C# - Anonymous Method

As the name suggests, an anonymous method is a method without a name. Anonymous methods in C# can be defined using the delegate keyword and can be assigned to a variable of delegate type.

Example: Anonymous Method

public delegate void Print(int value);

static void Main(string[] args)

{

Print print = delegate(int val) {

Console.WriteLine("Inside Anonymous method. Value: {0}", val);

};

print(100);

}

[Try it](https://www.tutorialsteacher.com/codeeditor?cid=cs-d67VSW)

Output:

Inside Anonymous method. Value: 100

Anonymous methods can access variables defined in an outer function.

Example: Anonymous Method

public delegate void Print(int value);

static void Main(string[] args)

{

int i = 10;

Print prnt = delegate(int val) {

val += i;

Console.WriteLine("Anonymous method: {0}", val);

};

prnt(100);

}

[Try it](https://www.tutorialsteacher.com/codeeditor?cid=cs-eIs9tQ)

Output:

Anonymous method: 110

Anonymous methods can also be passed to a method that accepts the delegate as a parameter.

In the following example, PrintHelperMethod() takes the first parameters of the Print delegate:

Example: Anonymous Method as Parameter

public delegate void Print(int value);

class Program

{

public static void PrintHelperMethod(Print printDel,int val)

{

val += 10;

printDel(val);

}

static void Main(string[] args)

{

PrintHelperMethod(delegate(int val) { Console.WriteLine("Anonymous method: {0}", val); }, 100);

}

}

[Try it](https://www.tutorialsteacher.com/codeeditor?cid=cs-dw7eFq)

Output:

Anonymous method: 110

Anonymous methods can be used as event handlers:

Example: Anonymous Method as Event Handler

saveButton.Click += delegate(Object o, EventArgs e)

{

System.Windows.Forms.MessageBox.Show("Save Successfully!");

};

C# 3.0 introduced the [lambda expression](https://www.tutorialsteacher.com/linq/linq-lambda-expression) which also works like an anonymous method.

Anonymous Method Limitations

* It cannot contain jump statement like goto, break or continue.
* It cannot access ref or out parameter of an outer method.
* It cannot have or access unsafe code.
* It cannot be used on the left side of the is operator.

 Points to Remember :

1. Anonymous method can be defined using the delegate keyword
2. Anonymous method must be assigned to a delegate.
3. Anonymous method can access outer variables or functions.
4. Anonymous method can be passed as a parameter.
5. Anonymous method can be used as event handlers.