

引用的本质：给对象起别名

语法： 数据类型 &别名 = 原名

注意事项：

- 引用必须初始化
- 引用在初始化后，不可以改变

引用可以作为函数参数

作用： 函数传参时，可以利用引用的技术让形参修饰实参

优点： 可以简化指针修改实参

```
void mySwap(int& a, int& b) {  
    int temp = a;  
    a = b;  
    b = temp;  
}  
int main()  
{  
    int a=3,b=5;  
    mySswap(3,5);  
    cout<<"a="<<a<<" b="<<b<<endl;  
    system("pause");  
    return 0;  
}
```

输出：a=5 b=3

结论：使用引用作为形参，可以修改实参。

引用可以做函数返回值

作用：引用是可以作为函数的返回值存在的

注意：不要返回局部变量引用

原因是：局部变量保存在栈区，当 函数调用完毕之后，编译器只保留一次数据，第二次再去使用这个变量就会输出乱码了。

用法： 函数调用作为左值

相当于给函数起了别名，例如： `int & ref2 = test();`

然后我们就可以使用 `ref2` 来调用这个函数。

引用本质上在C++内部实现是一个指针常量，即指向的值可以修改，但方向不可以修改。

结论：C++推荐用引用技术，因为语法方便，引用本质是指针常量，但是所有的指针操作编译器都帮我们做了