一个命名空间是一个独立的作用域。不同的命名空间，可以有同名的成员。

命名空间可以是不连续的，一个命名空间的定义可以分离在多个文件中（参考智能指针模板xxcig的定义）。

如果多个命名空间都定义了同一个函数，引用这同一个函数，会构成一组重载函数（前提是这些函数符合重载函数的要求）。

命名空间可以是全局的，也可以定义在另一个命名空间中，但不能定义在类和代码块中。

test\_a.h

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Copyright (c) 2013

\* All rights reserved

\*

\* 文件名称：test\_a.h

\* 摘 要：命名空间test\_a的定义

\*

\* 版本号：1.0 作者：刘珅珅 日期：2013/7/8

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#ifndef TEST\_A\_H

#define TEST\_A\_H

namespace test\_a {

class A {

public:

void func();

private:

int a;

};

void func\_test(int a);

}

#endif // TEST\_A\_H

test\_a.cpp

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Copyright (c) 2013

\* All rights reserved

\*

\* 文件名称：test\_a.cpp

\* 摘 要：命名空间test\_a的实现

\*

\* 版本号：1.0 作者：刘珅珅 日期：2013/7/8

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "test\_a.h"

#include <iostream>

using std::cout;

namespace test\_a {

void A::func()

{

a = 1;

cout << "A func test" << a << std::endl;

}

void func\_test(int a)

{

cout << "func\_test int a " << a << std::endl;

}

}

test\_b.h

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Copyright (c) 2013

\* All rights reserved

\*

\* 文件名称：test\_b.h

\* 摘 要：命名空间test\_b的定义

\*

\* 版本号：1.0 作者：刘珅珅 日期：2013/7/8

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#ifndef TEST\_B\_H

#define TEST\_B\_H

namespace test\_b {

class B {

public:

void func();

private:

int b;

};

void func\_test(float b);

// void func\_test(int b);

}

#endif // TEST\_B\_H

test\_b.cpp

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Copyright (c) 2013

\* All rights reserved

\*

\* 文件名称：test\_b.cpp

\* 摘 要：命名空间test\_b的实现

\*

\* 版本号：1.0 作者：刘珅珅 日期：2013/7/8

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "test\_b.h"

#include <iostream>

using std::cout;

using std::endl;

namespace test\_b {

void B::func()

{

b = 2;

cout << "func test B" << b << std::endl;

}

void func\_test(float b)

{

cout << "funct\_test float b" << b << endl;

}

// void func\_test(int b)

// {

// cout << "funct\_test int b" << b << endl;

// }

}

test\_c.h

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Copyright (c) 2013

\* All rights reserved

\*

\* 文件名称：test\_c.h

\* 摘 要：补充命名空间test\_a的定义

\*

\* 版本号：1.0 作者：刘珅珅 日期：2013/7/8

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#ifndef TEST\_C\_H

#define TEST\_C\_H

namespace test\_a {

int add(int a, int b);

}

#endif

test\_c.cpp

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Copyright (c) 2013

\* All rights reserved

\*

\* 文件名称：test\_c.cpp

\* 摘 要：补充命名空间test\_a的实现

\*

\* 版本号：1.0 作者：刘珅珅 日期：2013/7/8

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include "test\_c.h"

#include <iostream>

using std::cout;

using std::endl;

namespace test\_a {

int add(int a, int b)

{

int c = a + b;

cout << "add 结果为" << c << endl;

return c;

}

}

test.cpp

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Copyright (c) 2013

\* All rights reserved

\*

\* 文件名称：test.cpp

\* 摘 要：命名空间的测试

\*

\* 版本号：1.0 作者：刘珅珅 日期：2013/7/8

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include "test\_a.h"

#include "test\_b.h"

#include "test\_c.h"

using std::cout;

using std::endl;

using std::string;

using test\_a::func\_test;

using test\_b::func\_test;

using test\_a::A;

void func\_test(string test);

int main(int argc, char\* argv[])

{

int a = 5;

func\_test(a);

float b = static\_cast<float>(5.6);

func\_test(b);

string c = "123";

func\_test(c);

A d;

d.func();

int e = 5;

int f = 5;

test\_a::add(e, f);

return 0;

}

void func\_test(string test)

{

cout << "func\_test string " << test.c\_str() << endl;

}