# Comparator and Comparable

Java provides two interfaces to sort objects using data members of the class.

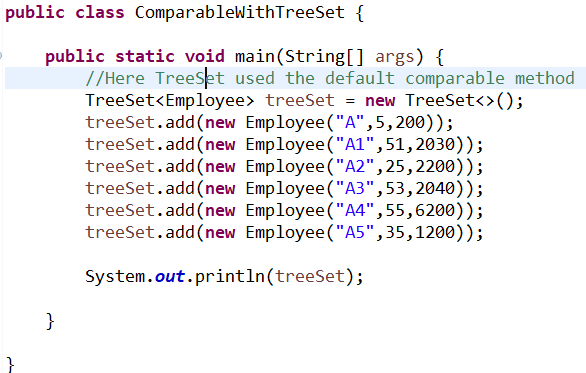
## Comparable

A comparable object is able to compare itself with another object. For one model class, it can compare only on basis of one attribute or field.

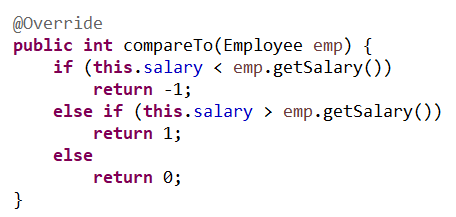
In compareTo method we pass only one argument and it compares with the this reference of the class

### How to use

* The model class first implements java.lang.Comparable interface to compare its instances.
* Then we can override the compareTo () method in that class.



CompareTo method in Employee class

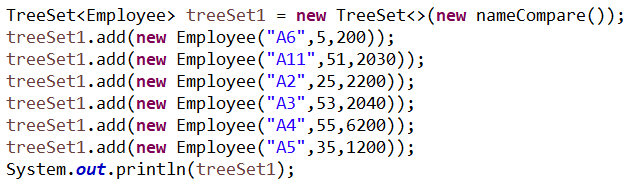


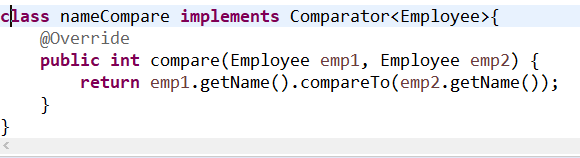
## Comparator

In above example, we sorted using salary. Suppose we want some other treeSet but sorted on the basis of name instead of salary. Here Comparator comes into picture. Unlike Comparable, Comparator is external to the element type we are comparing. It’s a separate class. We create multiple separate classes (that implement Comparator) to compare by different members.

### How to use

* Create a class that implements Comparator (and thus the compare () method that does the work previously done by compare ()).
* Make an instance of the Comparator class.
* Pass the instance to the treeSet.

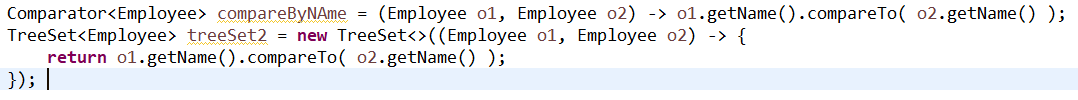




## Implementation for Comparable and comparator

/General/src/com/gurman/comparator/and/comparableprac/ComparableWithTreeSet.java

## Java 8 Comparator



## Comparator that compares based on two attributes.

