

[0061] Readability and Maintainability

Even though we haven't covered C# syntax yet, I bet you can figure out what at least one of these two programs does.

```

1  using System;
2
3  namespace LWTech.ChipAnderson.HelloWorldApp2
4  {
5      class Program
6      {
7          static void Main()
8          {
9              Console.WriteLine(reverse("Testing"));
10         }
11
12         static string reverse(string s) {
13             string t = "";
14
15             float x = 0f;
16             int x1 = s.Length;
17             do
18             {
19                 string y = "";
20                 if (x1 > 0 && x < s.Length)
21                 {
22                     y += s[s.Length - (int)(++x)];
23                 } else if (x > x1) {
24                     y += s;
25                 }
26                 t += y;
27             } while (x < s.Length);
28
29             return t;
30         }
31     }
32 }
33

```

```

1  using System;
2
3  namespace LWTech.ChipAnderson.HelloWorldApp2
4  {
5      class Program
6      {
7          static void Main()
8          {
9              Console.WriteLine(reverse("Testing"));
10         }
11
12         static string reverse(string s)
13         {
14             string result = "";
15
16             int i = s.Length - 1;
17             while (i >= 0)
18             {
19                 result = result + s[i];
20                 i = i - 1;
21             }
22             return result;
23         }
24     }
25 }
26 |

```

Which program would you be more comfortable adding a new feature to? If I told you they both did the same thing, would you believe me? Are you sure? Which one is more likely to have a bug in it?

Clearly, the second program is much easier to read and understand. Given that both programs execute in almost exactly the same time, which version do you think a business would value more?

Closely related to the concept of "Readability" is the concept of "Self-Documenting code" - i.e., a program where the code itself makes it clear what it does. Look back at the two code examples above. How many comments do you see in these programs? Zero. Now, ask yourself the following - if someone took the time to add comments to the first program, would that make the program more maintainable and more understandable than the second program? In general, I maintain that it would not.

The reason the second program is better is because the purpose of the program is "baked" into the identifier names that the program uses - i.e., the method and variable names.

While both programs have a method called "reverse()" that (probably) returns a reversed version of the given string, the indicator names inside the first program's code are quite cryptic while, in the second program, they make more sense. In addition, the second program uses a very straightforward kind of loop with a very well-defined exit condition and, unlike the first program, it doesn't create any extraneous variables. All of those characteristics make the second program more readable and understandable.

Maintainability

The concept of maintainability is more than just having readable, self-documented code. It's about have code that can be maintained while it is in production. "In production" means that the code is actively being used by users and thus is providing benefits (money!) to the company. If a bug is discovered in code that has been released and put into production, it is often fixed by a different set of developers - maintenance developers who are members of the Operations team.

Maintenance developers often have to fix problems in a high-pressure environment. If a bug has been found in production code, and if people think that bug needs fixing, it is probably because the bug is causing a significant problem for a large number of paying customers and therefore fixing it quickly is very important. I sometimes describe the process of fixing bugs in production code as working "with your hair on fire" - meaning you have to find and fix the bug quickly under a huge amount of pressure.

So guess who really, really, really appreciates readable, understandable code? Yes, it is the maintenance/operations team that often benefits!

Also, YOU... in the future...

