If any interface contains only one method, then it is called as functional interfaces.

Examples Comparable and Comparator.

<mark>Iterator vs ListIterator</mark>

Iterator is used for forward only traversal, only once.

Once you reach to end of collection you cannot tranverse back, hence, to traverse in both directions we use ListIterator

Comparable and Comparator

- These are functional interfaces.
- Both the interfaces are used for sorting user defined class objects
- If you want to sort product object on name by default, then product class should implement comparable interface and override compareTo(Product p) method
- If you want to sort based on some other field then use Comparator, for Comparator write separate class and implement Comparator interface and override compare(Product p1, Product p2)
  The logic in both methods should return integer either -ve or 0 or +ve.