

C SHARP VAR DATA TYPE AND ANONYMOUS TYPE

Var data type was introduced in C# 3.0. var is used to declare implicitly typed local variable means it tells the compiler to figure out the type of the variable at compilation time. A var variable must be initialized at the time of declaration.

Valid var statements

```
1. var str = "1";
2. var num = 0;
3. string s = "string";
4. var s2 = s;
5. s2 = null;
6. string s3 = null;
7. var s4 = s3;
```

At compile time, the above var statements are compiled to IL, like this:

```
1. string str = "1";
2. int num = 0;
3. string s2 = s;
4. string s4 = s3;
```

The compile-time type value of var variable cannot be null but the runtime value can be null.

```
1. // invalid var statements
2. var v; //need to initialize
3. var num = null; // can't be null at compile time
```

Once var variable is initialized its data type became fixed to the type of the initial data.

```
1. // invalid var statements
2. var v2 = "str12";
3. v2 = 3; // int value can't be assign to implicitly type string variable
   v2
```

Anonymous Types

An anonymous type is a simple class generated by the compiler within IL to store a set of values. var data type and new keyword is used to create an anonymous type.

```
1. var emp = new { Name = "Deepak", Address = "Noida", Salary=21000 };
```

At compile time, the compiler will create an anonymous type, as follows:

```
1. class __Anonymous1
2. {
3.     private string name;
4.     private string address;
5.     int salary; public string Name
6.     {
7.         get{return name; }
8.         set { name=value }
9.     }
10.    public string Address
11.    {
12.        get{ return address; }
13.        set{ address=value; }
14.    }
15.    public int Salary
16.    {
17.        get{ return salary; }
18.        set{ salary=value; }
19.    }
20. }
```

The anonymous type is very useful when you want to shape the result in your desired form like this:

```
1. var result =from book in Books
2. where book.Price > 200
3. orderby book.IssueDate descending
4. select new
5. {
6. Name = book.Name,
7. IssueNumber = "#" + book.Issue
8. };
```

In above example, I change the name of the "Issue" field of Book table to "IssueNumber" and add # before value to get desired output.

Source :<http://www.dotnet-tricks.com/Tutorial/csharp/RJK3170612-C-Sharp-Var-data-type-and-Anonymous-Type.html>