

**Individual elements:** sin key.

**Key-Value pairs:** dos valores, key y value <K,V>

**Primitive support:** byte, hort, int, long…

**FIFO:** first in first out

**O(1):**  rápido sin importar el tamaño.

**O(log(n)):** rápido, si es muy grande es lento.

**O(n):**  depende del tamaño.

**ArrayList :** List<Object> name = new ArrayList<>();

individual, permite duplicados, FIFO, LIFO, random Access by index

**HashSet:** Set<Object> name =new HashSet<>();

Individual, funciona “contains”, random Access by value

**HashMap:** Map<Key, Valor> name = new HashMap<>();

Key-value, contains, random Access by key

**TreeMap:** Map<> name = new TreeMap<>();

Control duplicados, sorted, contains, random Access by key

**SPRING**

**@Entity : especifica que se va a crear una entidad.**

**@Id : indica el id.**

**@GeneratedValue(strategy = GenerationType.AUTO)** : JPA autogenera

**@ManyToOne : varios a uno (Foreign keys)**

**:name : paramentro**

**@Service : la clase es un bean**

**@Autowired : el service lo necesita**

**@Repository : indica que es un objeto de Acceso a datos**