**Q) What is fail-safe and fail-fast Iterator in Java ?**

|  |  |
| --- | --- |
| **Fail-fast** | **Fail-safe** |
| While iterating collection items if any modification is done, then **ConcurrentModificationException**is thrown  This is said to be **fail-fast** | While iterating collection items if any modification is done and if **ConcurrentModificationException**is never thrown, then it is said to be **fail-safe** |
| Generally, Collection classes introduced in **Java 1.2 version**like ArrayList or HashSet falls under this category | New concurrent classes introduced in **Java 1.5 version** is fail-safe and never throws ConcurrentModificationException |
| Here, there is **no concept of cloned copy**. Hence, both iteration & modification happening in the same original copy leading to throwing of ConcurrentModificationException | This is because, modification happens in a **separate cloned copy & later JVM merges** both original with cloned copies |
| **Example:** ArrayList, LinkedList, HashSet, TreeSet | **Example:** CopyOnWriteArrayList, CopyOnWriteArraySet, ConcurrentHashMap |
| Above listed classes comes from **java.util**package | Above listed classes comes from **java.util.concurrent**package |