



What Is High-Quality Programming Code?

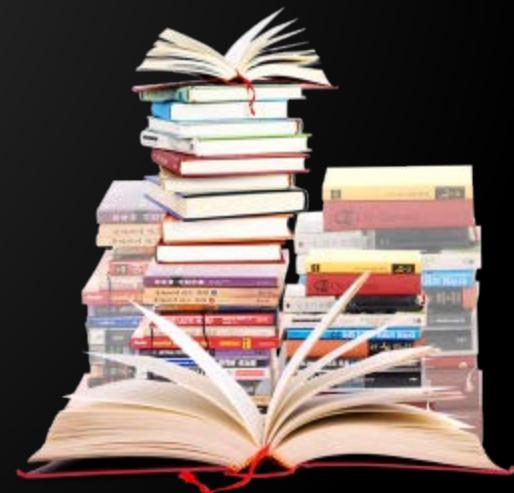
Code Correctness, Readability, Maintainability

Telerik Software Academy
Learning & Development
<http://academy.telerik.com>



Table of Contents

- ◆ Why Quality Is Important?
- ◆ Software Quality: External and Internal
- ◆ What is High-Quality Code?
- ◆ Code Conventions
- ◆ Managing Complexity
- ◆ Characteristics of Quality Code



What is High-Quality Programming Code?

```
    "e service
    " Scheduler
    Sched.getEvents()
    event = null;
    table = renderable ?
    getUserOrganiz
    e = null;
    on.han
```



Why Quality Is Important?

```
static void Main()
{
    int value=010, i=5, w;
    switch(value){case
10:w=5;Console.WriteLine(w);break;case 9:i=0;break;
        case 8:Console.WriteLine("8 ");break;
    default:Console.WriteLine("def ");
        Console.WriteLine("hoho ");
    for (int k = 0; k < i; k++,
Console.WriteLine(k - 'f'));break;} {
    Console.WriteLine("loop!"); }
}
```



What does this code do? Is it correct?

Why Quality Is Important? (2)

```
static void Main()
{
    int value = 010, i = 5, w;
    switch (value)
    {
        case 10: w = 5; Console.WriteLine(w); break;
        case 9: i = 0; break;
        case 8: Console.WriteLine("8 "); break;
        default:
            Console.WriteLine("def ");
            Console.WriteLine("hoho ");
            for (int k = 0; k < i; k++,
                Console.WriteLine(k - 'f')) ;
            break;
    }
    Console.WriteLine("loop!");
}
```



Now the code is formatted, but is still unclear.

◆ External quality

- Does the software behave correctly?
- Are the produced results correct?
- Does the software run fast?
- Is the software UI easy-to-use?
- Is the code secure enough?

◆ Internal quality

- Is the code easy to read and understand?
- Is the code well structured?
- Is the code easy to modify?



What is High-Quality Programming Code?

- ◆ High-quality programming code:
 - ◆ Easy to read and understand
 - ◆ Easy to modify and maintain
 - ◆ Correct behavior in all cases
 - ◆ Well tested
 - ◆ Well architected and designed
 - ◆ Well documented
 - ◆ Self-documenting code
 - ◆ Well formatted



What is High-Quality Programming Code? (2)

- ◆ **High-quality programming code:**
 - ◆ **Strong cohesion at all levels: modules, classes, methods, etc.**
 - ◆ **Single unit is responsible for single task**
 - ◆ **Loose coupling between modules, classes, methods, etc.**
 - ◆ **Units are independent one of another**
 - ◆ **Good formatting**
 - ◆ **Good names for classes, methods, variables, etc.**
 - ◆ **Self-documenting code style**

Code Conventions



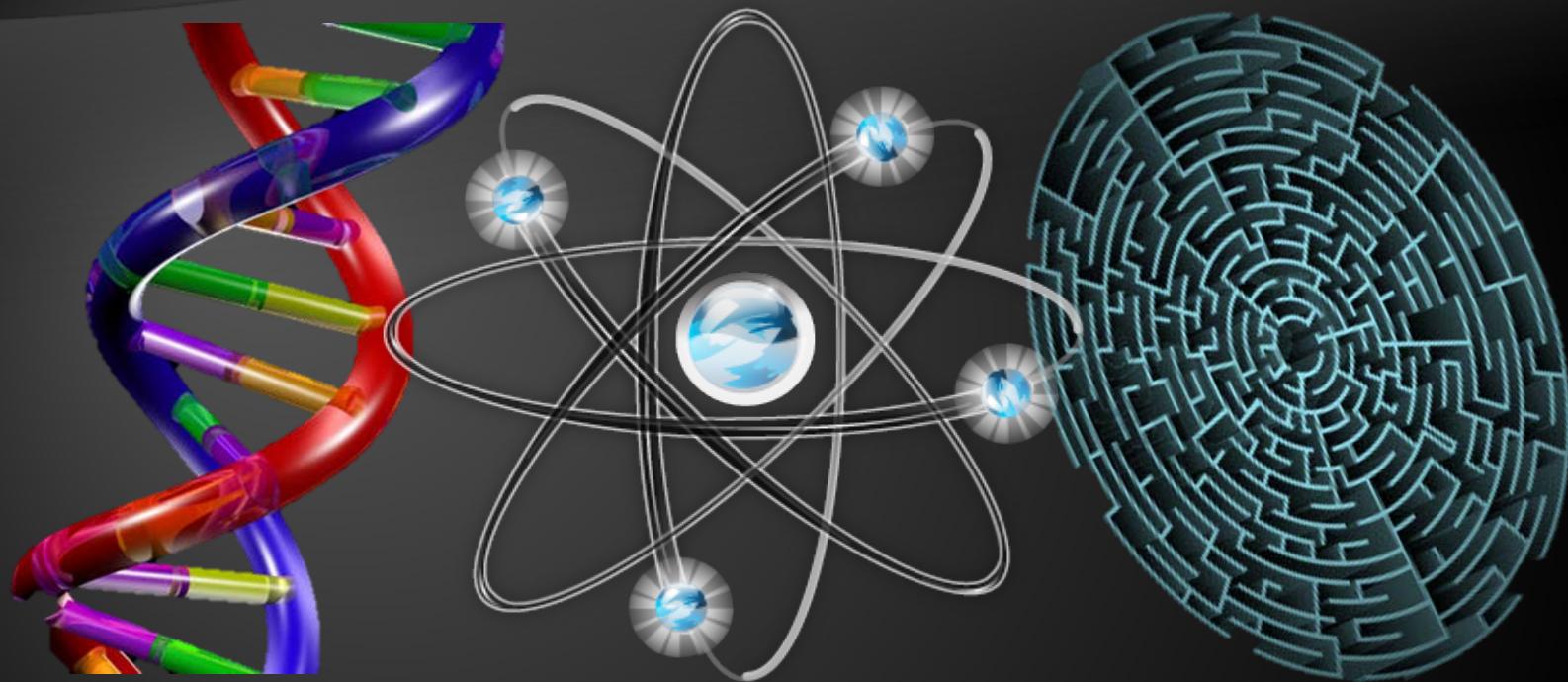
Code Conventions

- ◆ Code conventions are formal guidelines about the style of the source code:
 - ◆ Code formatting conventions
 - ◆ Indentation, whitespace, etc.
 - ◆ Naming conventions
 - ◆ PascalCase or camelCase, prefixes, suffixes, etc.
 - ◆ Best practices
 - ◆ Classes, interfaces, enumerations, structures, inheritance, exceptions, properties, events, constructors, fields, operators, etc.



Code Conventions (2)

- ◆ Microsoft has official C# code conventions
 - ◆ Design Guidelines for Developing Class Libraries:
<http://msdn.microsoft.com/en-us/library/ms229042.aspx>
- ◆ Semi-official JavaScript code conventions
 - ◆ <http://javascript.crockford.com/code.html>, <http://google-styleguide.googlecode.com/svn/trunk/javascriptguide.xml>
- ◆ Large organization follow strict conventions
 - ◆ Code conventions can vary in different teams
- ◆ High-quality code goes beyond code conventions
 - ◆ Software quality is a way of thinking!



Managing Complexity

Managing Complexity

- ◆ Managing complexity has central role in software construction
 - ◆ Minimize the amount of complexity that anyone's brain has to deal with at certain time
- ◆ Architecture and design challenges
 - ◆ Design modules and classes to reduce complexity
- ◆ Code construction challenges
 - ◆ Apply good software construction practices: classes, methods, variables, naming, statements, error handling, formatting, comments, etc.

Managing Complexity (2)

- ◆ Key to being an effective programmer:
 - ◆ Maximizing the portion of a program that you can safely ignore
 - ◆ While working on any one section of code
 - ◆ Most practices discussed later propose ways to achieve this important goal

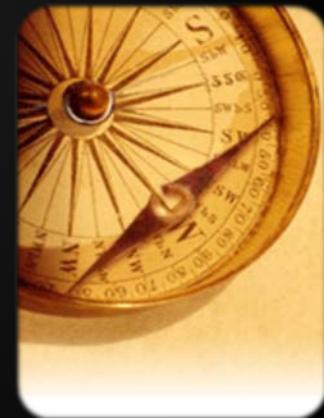




Code Quality: Characteristics

Key Characteristics of High-Quality Code

- ◆ Correct behavior
 - ◆ Conforming to the requirements
 - ◆ Stable, no hangs, no crashes
 - ◆ Bug free – works as expected
 - ◆ Correct response to incorrect usage
- ◆ Readable – easy to read
- ◆ Understandable – self-documenting
- ◆ Maintainable – easy to modify when required



Key Characteristics of High-Quality Code (2)

- ◆ Good identifiers' names
 - Good names for variables, constants, methods, parameters, classes, structures, fields, properties, interfaces, structures, enumerations, namespaces,
- ◆ High-quality classes, interfaces and class hierarchies
 - Good abstraction and encapsulation
 - Simplicity, reusability, minimal complexity
 - Strong cohesion, loose coupling

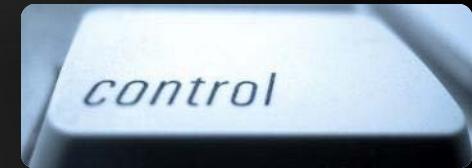
Key Characteristics of High-Quality Code (3)

- ◆ **High-quality methods**
 - ◆ Reduced complexity, improved readability
 - ◆ Good method names and parameter names
 - ◆ Strong cohesion, loose coupling
- ◆ **Variables, data, expressions and constants**
 - ◆ Minimal variable scope, span, live time
 - ◆ Simple expressions
 - ◆ Correctly used constants
 - ◆ Correctly organized data



Key Characteristics of High-Quality Code (4)

- ◆ Correctly used control structures
 - Simple statements
 - Simple conditional statements and simple conditions
 - Well organized loops without deep nesting
- ◆ Good code formatting
 - Reflecting the logical structure of the program
 - Good formatting of classes, methods, blocks, whitespace, long lines, alignment, etc.



Key Characteristics of High-Quality Code (5)

- ◆ **High-quality documentation and comments**
 - ◆ Effective comments
 - ◆ Self-documenting code
- ◆ **Defensive programming and exceptions**
 - ◆ Ubiquitous use of defensive programming
 - ◆ Well organized exception handling
- ◆ **Code tuning and optimization**
 - ◆ Quality code instead of good performance
 - ◆ Code performance when required



Key Characteristics of High-Quality Code (6)

- ◆ Following the corporate code conventions
 - ◆ Formatting and style, naming, etc.
 - ◆ Domain-specific best practices
- ◆ Well tested and reviewed
 - ◆ Testable code
 - ◆ Well designed unit tests
 - ◆ Tests for all scenarios
 - ◆ High code coverage
 - ◆ Passed code reviews and inspections



What Is High-Quality Programming Code?

Questions?

Free Trainings @ Telerik Academy

- ◆ C# Programming @ Telerik Academy

- ◆ csharpfundamentals.telerik.com



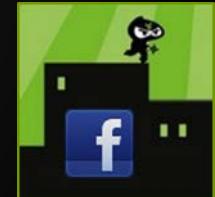
- ◆ Telerik Software Academy

- ◆ academy.telerik.com



- ◆ Telerik Academy @ Facebook

- ◆ facebook.com/TelerikAcademy



- ◆ Telerik Software Academy Forums

- ◆ forums.academy.telerik.com

