

Programação V

Prof. Alan

alanpedros@yahoo.com.br

Interface e Polimorfismo: Interfaces Comparable e Comparator

17 de agosto de 2007

1. Classe Casa

```
public class Casa implements Comparable {
    private float area;
    private float areaConstruida;
    private float numeroDeQuartos;
    private int vagasNaGaragem;
    private String nome;
    public Casa(String nome, float area, float areaConstruida,
        float numeroDeQuartos, int vagasNaGaragem) {
        this.area = area;
        this.areaConstruida = areaConstruida;
        this.numeroDeQuartos = numeroDeQuartos;
        this.vagasNaGaragem = vagasNaGaragem;
        this.nome = nome;
    }
    public float getArea() {
        return area;
    }
    public float getAreaConstruida() {
        return areaConstruida;
    }
    public float getNumeroDeQuartos() {
        return numeroDeQuartos;
    }
    public int getVagasNaGaragem() {
        return vagasNaGaragem;
    }
    public int compareTo(Object outro) {
        if (this.area < ((Casa) outro).getArea()) {
            return -1;
        } else if (this.area > ((Casa) outro).getArea()) {
            return 1;
        } else {
            return 0;
        }
    }
    public String getNome() {
        return nome;
    }
}
```

2. Primeiro Teste

```
public class PrimeiroTeste {
    public static void main(String[] args) {
        ArrayList lista = new ArrayList();
        Casa casa1 = new Casa("Casa 01", 100f, 80f, 3, 2);
        Casa casa2 = new Casa("Casa 02", 80f, 80f, 3, 2);
        Casa casa3 = new Casa("Casa 03", 250f, 80f, 3, 2);
        Casa casa4 = new Casa("Casa 04", 50f, 80f, 3, 2);
        Casa casa5 = new Casa("Casa 05", 450f, 80f, 3, 2);
        lista.add(casa1);
        lista.add(casa2);
        lista.add(casa3);
        lista.add(casa4);
        lista.add(casa5);
        Iterator listaFacil1 = lista.iterator();
        while (listaFacil1.hasNext()) {
```

```

        Casa casa = (Casa) listaFacil1.next();
        System.out.println("Casa: " + casa.getNomeÁ() + " // rea: "
            + casa.getArea());
    }
    Collections.sort(lista);
    System.out.println("*****");
    Iterator listaFacil2 = lista.iterator();
    while (listaFacil2.hasNext()) {
        Casa casa = (Casa) listaFacil2.next();
        System.out.println("Casa: " + casa.getNomeÁ() + " // rea: "
            + casa.getArea());
    }
}
}

```

3. Classe ComparadorAreaConstruida

```

public class ComparadorNumeroDeQuartos implements Comparator {
    public int compare(Object casa1, Object casa2) {
        Casa casa01 = (Casa) casa1;
        Casa casa02 = (Casa) casa2;
        if (casa01.getNumeroDeQuartos() < casa02.getNumeroDeQuartos()) {
            return -1;
        } else if (casa01.getNumeroDeQuartos() > casa02.getNumeroDeQuartos()) {
            return 1;
        } else {
            return 0;
        }
    }
}

```

4. Segundo Teste

```

public class SegundoTeste {
    public static void main(String[] args) {
        ArrayList lista = new ArrayList();
        Casa casa1 = new Casa("Casa 01", 100f, 80f, 10, 2);
        Casa casa2 = new Casa("Casa 02", 80f, 80f, 9, 2);
        Casa casa3 = new Casa("Casa 03", 250f, 80f, 8, 3);
        Casa casa4 = new Casa("Casa 04", 50f, 80f, 7, 4);
        Casa casa5 = new Casa("Casa 05", 450f, 80f, 6, 5);
        lista.add(casa1);
        lista.add(casa2);
        lista.add(casa3);
        lista.add(casa4);
        lista.add(casa5);
        Iterator listaFacil1 = lista.iterator();
        while (listaFacil1.hasNext()) {
            Casa casa = (Casa) listaFacil1.next();
            System.out.println("Casa: " + casa.getNome()
                + " // úNmero de quartos: " + casa.getNumeroDeQuartos());
        }
        Collections.sort(lista, new ComparadorNumeroDeQuartos());
        System.out.println("*****");
        Iterator listaFacil2 = lista.iterator();
        while (listaFacil2.hasNext()) {
            Casa casa = (Casa) listaFacil2.next();
            System.out.println("Casa: " + casa.getNome()
                + " // úNmero de quartos: " + casa.getNumeroDeQuartos());
        }
    }
}

```

- Exercício: Construir e testar um comparador para o número de vagas na garagem, de forma que no momento da ordenação, as casas fiquem em ordem decrescente.