

# Re: how to find the type of an object represented by an interface

---

*Source:*

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.aspnet/2007-02/msg02945.html>

---

- *From:* "Random" <cipherlad@xxxxxxxxxxxx>
  - *Date:* Fri, 23 Feb 2007 16:51:14 -0800
- 

Okay. Actually, I found the answer (seems I always stumble on it shortly after posting), but it led to another problem.

```
Sub GetNewObjectOfType(ByVal sender As IGenericInference)
Dim o As Object = sender
Dim T As Type = o.GetType()
```

```
'now, what I need to do is call a method that takes a generics argument
Dim newobject as Object = SomeObjectOfType(Of T)()
End Sub
```

```
Function SomeObjectOfType(Of T As IGenericInference)() As T
(code to create new object of type T)
End Function
```

Looks like I can't call `SomeObjectOfType(Of T)()` with a variable as a reference. Weird, because if `T` was passed into the Sub via a generics `Of` statement I would be able to pass it on.

"Milosz Skalecki [MCAD]" <mily242@xxxxxxxxxxxxxxxxxxxx> wrote in message <news:F0266DD5-BAE1-4288-A06D-A23D004BBBC9@xxxxxxxxxxxxxxxxxxxx>

Show us the code

--

Milosz

"Random" wrote:

How can I use reflection (or some other method) to find the type of an object that has been passed in to my method under an interface definition?

I try to use `GetType`, but that won't work.

Re: how to find the type of an object represented by an interface