Clean Code Homicidal Maniacs Read Code, Too

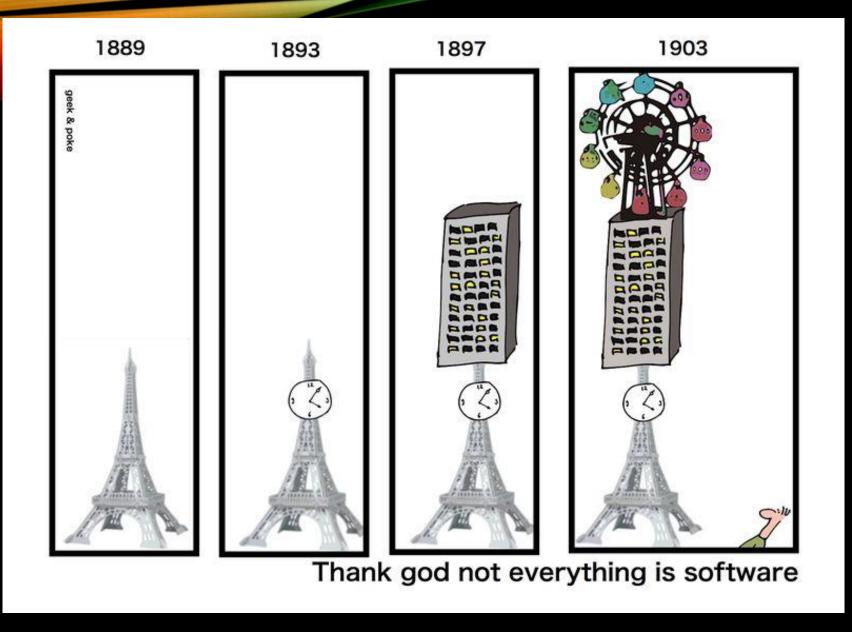
Presented by Jeremy Clark www.jeremybytes.com

What is Clean Code?

- Readable
- Maintainable
- Testable
- Elegant

Why Do We Care?

There's no such thing as write-once code



http://geekandpoke.typepad.com/geekandpoke/2012/03/thank-god-not-everything-is-software.html

Why Do We Care?

There's no such thing as write-once code

• Bug Fixes

- Business Changes
- Enhancements
- New Functionality

What Prevents Clean Code?

Ignorance

Arrogance

- Stubbornness
 Job Security
- Short-Timer Syndrome
 Scheduling

What Prevents Clean Code?

Number one reason:

"I'll clean it up later."

Pro Tip: "Later" never comes.

The Truth about Clean Code

- Clean Code saves time.
- We can't take a short-term view of software.

• We need to look at the lifespan of the application.

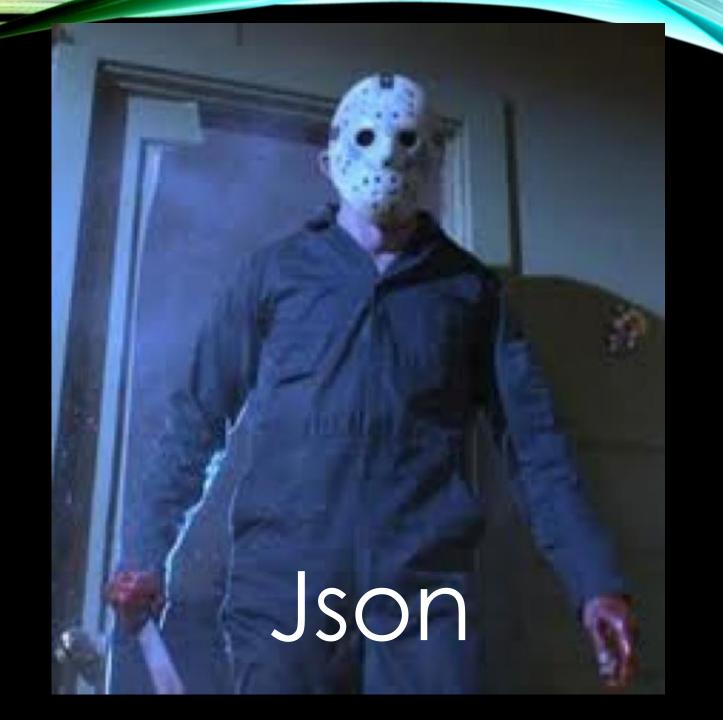
How Do You Write Clean Code?

• Rule of Thumb:

Imagine that the developer who comes after you is a homicidal maniac who knows where you live.

-Unknown

The Next Developer



This Might Take Awhile



The Problem

- Readable (by mere mortals)
- Maintainable
- Testable
- Elegant

All of these qualities are subjective.

Robert C. Martin

PRENTICE

Robert C. Martin Series

Clean Code

A Handbook of Agile Software Craftsmanship

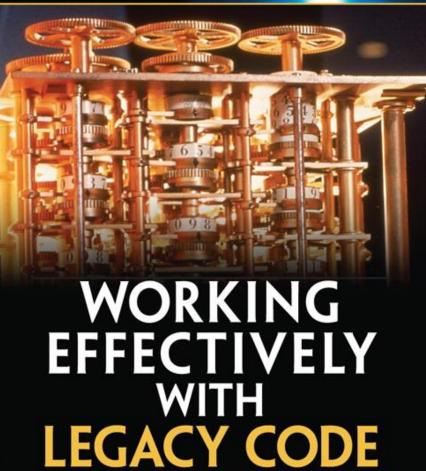


Foreword by James O. Coplien

Robert C. Martin

Michael C. Feathers

Robert C. Martin Series



Michael C. Feathers

The Dry Principle

Don't Repeat Yourself

• copy/pasta = spaghetti code



Naming

Intentional Naming

• theList • Not very good

• ProductList

• A bit better

ProductCatalog Good

Naming

- Use Nouns for Variables, Properties, Parameters
 indexer, currentUser, PriceFilter
- Use Verbs for Methods and Functions
 - SaveOrder(), getDiscounts(), RunPayroll()
- Pronounceable and Unambiguous
 - recdptrl = received patrol? record department role?

Naming Standard

- Camel Case?
- Pascal Case?
- Lower Case with Underscores?

It doesn't matter Have a Standard Be Consistent

Comments

// Determine if End of Day Time for Last Date
// has been reached
// If Last Date is null use Converted Date
// Based on Today's Date > Last Date
// And Curr Time >= End of Day Time

Comments

• Rule #1: Comments lie

• Code is updated or moved, but not the comments

Comments Lie



Comments

- Rule #1: Comments lie
 - Code is updated or moved, but not the comments
- Rule #2: Comments do not make up for bad code
 If the code is that unclear, rewrite the code

Good Comments

- Can be used to describe intent or clarification
 Ex: // Sample input: Oct 5, 2015 13:54:15 PDT
- Can be used to give warnings or consequences
 Ex: // We do a deep copy of this collection to make // sure that updates to one copy do not affect // the other

Good Comments

- Can be used for TODOs
 - Especially useful when the IDE supports it
 - These should be temporary

Bad Comments

• Avoid "journaling" comments

- Ex: // 03/20/1996 jjc Added tax calculation
- This is what source control is for: Who, What, When

Know Your Tools



Bad Comments

• Avoid "journaling" comments

- Ex: // 03/20/1996 jjc Added tax calculation
- This is what source control is for: Who, What, When
- Avoid "noise" comments
 - Ex: // Default constructor

Bad Comments

• Do not comment out code

- Code no longer in use should be deleted
- If needed, you can always retrieve it from source control

Know Your Tools



Functions and Methods



Functions and Methods

- Keep methods short
 - Should fit on a single screen
 - Prefer methods no longer than 10 lines

Do one thing!

Multiple Levels of Methods

• High level

- Overview of functionality
- Mid-level
 - More details, but not too deep
- Detail
 - The "weeds" of the functionality

Work in Small Chunks

If you aren't writing incremental code, you are writing excremental code.

What is Refactoring?

Making code better without changing the functionality

Refactoring and Unit Testing

- If you don't have unit tests, you don't know what your code does.
- Refactoring Step 1:
 - Bring your code under test.
- Refactoring Step 2:
 - Safely and confidently update the code.

The Watcher



Making Code Cleaner

ILE EDIT VIEW			WINDOW HELP	uick Launch (Ctrl+Q)	
G - O Test Explorer	CatalogViewModel.cs + ×	Debug ← P → h f		 ✓ Solution Explorer ✓ O O A Yo - ₽ ✓ Search Solution Explorer 	0000
	<pre>private IPersonService GetServiceFromContainer() { if (!_container.IsRegistered<ipersonservice>()) throw new MissingFieldException("IPersonService is not available from the DI Container" return _container.Resolve<ipersonservice>(); } private void PopulateCatalogFromService() </ipersonservice></ipersonservice></pre>			Solution 'Refactorin Stuff We Care A Stuff We Care A C Module.Cat Module.Cat Module.Cat C Catalogy C ShareTre C Catalogy C ShareTre C Common People.Servi C Services C ShareTre C Services C ShareTre C Services C ShareTre C Services C ShareTre C Services C ShareTre C Services C ShareTre C Services C ShareTre C Shar	about alog.Test es fiewModelTest.cs estObjects.cs Application ce
	<pre>var asyncBegin var task = Tas asyncBegin task.ContinueW { _fullPeopl ResetFilte LastUpdate }, TaskContinu</pre>	<pre>List<person>(); = _service.BeginGetPeople(null, nul k<list<person>>.Factory.FromAsync(, _service.EndGetPeople); ith(t => eList = t.Result; rToDefaults(); Time = DateTime.Now; ationOptions.NotOnFaulted); sFromService(task);</list<person></person></pre>	1);		
100 % - 4	1	STI 000501 VICe(Cask);		Properties Solution	Team Fx Class View
	Error List Output Find Results 1 Find	Symbol Results Undo Close Code Metrics Results Data Tools O	perations		
ady			Ln 253	Col 1 Ch 1	INS

Be a Clean Code Advocate

The Boy Scout Rule Always leave the campground cleaner than you found it.

The Clean Coder Rule

Always leave the code cleaner than you found it.

Thank You!

- http://www.jeremybytes.com
- jeremy@jeremybytes.com
- @jeremybytes