Xamarin Studio
File
Edit
View
Search
Project
Build
Run
Version Control
Tools
Window
Help
Debug
protected bool
SetProperty<T> (
ref T storage,
T value,
string propertyName = null
)

CLEANCODE

AGENDA

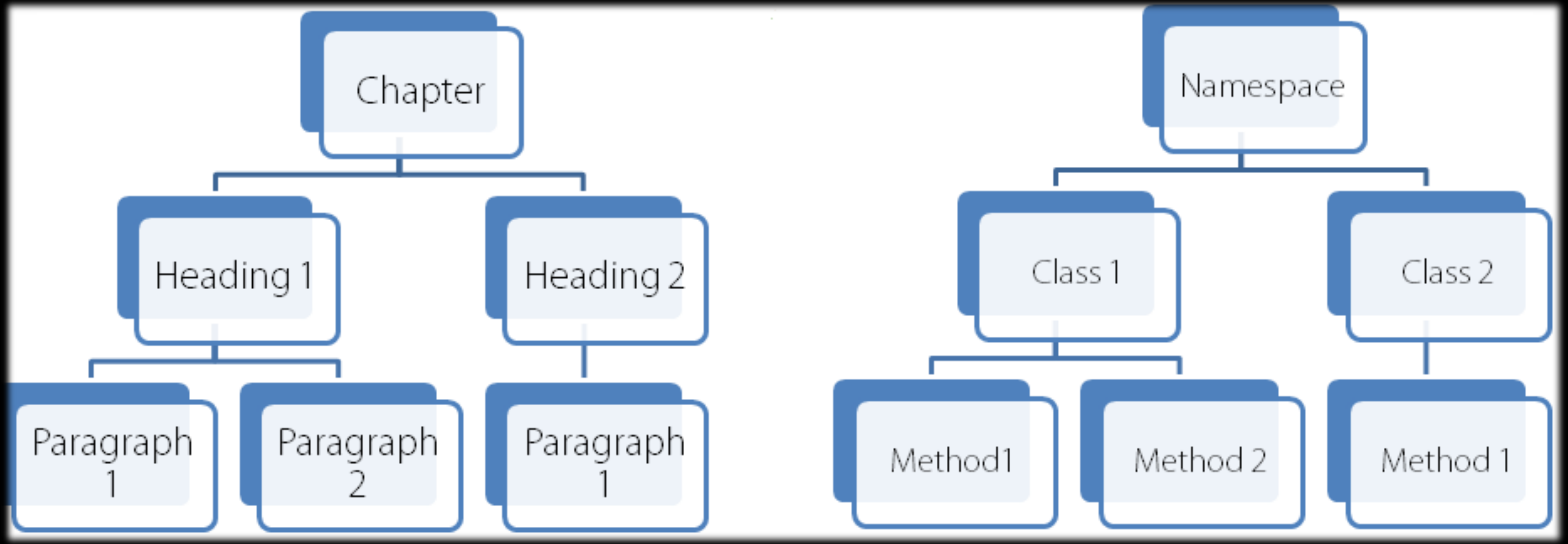
- What is Clean Code?
- Why Clean Code matters?
- How to write Clean Code?
- Code Sample
- What prevents Clean Code?
- Conclusion

"Anyone can write code that a computer can understand.
Good programmers write code that humans can understand."

- Martin Fowler

What is Clean Code?

- Reader - focused development Style
- Easy to Write
- Easy to Read
- Easy to Maintain



Why Clean Code Matters?

- There is no such thing as write once code
  - Bug Fixes
  - Enhancement
  - New Functionality

"ALWAYS CODE AS IF THE GUY WHO ENDS UP MAINTAINING YOUR CODE WILL BE A VIOLENT PSYCHOPATH WHO KNOWS WHERE YOU LIVE."

- Martin Golding

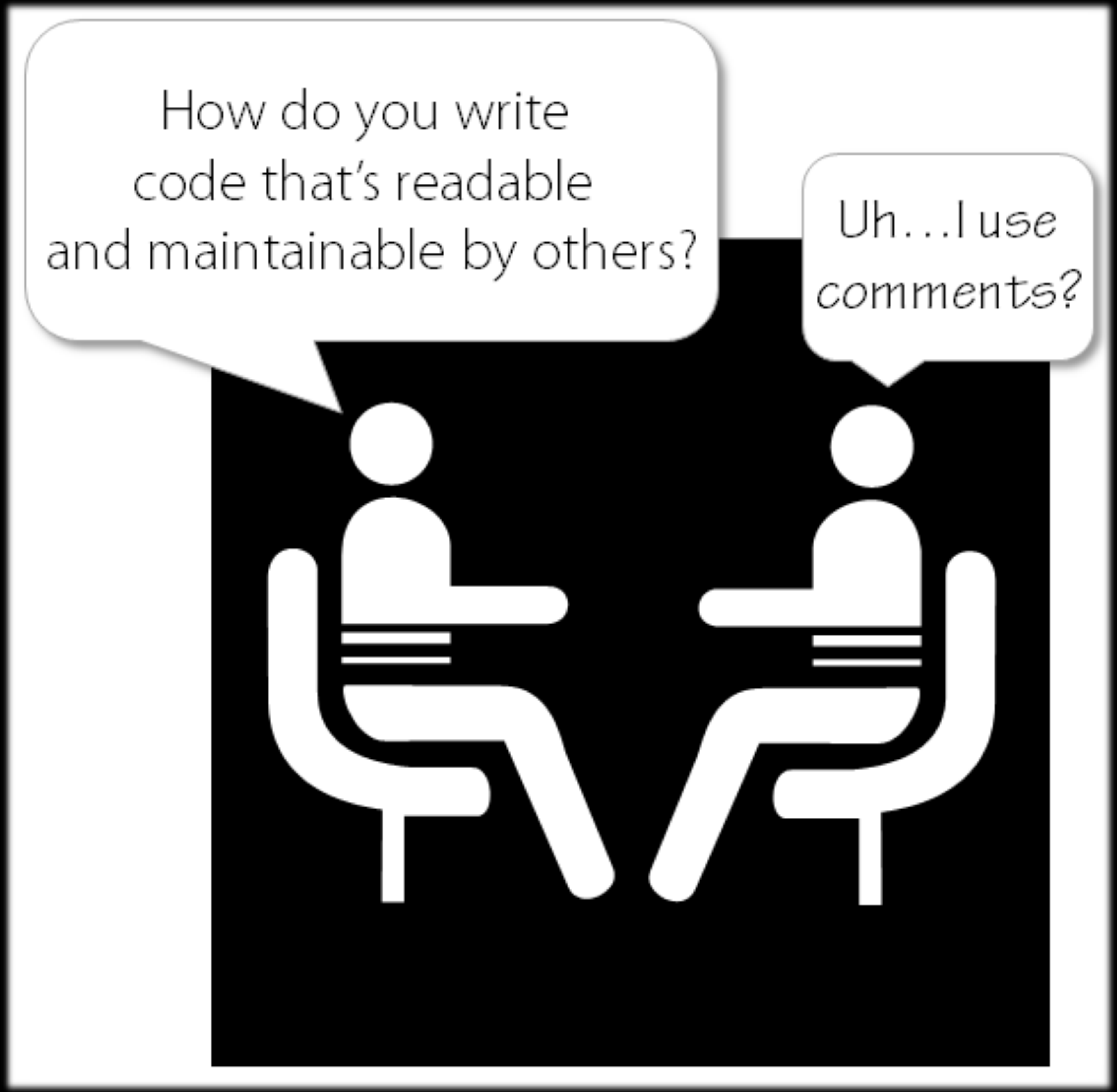
“The first 90 percent of the code accounts for the first 90 percent of the development time.

The remaining 10 percent of the code accounts for the other 90 percent of the development time!.”

– Tom Cargill, Bell Labs

How to write Clean Code?

- Variable and method name
- Better functions
- Comments
- Other code smells


How do you write code that's readable and maintainable by others?
Uh…I use comments?

Variable and method Name

1. Naming Convention

Context	Prefix	How it would look
Data member	*m*	*mVariableName*
Global Variable	*g*	*gVariableName*
Static Variable	*s*	*sVariableName*
Constant	*k*	*kConstant*
Enumerations	*e*	*eName*
Input Parameter	*in*	*inVariableName*
Output Parameter	*out*	*outVariableName*
Input/Output Parameter	*Io*	*ioVariableName*

2. Use intention revealing name

3. Use pronounceable name

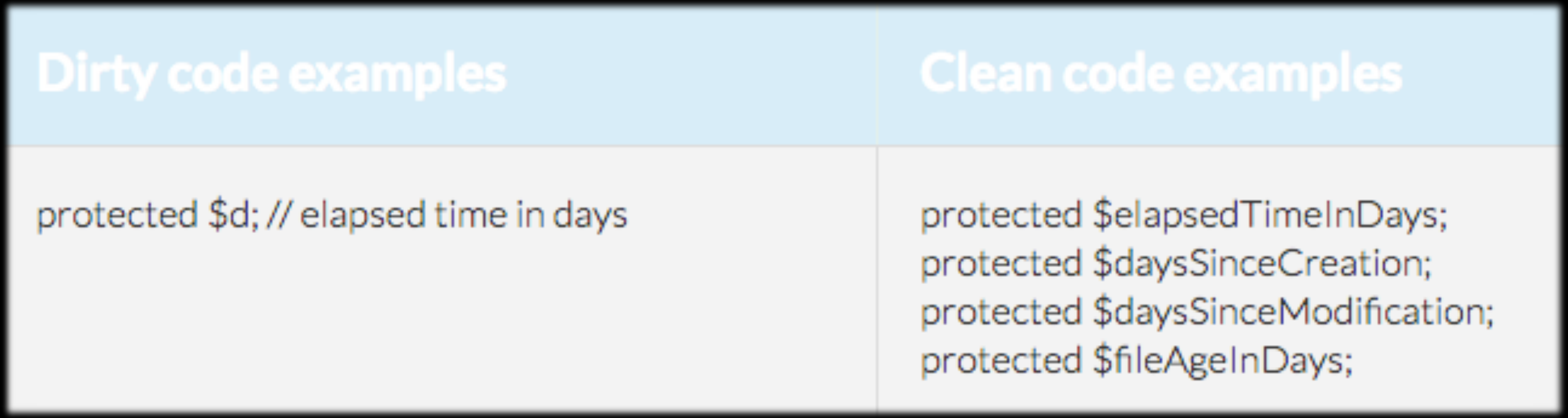
4. Use verbs for function names and nouns for classes and attributes

5. Use camelCase for variable and (or PascalCase for) function name

6. The class should start with upper case

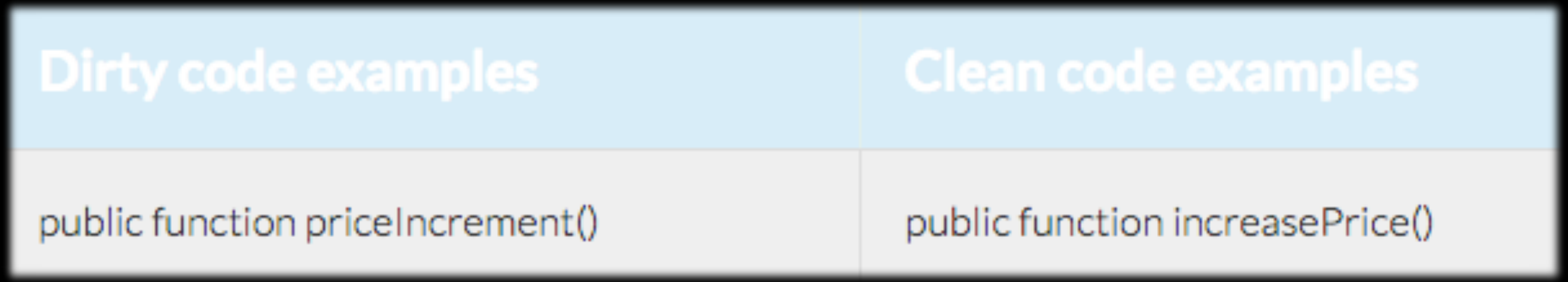
7. Use one word per concept (Eg. Don’t use **get** and **fetch** to do same thing)

8. Don’t hesitate to use technical terms in names



Dirty code examples	Clean code examples

protected $d; // elapsed time in days	protected $elapsedTimeInDays;
	protected $daysSinceCreation;
	protected $daysSinceModification;
	protected $fileAgeInDays;



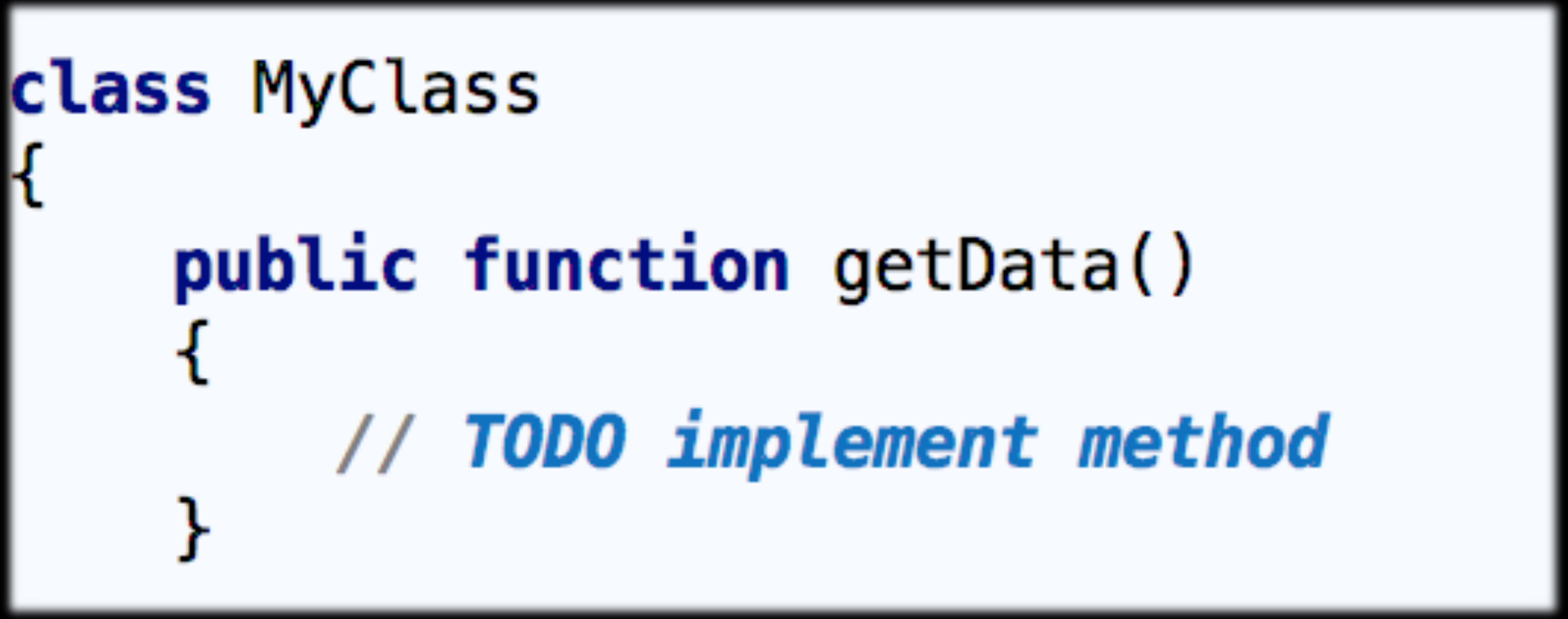
Dirty code examples	Clean code examples
public function priceIncrement()	public function increasePrice()

Better Functions

- The smaller the better
- A function should only do one thing
- Less arguments are better
- No side effects
- Don’t repeat yourself

Comments

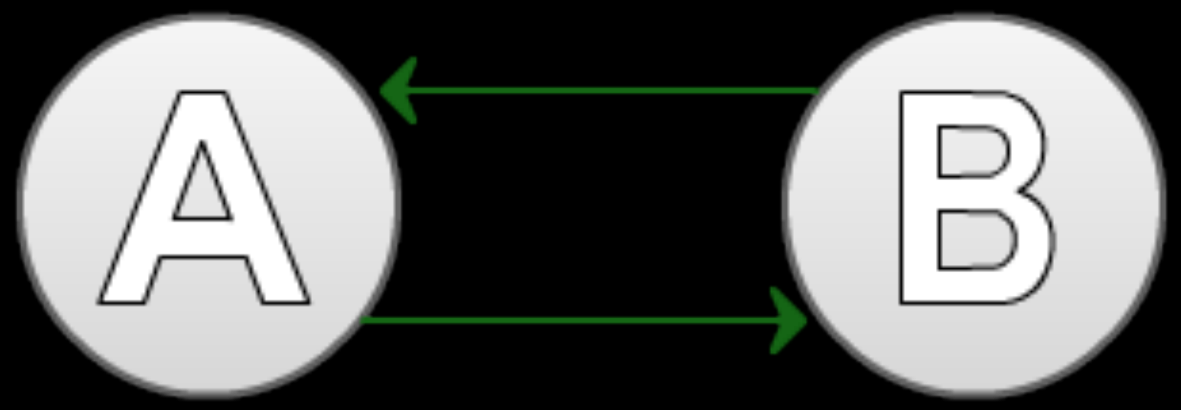
- Don’t comment bad code, rewrite it
- Warn of consequences in comments
- Emphasis important points in comments
- Can be used for TODOs
- Noise comments are bad
- Do not comment out code
- Comments may lie



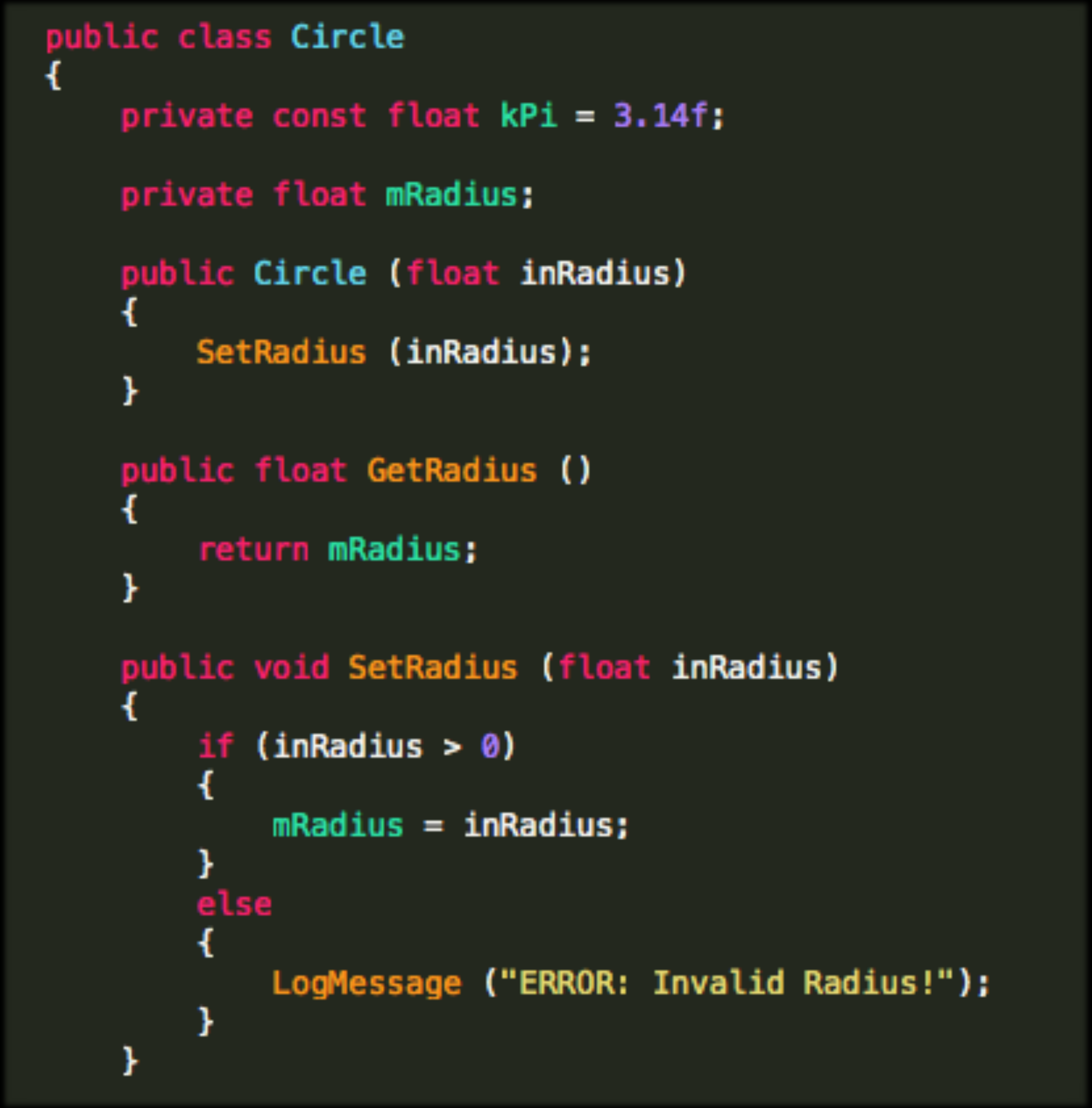
```
class MyClass
{
    public function getData()
    {
        // TODO implement method
    }
```

Other code smells

- Overuse of static
- Long if conditions - replace with function
- Hard-coding
- Circular dependency
- Formatting (Indentation)



Code Sample



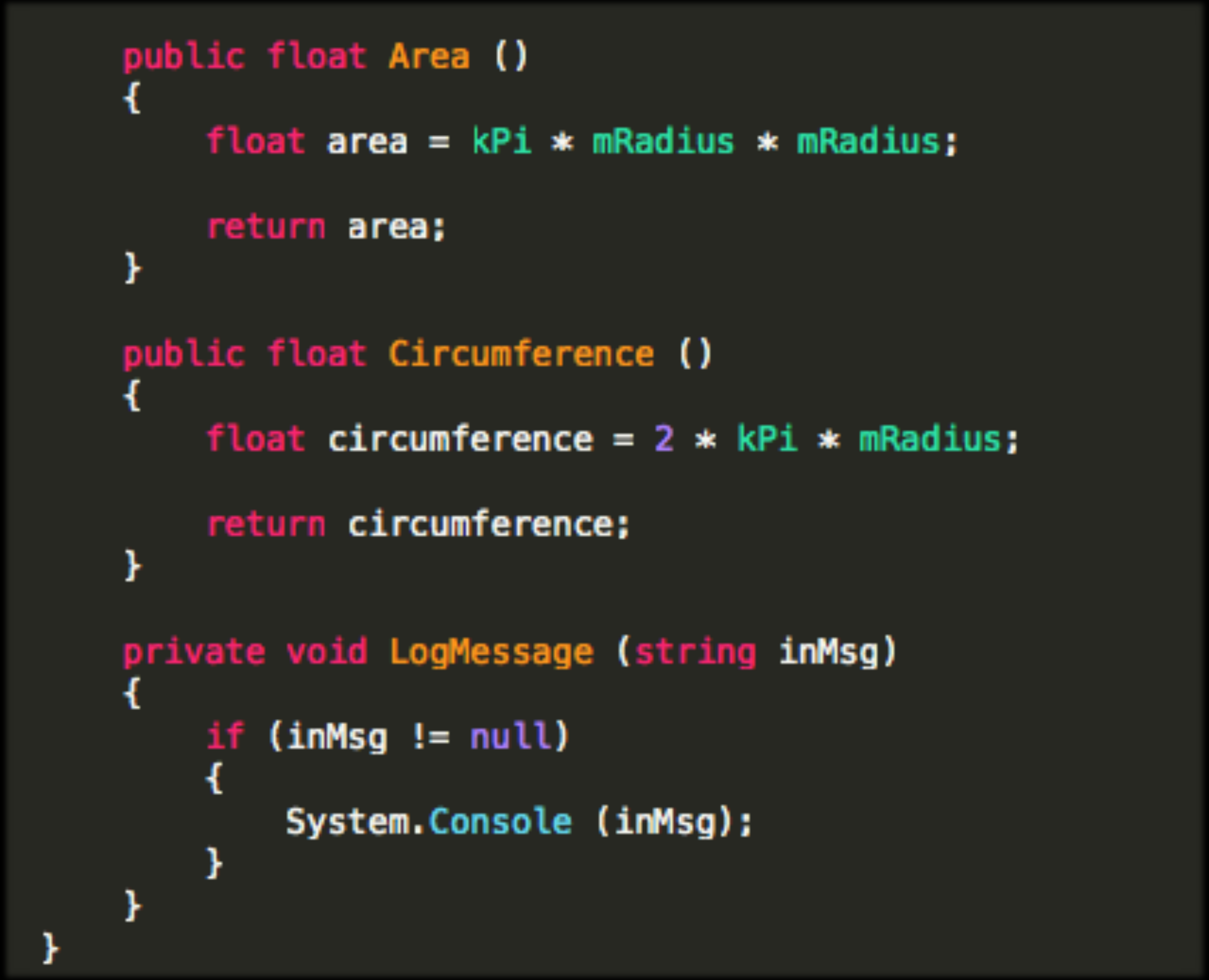
```
public class Circle
{
    private const float kPi = 3.14f;

    private float mRadius;

    public Circle (float inRadius)
    {
        SetRadius (inRadius);
    }

    public float GetRadius ()
    {
        return mRadius;
    }

    public void SetRadius (float inRadius)
    {
        if (inRadius > 0)
        {
            mRadius = inRadius;
        }
        else
        {
            LogMessage ("ERROR: Invalid Radius!");
        }
    }
```



```
    public float Area ()
    {
        float area = kPi * mRadius * mRadius;

        return area;
    }

    public float Circumference ()
    {
        float circumference = 2 * kPi * mRadius;

        return circumference;
    }

    private void LogMessage (string inMsg)
    {
        if (inMsg != null)
        {
            System.Console (inMsg);
        }
    }
}
```

What prevents Clean Code?

"I WILL CLEAN IT UP LATER."

Actually, LATER never comes.

Conclusion

- Better use of your time
- Easier Onboarding for New Team Members
- Easier Debugging
- More Efficient Maintenance
- You’ll feel good 😎

The Clean Coder Rule :

"ALWAYS LEAVE THE CODE CLEANER THAN YOU FOUND IT."

Thank You