

# Classes em C++

## Nome da Classe Injetada em C++

### Introdução

O conceito de "nome da classe injetada" em C++ refere-se à habilidade de uma classe ou união de referenciar a si mesma através de um nome. Este nome é "injetado" no escopo da classe ou união e pode ser usado para declarar membros, criar instâncias e definir tipos aninhados. Isso é especialmente útil em templates e quando se deseja referenciar a classe atual de forma mais explícita.

### 1. Definição e Sintaxe

- Definição: O nome de uma classe ou união é automaticamente introduzido no escopo do corpo da classe ou união, permitindo o uso desse nome para se referir à própria classe ou união.

- Sintaxe:

```
class NomeClasse {  
    // Corpo da classe  
};
```

```
union NomeUniao {  
    // Corpo da união  
};
```

### 2. Exemplo Simples

## Classes em C++

- Exemplo com Classe:

```
class MinhaClasse {  
  
public:  
  
    MinhaClasse* obterPtr() {  
  
        return this;  
  
    }  
  
};  
  
int main() {  
  
    MinhaClasse obj;  
  
    MinhaClasse* ptr = obj.obterPtr();  
  
    return 0;  
  
}
```

- Exemplo com União:

```
union MinhaUniao {  
  
    int inteiro;  
  
    float flutuante;  
  
  
    void setInt(int valor) {  
  
        this->inteiro = valor;  
  
    }  
  
};
```

## Classes em C++

```
int main() {  
    MinhaUniao u;  
    u.setInt(5);  
    return 0;  
}
```

### 3. Uso em Templates

- Definição: O nome da classe injetada é especialmente útil em classes template, onde ele pode ser usado para referenciar a própria classe template dentro de suas definições.

- Exemplo:

```
template <typename T>  
class MinhaClasseTemplate {  
public:  
    void metodo() {  
        MinhaClasseTemplate* ptr = this;  
    }  
};
```

### 4. Tipos Aninhados e Nome da Classe Injetada

- Definição: O nome da classe injetada pode ser usado para declarar tipos aninhados dentro da classe.

- Exemplo:

```
class Externa {
```

## Classes em C++

public:

```
class Interna {
```

```
public:
```

```
    Interna* obterPtr() {
```

```
        return this;
```

```
    }
```

```
};
```

```
    Interna criarInterna() {
```

```
        return Interna();
```

```
    }
```

```
};
```

```
int main() {
```

```
    Externa e;
```

```
    Externa::Interna i = e.criarInterna();
```

```
    return 0;
```

```
}
```

### 5. Referência a Tipos de Base

- Definição: Em uma classe derivada, o nome da classe injetada pode ser usado para referenciar a classe base.

- Exemplo:

```
class Base {
```

## Classes em C++

public:

using TipoBase = int;

};

class Derivada : public Base {

public:

TipoBase valor; // Refere-se a Base::TipoBase

};

### 6. Vantagens do Uso do Nome da Classe Injetada

- Clareza: Melhora a clareza do código ao tornar explícito que uma referência está apontando para a própria classe ou união.
- Segurança: Reduz a chance de erros, especialmente em templates, onde o nome da classe injetada assegura que a referência é corretamente resolvida.
- Flexibilidade: Facilita a criação de tipos aninhados e a utilização de membros da própria classe dentro de suas definições.

### Dicas de Boas Práticas

- Consistência: Use o nome da classe injetada consistentemente para melhorar a legibilidade e a manutenibilidade do código.
- Documentação: Documente o uso do nome da classe injetada para que outros desenvolvedores compreendam sua finalidade e funcionamento.
- Evite Ambiguidade: Em classes derivadas, use o nome da classe injetada para evitar ambiguidade

## **Classes em C++**

ao referenciar membros da classe base.

Esta seção abrange os conceitos sobre o nome da classe injetada em C++. Para mais detalhes, consulte a documentação oficial: <https://en.cppreference.com/w/cpp/language/injected-class-name>