

Re: C# Fundamentals Part 3: ReferenceEquals question

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.csharp/2006-04/msg00066.html>

- *From:* "Willy Denoyette [MVP]" <willy.denoyette@xxxxxxxxxx>
 - *Date:* Sat, 1 Apr 2006 19:19:52 +0200
-

"Jon Skeet [C# MVP]" <skeet@xxxxxxxx> wrote in message
<news:MPG.1e98ac645fe6f01898d01f@xxxxxxxxxxxxxxxxxxxxxxxxxxxx>

| Willy Denoyette [MVP] <willy.denoyette@xxxxxxxx> wrote:

| > | In this case, no new instance is being created :(

| >

| > Yes, but all String(Char[]) or String(Char*) constructors are special cases,

| > take a look at the remark clause(s) in the Framework reference guide.

| > ...

| > If value is a null reference (Nothing in Visual Basic) or contains no

| > element, an Empty instance is initialized.

| > ...

|

| I don't think that documentation even makes sense. What is an "Empty
| instance" when Empty is a property? It should probably be worded as "a
| new object is not created; instead, the value of String.Empty is
| returned". Even that's not perfect as constructors initialise an
| instance rather than returning it, but it makes more sense than what's
| there, IMO...

|

Actually, String.Empty is a public static field of type String initialized to "" by the String .ctor. That means that the first string in an AD initializes this field to point (a reference) to the interned (process wide) empty string object.

| > On the current version of the CLR, it means that:

| >

| > string x = String.Empty;

| > string y = new string (new char[] { });

| >

| > x and y are references to the same string 'object', but:

| > – no new instances of String.Empty are ever created, and it's references are

| > not tracked (like x and y in above)

Re: C# Fundamentals Part 3: ReferenceEquals question

|
| Again, "instances of String.Empty" doesn't make sense, as String.Empty
| is a property, not a type.
|

Sorry I didn't make myself more clear, My point was to show the remark did not make much sense, that's why I said there is never an 'instance' of an 'Empty' string created by this constructor.

And ...Again, it's a static field of type String.

| Besides, you can certainly get different instances of System.String
| which *are* empty in other ways:
|

Did I say otherwise?

```
| using System;  
| using System.Text;  
|  
| class Test  
| {  
| static void Main()  
| {  
| string x = new StringBuilder().ToString();  
| string y = new StringBuilder().ToString();  
| Console.WriteLine (object.ReferenceEquals(x, y));  
| }  
| }  
|
```

| Either way, this constructor means that the C# spec is violated – the
| "new" operator doesn't return a new instance.
|

Let's say, it violates the specs for this particular case (IMO an optimization with no added value anyway), IMO it's the C# the spec which is too strict, string objects are so special it should mention this as an exceptions.

Willy.

.