Difference between Comparator and Comparable in Java

Difference between Comparator and Comparable in java

One of the common interview question is "What are differences between Comparator and Comparable". or "How will you sort collection of employee objects by its id or name". For that we can use two interfaces, i.e., `Comparator` and `Comparable`. Before we actually see differences,let me give you brief introduction of both.

Comparable interface

Class whose objects to be sorted must implement this interface. In this, we have to implement `compareTo(Object)` method. For example:

Hide Copy Code

```
public class Country implements Comparable<Country>{
      @Override
   public int compareTo(Country country) {
       return (this.countryId < country.countryId ) ? -1: (this.countryId > country.countryId ) ? 1:0 ;
}}
```

If any class implements comparable inteface then collection of that object can be sorted automatically using `Collection.sort()` or `Arrays.sort().Object` will be sort on the basis of `compareTo` method in that class. Objects which implement `Comparable` in Java can be used as keys in a `SortedMap` like `TreeMap` or `SortedSet` like `TreeSet` without implementing any other interface.

Comparator interface

The class whose objects to be sorted do not need to implement this interface. Some third class can implement this interface to sort. E.g., `CountrySortByIdComparator` class can implement `Comparator` interface to sort collection of country object by id. For example:

Hide Copy Code

```
public class CountrySortByIdComparator implements Comparator<Country>{

   @Override
   public int compare(Country country1, Country country2) {

      return (country1.getCountryId() < country2.getCountryId() ) ? -1:
        (country1.getCountryId() > country2.getCountryId() ) ? 1:0 ;
   }
}
```

Using `Comparator` interface, we can write different sorting based on different attributes of objects to be sorted. You can use anonymous comparator to compare at particular line of code. For example:

Hide Copy Code

```
Country indiaCountry=new Country(1, "India");
Country chinaCountry=new Country(4, "China");
Country nepalCountry=new Country(3, "Nepal");
Country bhutanCountry=new Country(2, "Bhutan");

List<Country> listOfCountries = new ArrayList<Country>();
listOfCountries.add(indiaCountry);
listOfCountries.add(chinaCountry);
listOfCountries.add(nepalCountry);
listOfCountries.add(bhutanCountry);



Collections.sort(listOfCountries,new Comparator<Country>() {

   @Override
   public int compare(Country o1, Country o2) {

      return o1.getCountryName().compareTo(o2.getCountryName());
   }
});
```

Comparator vs Comparable

Package

Sorting logic	Sorting logic must be in same class whose objects are being sorted. Hence this is called natural ordering of objects	Sorting logic is in separate class. Hence we can write different sorting based on different attributes of objects to be sorted. E.g. Sorting using id,name etc.
Implementation	Class whose objects to be sorted must implement this interface. e.g `Country` class needs to implement comparable to collection of country object by id	Class whose objects to be sorted do not need to implement this interface. Some other class can implement this interface. E.g.- `CountrySortByIdComparator` class can implement `Comparator` interface to sort collection of country object by id
Sorting method	`int compareTo(Object o1)` This method compares this object with `o1` object and returns a integer. Its value has following meaning 1. positive – this object is greater than `o1` 2. zero – this object equals to `o1` 3. negative – this object is less than `o1`	`int compare(Object o1,Object o2)` This method compares `o1` and `o2` objects. and returns a integer.Its value has following meaning. 1. positive – `o1` is greater than `o2` 2. zero – `o1` equals to `o2` 3. negative – `o1` is less than `o1`

Calling method	`Collections.sort(List)` Here objects will be sorted on the basis of `CompareTo` method	`Collections.sort(List, Comparator)` Here objects will be sorted on the basis of Compare method in Comparator
Java.lang.Comparable	`Java.util.Comparator`	

Java code

For Comparable

We will create class country having attribute id and name. This class will implement Comparable interface and implement `CompareTo` method to sort collection of country object by id.

1. Country.java

Hide Shrink ▲ Copy Code

```
package org.arpit.javapostsforlearning;



public class Country implements Comparable<Country>{
   int countryId;
   String countryName;


   public Country(int countryId, String countryName) {
      super();
      this.countryId = countryId;
      this.countryName = countryName;
   }

   @Override
   public int compareTo(Country country) {
     return (this.countryId < country.countryId ) ? -1: (this.countryId > country.countryId ) ? 1:0 ;
   }


   public int getCountryId() {
      return countryId;
   }


   public void setCountryId(int countryId) {
      this.countryId = countryId;
   }


   public String getCountryName() {
      return countryName;
   }


   public void setCountryName(String countryName) {
      this.countryName = countryName;
   }

}
```

2.ComparatorMain.java

Hide Shrink ▲ Copy Code

```
package org.arpit.javapostsforlearning;

import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

public class ComparableMain {


 public static void main(String[] args) {
   Country indiaCountry=new Country(1, "India");
   Country chinaCountry=new Country(4, "China");
   Country nepalCountry=new Country(3, "Nepal");
   Country bhutanCountry=new Country(2, "Bhutan");

        List<Country> listOfCountries = new ArrayList<Country>();
        listOfCountries.add(indiaCountry);
        listOfCountries.add(chinaCountry);
        listOfCountries.add(nepalCountry);
        listOfCountries.add(bhutanCountry);

        System.out.println("Before Sort  : ");
        for (int i = 0; i < listOfCountries.size(); i++) {
    Country country=(Country) listOfCountries.get(i);
    System.out.println("Country Id: "+country.getCountryId()+"||"+"Country name:
"+country.getCountryName());
   }
        Collections.sort(listOfCountries);

        System.out.println("After Sort  : ");
        for (int i = 0; i < listOfCountries.size(); i++) {
    Country country=(Country) listOfCountries.get(i);
    System.out.println("Country Id: "+country.getCountryId()+"|| "+"Country name:
"+country.getCountryName());
   }
 }

}
```

Output

Hide Copy Code

```
Before Sort  :
Country Id: 1||Country name: India
Country Id: 4||Country name: China
Country Id: 3||Country name: Nepal
Country Id: 2||Country name: Bhutan
After Sort  :
Country Id: 1|| Country name: India
Country Id: 2|| Country name: Bhutan
Country Id: 3|| Country name: Nepal
Country Id: 4|| Country name: China
```

For Comparator

We will create class country having attribute id and name and will create another class `CountrySortByIdComparator` which will implement Comparator interface and implement a compare method to sort collection of country object by id and we will also see how to use anonymous comparator.

1.Country.java

Hide Shrink Copy Code

```
package org.arpit.javapostsforlearning;

public class Country{
   int countryId;
   String countryName;

   public Country(int countryId, String countryName) {
      super();
      this.countryId = countryId;
      this.countryName = countryName;
   }

   public int getCountryId() {
      return countryId;
   }


   public void setCountryId(int countryId) {
      this.countryId = countryId;
   }


   public String getCountryName() {
      return countryName;
   }


   public void setCountryName(String countryName) {
      this.countryName = countryName;
   }

}
```

2.CountrySortbyIdComparator.java

Hide Copy Code

```
package org.arpit.javapostsforlearning;

import java.util.Comparator;



 public class CountrySortByIdComparator implements Comparator<Country>{

    @Override
    public int compare(Country country1, Country country2) {

        return (country1.getCountryId() < country2.getCountryId() ) ? -1: (country1.getCountryId() >
country2.getCountryId() ) ? 1:0 ;
    }

}
```

3.ComparatorMain.java

Hide Shrink ▲ Copy Code

```
package org.arpit.javapostsforlearning;

import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.List;

public class ComparatorMain {


 public static void main(String[] args) {
   Country indiaCountry=new Country(1, "India");
   Country chinaCountry=new Country(4, "China");
   Country nepalCountry=new Country(3, "Nepal");
   Country bhutanCountry=new Country(2, "Bhutan");

        List<Country> listOfCountries = new ArrayList<Country>();
        listOfCountries.add(indiaCountry);
        listOfCountries.add(chinaCountry);
        listOfCountries.add(nepalCountry);
        listOfCountries.add(bhutanCountry);

        System.out.println("Before Sort by id : ");
        for (int i = 0; i < listOfCountries.size(); i++) {
    Country country=(Country) listOfCountries.get(i);
    System.out.println("Country Id: "+country.getCountryId()+"||"+"Country name:
"+country.getCountryName());
   }
        Collections.sort(listOfCountries,new CountrySortByIdComparator());

        System.out.println("After Sort by id: ");
        for (int i = 0; i < listOfCountries.size(); i++) {
    Country country=(Country) listOfCountries.get(i);
    System.out.println("Country Id: "+country.getCountryId()+"|| "+"Country name:
```

```
"+country.getCountryName());
  }


       Collections.sort(listOfCountries,new Comparator<Country>() {

   @Override
   public int compare(Country o1, Country o2) {
    return o1.getCountryName().compareTo(o2.getCountryName());
   }
  });

  System.out.println("After Sort by name: ");
       for (int i = 0; i < listOfCountries.size(); i++) {
   Country country=(Country) listOfCountries.get(i);
   System.out.println("Country Id: "+country.getCountryId()+"|| "+"Country name:
"+country.getCountryName());
  }
 }

}
```

Output:

Hide Copy Code

```
Before Sort by id :
Country Id: 1||Country name: India
Country Id: 4||Country name: China
Country Id: 3||Country name: Nepal
Country Id: 2||Country name: Bhutan
After Sort by id:
Country Id: 1|| Country name: India
Country Id: 2|| Country name: Bhutan
Country Id: 3|| Country name: Nepal
Country Id: 4|| Country name: China
After Sort by name:
Country Id: 2|| Country name: Bhutan
Country Id: 4|| Country name: China
Country Id: 1|| Country name: India
Country Id: 3|| Country name: Nepal
```