

C# Tip 0001 — Utility of the var Keyword

© 2009 Rex Jaeschke. All rights reserved.

Issue: The var keyword introduced in V3 looks like syntactic sugar. It is useful in general programming?

Response: Unless you wish to define a variable having an anonymous type, you are not required to use var. However, it certainly provides notational convenience. Consider the following code fragment:

```
/*1*/ int i = 10;
/*2*/ List<OpenXmlElement> statusXml = new List<OpenXmlElement>();
/*3*/ foreach (OpenXmlElement element in statusXml)
    {
        // ...
    }
/*4*/ System.Collections.Generic.List<System.Collections.Generic.List<String>>
    list = new
        System.Collections.Generic.List<System.Collections.Generic.List<String>>();
```

This can be rewritten using var, as follows:

```
/*1*/ var i = 10;
/*2*/ var statusXml = new List<OpenXmlElement>();
/*3*/ foreach (var element in statusXml)
    {
        // ...
    }
/*4*/ var list = new
    System.Collections.Generic.List<System.Collections.Generic.List<String>>();
```

The presence of var instead of the actual type allows the compiler to determine each variable's type from the initializer, which cannot be null. The inferred type can be any built-in, user-defined, anonymous, or library type. The keyword var can be used on local variables and to control the for, foreach, and using statements.

The use of var certainly helps when using long and/or complicated names, especially when fully qualified namespace prefixes are used, as in case 4 above.

The keyword var must be used when an anonymous type is needed; for example:

```
var x = new { Direction = "North", Distance = 4.5 };
```

(Expressions having anonymous types can often occur in Language-Integrated Query [LINQ] queries.)