**Why map doesn’t belong to collection?**

If you look at the respective data structure you can easily guess why Map is not a part of Collection. Each Collection stores a single value where as a Map stores key-value pair. So methods in Collection interface are incompatible for Map interface.For example in Collection we have add(Object o). What would be such implementation in Map. It doesn't make sense to have such a method in Map. Instead we have a put(key,value)  method in Map.

Same argument goes for addAll(), remove(), removeAll() methods. So the main reason is the difference in the way data is stored in Map and Collections.

Also if you recall Collection interface implemented Iterable interface i.e any interface with .iterator() method should return an iterator which must allow us to iterate over the values stored in the Collection. Now what would such method return for a Map? Key iterator or a Value iterator? This does not make sense either.

There are ways in which we can iterate over keys and values stores in a Map and that is how it is a part of Collection framework.Do not get confuse here. Let me repeat this - Map is a part of Collection framework but it does not implement Collection interface. We will see in next few posts how Map works internally(that's really an interesting topic) and also how we manipulate data in it.

**Methods in map**

[1. void clear](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html#clear())()

Removes all of the mappings from this map (optional operation).

2. boolean [containsKey](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html#containsKey(java.lang.Object))([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html) key)

Returns true if this map contains a mapping for the specified key.

3. boolean [containsValue](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html#containsValue(java.lang.Object))([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html) value)

Returns true if this map maps one or more keys to the specified value.

4. [Set](https://docs.oracle.com/javase/7/docs/api/java/util/Set.html)<[Map.Entry](https://docs.oracle.com/javase/7/docs/api/java/util/Map.Entry.html)<[K](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html),[V](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html)>> entrySet()

Returns a [Set](https://docs.oracle.com/javase/7/docs/api/java/util/Set.html) view of the mappings contained in this map.

5. boolean [equals](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html#equals(java.lang.Object))([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html) o)

Compares the specified object with this map for equality.

6. V  [get](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html#get(java.lang.Object))([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html) key)

Returns the value to which the specified key is mapped, or null if this map contains no mapping for the key.

7. int hashCode()

Returns the hash code value for this map.

8. boolean isEmpty()

Returns true if this map contains no key-value mappings.

9. Set<K> keyset()

Returns a [Set](https://docs.oracle.com/javase/7/docs/api/java/util/Set.html) view of the keys contained in this map.

10, V [put](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html#put(K,%20V))([K](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html) key, [V](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html) value)

Associates the specified value with the specified key in this map (optional operation).

11. void [putAll](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html#putAll(java.util.Map))([Map](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html)<? extends [K](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html),? extends [V](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html)> m)

Copies all of the mappings from the specified map to this map (optional operation).

12. V [remove](https://docs.oracle.com/javase/7/docs/api/java/util/Map.html#remove(java.lang.Object))([Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html) key)

Removes the mapping for a key from this map if it is present (optional operation).

13. int size()

Returns the number of key-value mappings in this map.

14. Collection<V> values()

Returns a [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html) view of the values contained in this map.