threaded** environment.

**2 Hashtable (Thread-Safe but Slow)**

* **Synchronized on every method**, meaning only **one thread can access it at a time**.
* **No null keys or null values** allowed.
* Slower because of **full map-level locking**, even for reads.

**3 ConcurrentHashMap (Thread-Safe and Fast)**

* **Uses bucket-level locks (Java 8+), improving concurrency.**

Only 1 bucket locked per write operation, so e.g. **if you insert 4 unique elements**, and they are mapped to **4 different buckets**, then **4 separate buckets will be locked individually during each write operation**.

* **Multiple threads can read and write simultaneously** (unlike Hashtable).
* **Does not allow null keys or null values.**
* **Faster than Hashtable in multi-threaded environments.**