Comparable Interface

Comparable is an interface which is present in java.lang package
In this interface,there is only one abstract method compareTo()
It returns negative, if objOne < objTwo
It returns positive, if objOne > objTwo
It returns zero, if objOne == objTwo
Only one sort sequence can be created
It is implemented by String, Wrapper Classes etc.

Note :-

The Collections.sort() uses compareTo() internally to sort the elements of ArrayList. Therefore the objects present inside the ArrayList must be Comparable type. i.e the class for which the objects are created must implement Comparable interface and provide implementaion to compareTo().

Syntax :-

```
public int compareTo(T o);
```

ex:-

```
public class CollFrameWork implements Comparable{
	int eid;
	CollFrameWork(int eid) {
		this.eid = eid;
	}
	@Override
	public String toString() {
		return "eid=" + eid ;
	}
	@Override
	public int compareTo(Object o) {
		CollFrameWork c=(CollFrameWork)o;
		if(eid==c.eid) {
			return 0;
		}
		else if(eid>c.eid) {
			return 1;
		}

		return -1;
	}
	public static void main(String[] args) {
		CollFrameWork cf=new CollFrameWork(3);
		CollFrameWork cf1=new CollFrameWork(1);
		CollFrameWork cf2=new CollFrameWork(2);
```

```
		ArrayList a=new ArrayList();
		a.add(cf);
		a.add(cf1);
		a.add(cf2);

		Collections.sort(a);
		for(Object ob:a) {
			System.out.println(ob);
		}
	}
}
```

ex2:-

```
public class CollFrameWork implements Comparable{
	int eid;
	String name;
	int salary;
	CollFrameWork(int eid, String name, int salary) {
		this.eid = eid;
		this.name = name;
		this.salary = salary;
	}
	@Override
	public String toString() {
		return "CollFrameWork [eid=" + eid + ", name=" + name + ", salary=" + salary
+ "]";
	}
	@Override
	public int compareTo(Object o) {
		CollFrameWork c=(CollFrameWork)o;
		if(salary==c.salary) {
			return 0;
		}
		else if(salary>c.salary) {
			return 1;
		}

		return -1;
	}
	public static void main(String[] args) {
		CollFrameWork cf=new CollFrameWork(3,"sathya",20000);
		CollFrameWork cf1=new CollFrameWork(1,"sathyam",22500);
		CollFrameWork cf2=new CollFrameWork(3,"malavika",12000);

		ArrayList a=new ArrayList();
		a.add(cf);
		a.add(cf1);
		a.add(cf2);
```

```
                Collections.sort(a);
                for(Object ob:a) {
                        System.out.println(ob);
                }
        }
}
```