Las interfases Collection y Map son las padres de las interfases en el Framework de colecciones de Java. Es notorio que la jerarquía consta de dos diferentes árboles.

**Collection:** Es la raíz de la jerarquía. Representa un grupo de objetos. Esta interfaz es el último denominador común que todas las colecciones implementan y es usada cuando se desea manipularlas con el máximo de generalidad deseado..

**Set:** una colección que no puede tener elementos duplicados. Un ejemplo son las cartas de un naipe. .

**List**: Una colección ordenada. Las listas pueden contener elementos duplicados. Generalmente se puede tener control sobre donde cada elemento es insertado y se puede acceder a cada elemento por su índice (posición).

**Queue:** Una colección utilizada para guardar varios elementos por prioridad. Provee operaciones adicionales de inserción, extracción e inspección. Cada implementación de Queue debe especificar sus propiedades de ordenamiento.

**Map:** Un objeto que trabaja por parejas siendo sus componentes clave y valor. Map no permite tener claves duplicadas, cada clave debe corresponder al menos a un valor.

**SortedSet:** Mantiene sus elementos en un orden ascendente.

**SortedMap:** Mantiene sus mapeos en un orden ascendente por clave.

**Listas**

La colección más básica de Java. También, es la más usada por los programadores que no han investigado el framework de colecciones a fondo por hacernos pensar que se trata de una especie de array hipervitaminado ya que hace su trabajo y es fácil de entender.

**Los sets**, o conjuntos, son colecciones que por norma general no admiten elementos iguales en su interior. Como mencionaba antes, dos elementos A y B son iguales si A.equals(B).

Podemos añadir y eliminar elementos de un set, así como saber si determinado elemento está en un set, pero no tenemos acceso aleatorio, por lo que hay que tener muy claro cuando queremos usarlos.

**Las colas** son estructuras que ofrecen un gran rendimiento al obtener elementos de su principio o de su final, representando colas LIFO / FIFO, aunque también veremos colas ordenadas en función de otros criterios.

Deberás usar una cola cuando vayas a recuperar siempre el primer o último elemento de una serie. Se usan para implementar las mencionadas colas LIFO / FIFO, así como colas de prioridades (como puede ser un sistema de tareas o de procesos).

Cabe destacar que hay dos tipos de colas, Queues y Deques. Las primeras solo proporcionan métodos para acceder al último elemento de la cola, mientras que las Deques permiten acceder a cualquiera de los dos extremos.