匿名函数：传递给委托构造函数的未命名代码块。

例：程序anonymous\_method\_test1

// Copyright 2016.刘珅珅

// author：刘珅珅

// 匿名方法

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

namespace anonymous\_method\_test1

{

// 委托不能进行重载

delegate void CountIt();

delegate void CountOther(int end);

class AnonymousMethodTest

{

static void Main(string[] args)

{

CountIt count = delegate // 匿名方法

{

for (int i = 0; i <= 5; ++i)

{

*Console*.*WriteLine*(i);

}

};

count();

*Console*.*WriteLine*();

*Console*.*WriteLine*("with parameter:");

CountOther count1 = delegate(int end)

{

for (int i = 0; i <= end; ++i)

{

*Console*.*WriteLine*(i);

}

};

count1(5);

}

}

}

输出结果：

0

1

2

3

4

5

with parameter:

0

1

2

3

4

5

具有返回值得匿名函数

例：程序anonymout\_method\_test2

// Copyright 2016.刘珅珅

// author：刘珅珅

// 带有返回值的匿名方法

using *System*;

using *System*.*Collections*.*Generic*;

using *System*.*Linq*;

using *System*.*Text*;

using *System*.*Threading*.*Tasks*;

namespace anonymout\_method\_test2

{

delegate int CountIt(int end);

class AnonymousMethodTest

{

static void Main(string[] args)

{

CountIt count = delegate(int end)

{

int sum = 0;

for (int i = 0; i <= end; ++i)

{

sum += i;

}

return sum;

};

int result = count(5);

*Console*.*WriteLine*("Summation of 5 is " + result);

}

}

}