



DIGITAL
INNOVATION
ONE

Estrutura de Dados Em JAVA

João Ricardo Côre Dutra
Desenvolvedor Java backend na everis

Aula 1 | Etapa 6:

Introdução ao Curso e Conceitos Básicos

Generics em Java



```
Lista<String> minhaLista = new Lista<>();
```



```
public class Lista<T> {  
    private T t;  
    .  
    .  
    .  
}
```

Generics

Contexto:

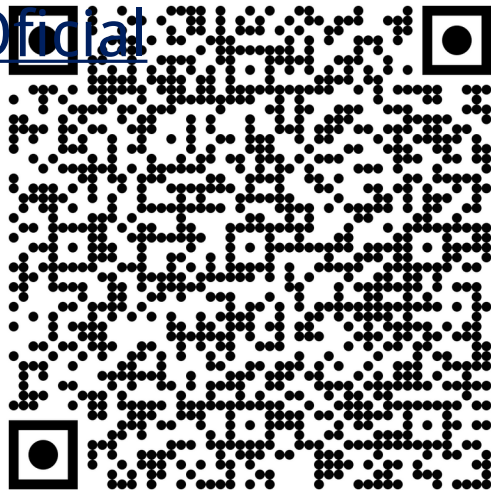
- Evitar casting excessivo
- Evitar códigos redundantes
- Encontrar erros em tempo de compilação
- Introduzido desde o Java SE 5.0

Wildcards

- Unknown Wildcards (Unbounded)
- Bounded Wildcard (Upper Bounded/Lower Bounded)

Documentação

Oficial



Unknown Wildcard

```
public void imprimeLista(List<?> lista){  
    for(Object obj : lista){  
        System.out.println(obj);  
    }  
}  
  
List<Aluno> minhaLista = new List<Aluno>();  
imprimeLista(minhaLista);
```



UpperBounded Wildcard

```
public void imprimeLista(List<? extends Pessoa>  
listaPessoas){  
    for(Pessoa p : listaPessoas){  
        System.out.println(p);  
    }  
}  
  
List<Aluno> minhaLista = new List<Aluno>();  
imprimeLista(minhaLista);
```



LowerBounded Wildcard

```
public void imprimeLista(List<? super Pessoa>
listaPessoas){
    for(Pessoa p : listaPessoas){
        System.out.println(p);
    }
}

List<Aluno> minhaLista = new List<Aluno>();
imprimeLista(minhaLista);
```

Convenção

- **K** para "Key", exemplo : Map<K,V>
- **V** para "Value", exemplo : Map<K,V>
- **E** para "Element", exemplo : List<E>
- **T** para "Type", exemplo : Collections#addAll
- **?** quando genérico

Mais Informações

The Java™ Tutorials

Generics (Updated)

Why Use Generics?

Generic Types

Raw Types

Generic Methods

Bounded Type Parameters

Generic Methods and

Bounded Type

Parameters

Generics, Inheritance, and Subtypes

Type Inference

Wildcards

Upper Bounded

Wildcards

Unbounded Wildcards

Lower Bounded

Wildcards

Wildcards and

Subtyping

« Previous • Trail • Next »

*The Java Tutorials have been written for JDK 8. Examples and
longer available.*

See [Java Language Changes](#) for a summary of updated language

See [JDK Release Notes](#) for information about new features, en

Why Use Generics?

In a nutshell, generics enable types (classes and interfaces) to
declarations, type parameters provide a way for you to re-use
parameters are types.

Code that uses generics has many benefits over non-generic

- Stronger type checks at compile time.

A Java compiler applies strong type checking to generic
difficult to find.



[Link para documentação Java](#)

Dúvidas durante o curso?

> Fórum do curso

> Comunidade online (discord)

 EM CASO DE
DÚVIDAS





DIGITAL
INNOVATION
ONE

Vamos Refatorar a Classe No() ?



[Vecteezy.com](https://www.vecteezy.com)