

# Genéricos delimitados

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves

# Problema

Uma empresa de consultoria deseja avaliar a performance de produtos, funcionários, dentre outras coisas. Um dos cálculos que ela precisa é encontrar o maior dentre um conjunto de elementos. Fazer um programa que leia um conjunto de produtos a partir de um arquivo, conforme exemplo, e depois mostre o mais caro deles.

**Computer,890.50**

**IPhone X,910.00**

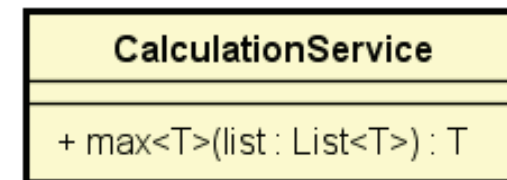
**Tablet,550.00**

Most expensive:

IPhone, 910.00

<https://github.com/acenelio/generics2-java>

Criar um serviço de cálculo:



Nota: Java possui:  
Collections.max(list)

```
package services;

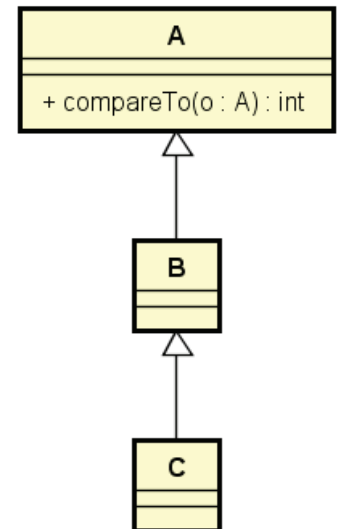
import java.util.List;

public class CalculationService {

    public static <T extends Comparable<T>> T max(List<T> list) {
        if (list.isEmpty()) {
            throw new IllegalStateException("List can't be empty");
        }
        T max = list.get(0);
        for (T item : list) {
            if (item.compareTo(max) > 0) {
                max = item;
            }
        }
        return max;
    }
}
```

# Versão alternativa (completa)

```
public static <T extends Comparable<? super T>> T max(List<T> list) {  
    if (list.isEmpty()) {  
        throw new IllegalStateException("List can't be empty");  
    }  
    T max = list.get(0);  
    for (T item : list) {  
        if (item.compareTo(max) > 0) {  
            max = item;  
        }  
    }  
    return max;  
}
```



# Tipos curinga (wildcard types)

<http://educandoweb.com.br>

Prof. Dr. Nelio Alves