

`auto` 是 C++11 引入的一个关键字，用于自动类型推导。使用 `auto` 关键字可以让编译器根据变量的初始化值自动推断出它的类型。这样可以减少代码中显式指定类型的需要，尤其是在类型比较复杂时，代码变得更简洁明了。

## 用法示例

### 1. 基本使用

```
cpp                                                                    复制代码

auto x = 10;           // 自动推导为 int
auto y = 3.14;         // 自动推导为 double
auto z = "Hello";      // 自动推导为 const char*
```

### 2. 与容器一起使用 在处理 STL 容器时，`auto` 非常方便。例如，用于迭代器类型或复杂的模板类型：

```
cpp                                                                    复制代码

#include <vector>
#include <iostream>
using namespace std;

int main() {
    vector<int> numbers = {1, 2, 3, 4, 5};

    // 使用 auto 来推导迭代器类型
    for (auto it = numbers.begin(); it != numbers.end(); ++it) {
        cout << *it << " ";
    }

    // 使用 auto 来推导范围循环中的元素类型
    for (auto num : numbers) {
        cout << num << " ";
    }

    return 0;
}
```

### 3. 函数返回类型 函数的返回类型也可以用 `auto` 推导。为了使用 `auto` 作为返回类型，必须在函数定义之前添加 `->` 符号和返回类型的推断标记：

```
cpp                                                                    复制代码

auto add(int a, int b) -> int {
    return a + b;
}
```