**✅ Difference between Collection and Collections in Java**

| **Feature** | **Collection** | **Collections** |
| --- | --- | --- |
| 📘 Type | **Interface** | **Utility class** |
| 🧠 Defined in | java.util.Collection | java.util.Collections |
| 📦 Purpose | Root interface of all **collections** | Contains **static utility methods** |
| 👨‍👩‍👧‍👦 Examples | List, Set, Queue all extend it | sort(), reverse(), synchronizedList() |
| 🛠️ Usage | To **define collection types** | To **operate on collections** |

**📌 1. Collection (Interface)**

This is the **parent interface** for all collection types like List, Set, Queue.

java

CopyEdit

Collection<String> names = new ArrayList<>();

names.add("Alice");

names.add("Bob");

Think of it as the **blueprint**.

**📌 2. Collections (Utility Class)**

This is a **helper class** with **static methods** for common operations.

java

CopyEdit

List<String> list = new ArrayList<>();

list.add("Zebra");

list.add("Apple");

Collections.sort(list); // ✅ sorts the list

Collections.reverse(list); // ✅ reverses it

It makes working with collections easier.

**🧾 Summary:**

| **Concept** | **Collection** | **Collections** |
| --- | --- | --- |
| Is a… | Interface | Class (final, with static methods) |
| Part of… | Collection hierarchy | Utility/helper functions |
| Use for… | Creating data structures | Performing actions (sort, sync, etc) |