Unmodifiable Collections are NOT immutable collections

It is very important to understand that unmodifiable collections are NOT immutable collections.

They become immutable collections, if the elements in the collections themselves are fully immutable.

They are collections with limited functionality that can help us minimize mutability.

- You can't remove, add or clear elements from an immutable collection.
- You also can't replace or sort elements.
- Mutator methods will throw an UnsupportedOperationException.
- You can't create this type of collection with nulls.

Unmodifiable Collections vs. Unmodifiable Collection Views

The three primary Collection interfaces, List, Set or Map, have methods to get an unmodifiable copy on the specific interface, related to the collection type, as shown,

In addition, the java.util.Collections class offers methods, to get unmodifiable views as shown.

These methods allow us to get closer to the ideal of immutability, if it's needed.

	Unmodifiable Copy of Collection	**Unmodifiable View of Collection**
List	List.copyOf List.of	Collections.unmodifiableList
Set	Set.copyOf Set.of	Collections.unmodifiableSet Collections.unmodifiableNavigableSet Collections.unmodifiableSortedSet
Map	Map.copyOf Map.entry(K k, V v) Map.of Map.ofEntries	Collections.unmodifiableMap Collections.unmodifiableNavigableMap Collections.unmodifiableSortableMap