

COMPARACIÓN

COLECCIONES

COMPARACIÓN EN JAVA

- Java posee diferentes operadores y métodos de comparación
 - `>`, `>=`, `<`, `<=`, `==`, `!=`
 - `equals`
- Para poder comparar, los tipos de datos deben ser “compatibles”
- Los tipos primitivos tienen orden natural: números, letras, ...
- ¿Qué pasa con nuestras clases?
 - Personas, Productos, Facturas, ...

INTERFACES: COMPARABLE Y COMPARATOR

- *Comparable*
 - permitirá indicar la forma principal de comparación entre objetos de un tipo.
 - indispensable para poder ordenar.
- *Comparator*
 - permite indicar otras formas secundarias de comparación.
- Ambas utilizan genéricos

COMPARABLE

- Interfaz funcional (lambda)
- Un único método: *compareTo(other)*:
 - Devuelve un entero
 - Menor que 0, si el objeto actual es menor que *other*
 - 0, si ambos son iguales
 - Mayor que 0, si el objeto actual es mayor que *other*
- *Bound* de aquellas colecciones genéricas que tienen orden

COMPARATOR

- Interfaz funcional similar a Comparable.
- Un método: *compare(object1, object2)*.
- Permite proporcionar una forma secundaria de comparación / ordenación.