

JAVA GENERICSS



GENERICOS

Utilizado para desenvolver algoritmos genéricos, que não dependem de um tipo específico. Além de promover a reusabilidade de código.

Introduzido na versão J2SE 5.0 do Java.





CLASSESGENERICAS

emploDeCódigo_Generics >  Printer.java > ...

```
1 import java.util.List;  
2  
3 public class Printer <T>{  
4     private T algumaCoisa;  
5  
6     public Printer(T algumaCoisa) {  
7         this.algumaCoisa = algumaCoisa;  
8     }  
9  
10    public void print() {  
11        System.out.println(algumaCoisa);  
12    }  
13  
14    public void printLista(List<?> algumaCoisa){  
15        for (Object o : algumaCoisa) {  
16            System.out.println(o);  
17        }  
18    }  
19}
```





MÉTODOS GENÉRICOS

```
public <U extends Animal & Serializable> void metodoGenerico(U coisaGenerica){  
    System.out.println(coisaGenerica);  
}  
  
public <U extends Animal> void metodoGenerico2(U coisaGenerica){  
    System.out.println(coisaGenerica);  
}  
  
public <U> void metodoGenerico3(U coisaGenerica){  
    System.out.println(coisaGenerica);  
}
```



MULTIPOS TIPOS GENERICOS

```
public <T, U, V> void multipleGenericType(T algumaCoisa,  
U outraCoisa, V outraCoisaAinda) {  
  
    System.out.println(algumaCoisa);  
    System.out.println(outraCoisa);  
    System.out.println(outraCoisaAinda);  
}
```

BOUNDED GENERICS



`<T extends Animal>`

`<T extends Animal & Serializable>`

Chamamos de “Bounded Generics” quando o tipo genérico possui uma amarração a uma certa classe abstrata ou interface

BOUNDED WILDCARDS

(01)

```
27     List<Integer> lista2 = new ArrayList<>();
28     lista2.add(e:1);
29     lista2.add(e:2);
30     lista2.add(e:3);
31     printer2.printLista(lista2);
32
33     List<Gato> lista3 = new ArrayList<>();
34     lista3.add(new Gato(nome:"Garfield", idade:3));
35     lista3.add(new Gato(nome:"Tom", idade:5));
36     lista3.add(new Gato(nome:"Felix", idade:2));
37     printer4.printLista(lista3);
```

(02)

```
15     public void printLista(List<?> algumaCoisa){
16         for (Object o : algumaCoisa) {
17             System.out.println(o);
18         }
19     }
```

THANK YOU!

