Loops and Garbage Collection

Asked 16 years, 2 months ago   Modified 15 years, 5 months ago   Viewed 4k times

4

I am working on a web application and I have run into the following situation.

```vbnet
Dim a as Object
Dim i as Integer = 0

Try

    For i=1 to 5

        a = new Object()

        'Do stuff '

        a = Nothing

    Next

Catch

Finally

    a = Nothing

End Try
```

Do i need to do the a=Nothing in the loop or will the garbage collector clean a up?

`.net`   `vb.net`   `garbage-collection`   `loops`

Share
Improve this question
Follow

edited Jul 21, 2009 at 18:03
MarkJ
30.4k ● 5 ● 71 ● 113

asked Oct 6, 2008 at 18:02
wax eagle
541 ● 11 ● 27

5 Answers

Sorted by: Highest score (default) ▲▼

12

In .NET, you generally do not **need** to set a variable reference = `Nothing` (`null` in C#). The garbage collector will clean up, eventually. The reference itself will be destroyed when it goes out of scope (either when your method exits or when the object of this class is finalized.) Note that this doesn't mean the **object** is destroyed, just the reference to it. The object will still be destroyed non-deterministically by the collector.

However, setting your reference = `Nothing` will provide a hint to .NET that the object may be garbage, and doesn't necessarily hurt anything -- aside from code clutter. If you were to keep it in there, I'd recommend removing it from `Try` block; it's already in the `Finally` block and will therefore always be called. (Aside from certain catastrophic exceptions; but in those cases it wouldn't get called in the `Try` block either!)

Finally, I have to admit that I agree with Greg: Your code would be cleaner without this. The hint to the runtime that you're done with the reference is nice, but certainly not critical. Honestly, if I saw this in a code review, I'd probably have the developer rewrite it thusly:

```
Dim a as Object
Dim i as Integer = 0

For i=1 to 5
    a = new Object()
    'Do stuff
Next
```

Share Improve this answer Follow

answered Oct 6, 2008 at 18:08

John Rudy
37.8k ● 14 ● 66 ● 101

---

5   Actually setting the variable to null may cause the object's lifetime to be extended rather than shortened (blogs.msdn.com/csharpfaq/archive/2004/03/26/97229.aspx). Though the general message of the post is correct -- don't bother setting things to null. – Greg Beech Oct 6, 2008 at 20:24

Touche. The .NET JIT continues to surprise me with its intelligence and efficiency! – John Rudy Oct 7, 2008 at 1:59

5

It is almost never necessary to explicitly assign Nothing to a variable. The garbage collector's job is to take care of memory allocation for you, specifically to relieve you of this responsibility. So no, you don't need to assign `a = Nothing` inside the loop.

You also do not need the `try/finally` block that assigns Nothing around the whole thing. This is really just extra clutter that the runtime system will take care of anyway.

Share Improve this answer Follow

answered Oct 6, 2008 at 18:05

Greg Hewgill
990k ● 191 ● 1.2k ● 1.3k

▲

2

▼

No you don't need it. .NET has garbage collection. And since it looks like this code is in a method scope the garbage collection will clean up any local variables.

Share Improve this answer Follow

answered Oct 6, 2008 at 18:04

[Epaga]
39k ● 59 ● 162 ● 246

▲

0

▼

The GC will clean it up.

Share Improve this answer Follow

answered Oct 6, 2008 at 18:07

user13165

▲

0

▼

Like everyone stated above you do not need to explicitly set your variables to nothing, as it is handled automatically. However, if for whatever reason you want to force the GC to collect, you can run this:

```
System.GC.Collect()
```

Share Improve this answer Follow

answered Oct 6, 2008 at 18:40

[Anders]
12.5k ● 36 ● 101 ● 146

Forcing garbage collection is probably a really bad idea. (It definitely is in Java!0 – Stephen C
Jul 23, 2009 at 5:51