Comparable e Comparator

Comparable è un interfaccia che spiega in come modo due oggetti della stessa classe devono essere ordinati. Tipicamente si usare quando si invoca il metodo *.sort()* delle liste.
Implementando Comparable si deve aggiungere il metodo compareTo():

```
public class Auto implements Comparable<Auto>

    private int annoFabbricazione;
    private String marca;
    ....

//Constructor, getters e setters vari

    ...
    public int compareTo(Auto a){ //Questo metodo compareTo ritorna un tipo intero, che
in base al segno

    return this.anno-o.anno;      //Permetterà di stabilire chi è il maggiore tra i due.
    }
```

Anche *Comparator* è un interfaccia utilizzata quando viene chiamato il metodo .sort().
Implementando Comparator in una classe specifichiamo in che modo .sort() cercherà di riordinare gli elementi. Ad esempio:

```
public class AutoMarcaCMP implements Comparator<Auto>

    @Override
    public int compare (Auto i, Auto j){ //Metodo di Comparator da sovrascrivere

    return i.getMarca().compareTo(j.getMarca());
    }
```

Il metodo .sort() fa parte della classe Liste che estende a sua volta la classe *Collections*. Un esempio di utilizzo (considerando gli esempi di sopra):

```
...
Auto auto1 = new Auto("Opel", 2002, 4, "no");
Auto auto2 = new Auto("BMW", 2001, 4, "si");
Auto auto3 = new Auto("Opel", 2003, 4, "boh");

ArrayList<Auto> lista = new ArrayList<>();
lista.add(auto1);
lista.add(auto2);
```

```
lista.add(auto3);

Collections.sort(lista);// lista contendo oggetti di tipo Auto non ha un modo
//"canonico" per essere ordinata, quindi ha bisogno che dentro Auto sia "spiegato" in che
modo ordinarle.
//In questo caso, attraverso l'anno di fabbricazione.

Collections.sort(lista, Collections.reverseOrder());  //oltre che la lista possiamo anche
passare a sort un Comparatore, in questo caso, Collections.reverseOrder() è standard e
inverte l'ordine già deciso di lista.

Collections.sort(lista, new AutoMarcaCMP); // invece ordina lista attraverso
l'ordinamento imparartito dal Comparator AutoMarcaCMP, ovvero in questo caso per ordine
alfabetico dell'attributo marca.
...
```