

C++：引用的数组和数组的引用之间的区别？



C++ 专栏收录该内容

0 订阅 8 篇文章

订阅专栏

一、声明、定义和初始化

- 1、声明：声明变量类型和名字
- 2、定义：根据类型分配内存 **地址空间**
- 3、初始化：将初始值拷贝到变量的内存地址空间中

二、为什么声明一个"引用"必须初始化？

因为引用需要绑定到初始化对象，所以引用必须初始化

三、引用的**数组**和数组的引用之间的区别

首先，需要明确引用的概念，引用是对某个目标对象取了一个别名，对该引用的操作与对变量直接操作的效果是一样的。

- 1、引用的数组：字面意思可以理解为定义一个数组，数组中存放的元素都是"引用"，但这个明显就是**不合法的**，因为引用只是取了一个别名，本身并不占用内存，而数组是会占用内存的，所以**引用的数组是错误的**。

示例：

```
int array[2]={1, 2};
int& a[] = array //可以理解为(int&) a[],即a为数组，数组中的元素为引用
```

- 2、数组的引用：对数组名取一个别名

```
int array[2]={1, 2};
int (&a)[2]= array; //a为引用，引用对象为array
a[0]=3;
```

cout<<array[0]<<endl; //输出为3，对a中元素进行修改就是对array中的元素进行修改

```
// number of elements in a raw array:
template<typename T, unsigned N>
std::size_t len(T(&)[N])
{
    return N;
}

// number of elements for a type having size_type:
template<typename T>
typename T::size_type len(const T& t)
{
    return t.size();
}
```