

Defensive Coding in C#

WHY DEFENSIVE CODING MATTERS



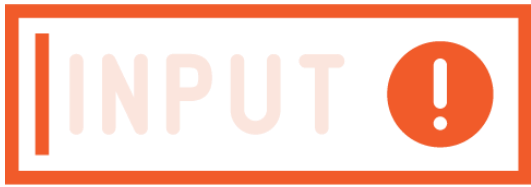
Deborah Kurata

CONSULTANT | SPEAKER | AUTHOR | MVP | GDE

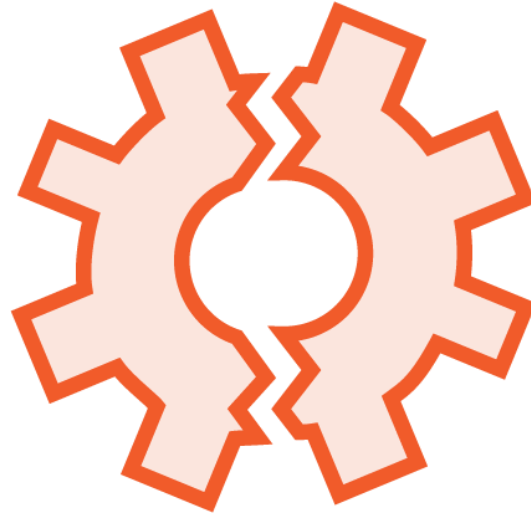
@deborahkurata | blogs.msmvps.com/deborahk/



What Are We Defending Our Code From?



**Incorrect
entry**



**Invalid
operations**



**System
mishaps**



**Future
developers**





Incorrect entry -> Add validation

Product Name

Name (required)



Product name is required.

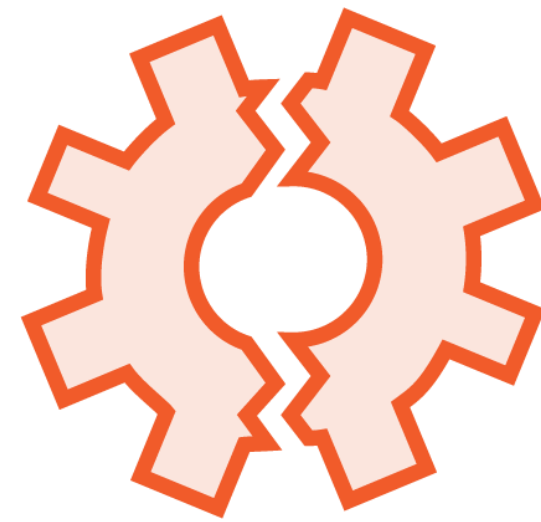
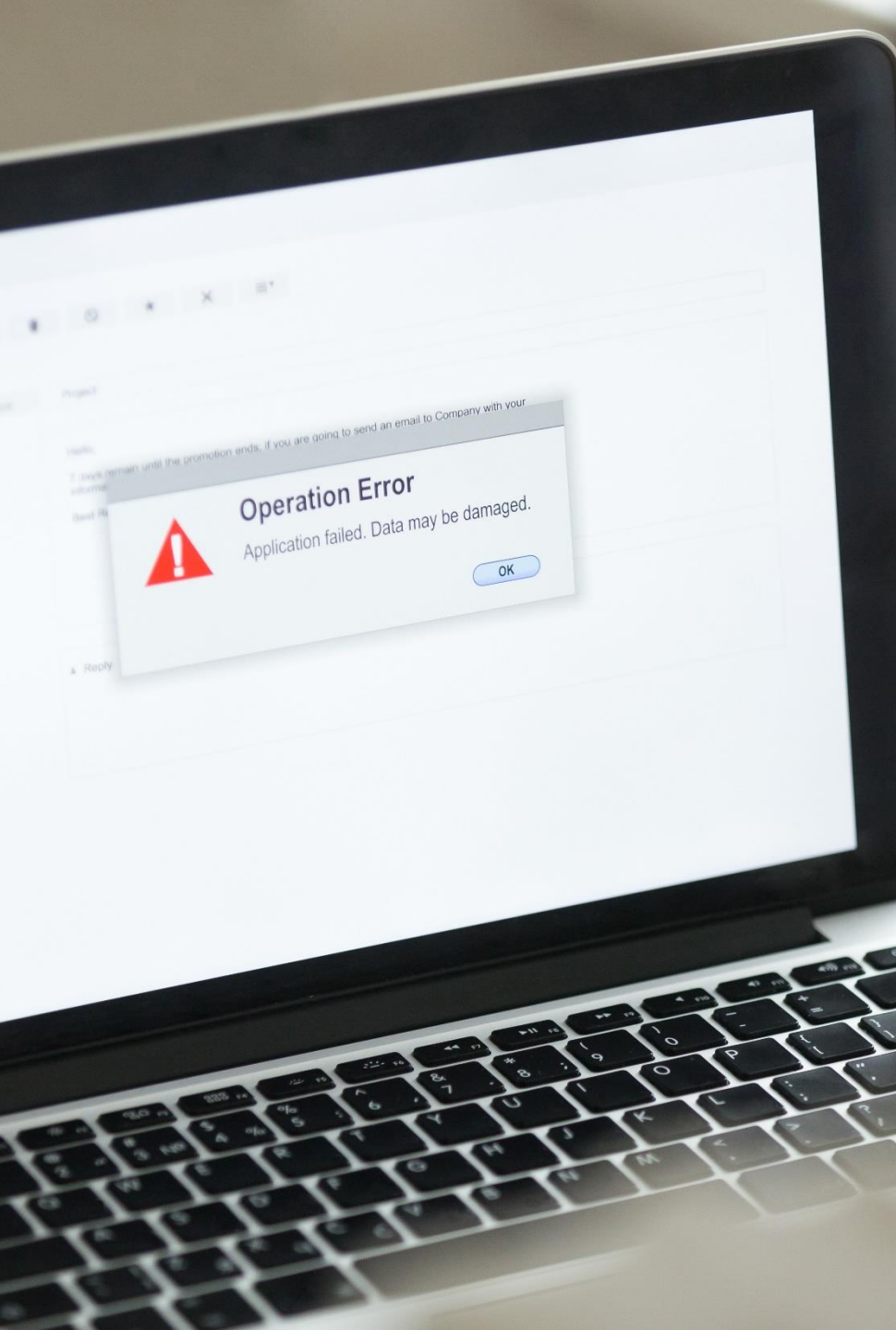
Product Name

Curved claw steel hammer with wooden handle



Product name cannot exceed 32 characters.





Invalid operations -> Check arguments
-> Unit test operations

$$\text{Profit Margin} = \frac{\text{Suggested Price} - \text{Cost}}{\text{Suggested Price}}$$





System mishaps -> Add checks
-> Manage exceptions





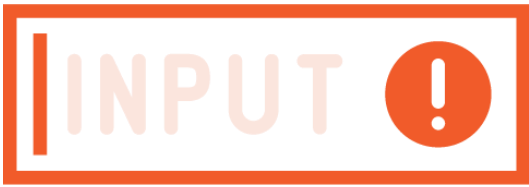
Here be dragons! -> Write clean code
-> Rerun unit tests



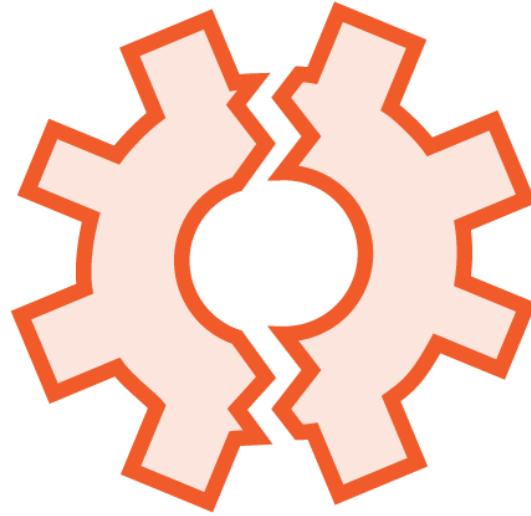
Image from the James Ford Bell Library, University of Minnesota



Malicious Users?



Incorrect
entry



Invalid
operations



System
mishaps



Future
developers



"Anything that **can** go wrong
will go wrong."

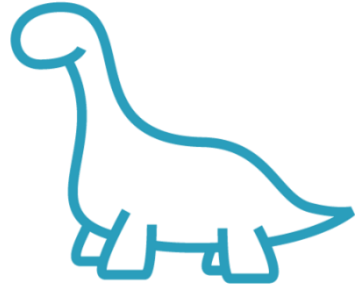
Murphy's Law



How Can We Write Great Code?



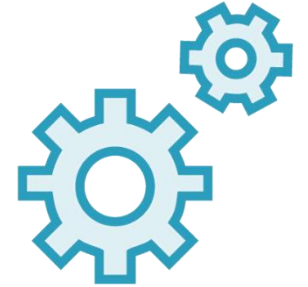
Changing
requirements



Legacy issues



Time pressures



Changing
environment



Maintenance



Multiple developers



Ravages of time



Defensive Programming (coding)

“is an approach to improve software and source code, in terms of:

- General **quality** – reducing the number of software bugs and problems.
- Making the source code **comprehensible** - the source code should be readable and understandable, so it is approved in a code audit.
- Making the software behave in a **predictable** manner despite unexpected inputs or user actions.”

- Wikipedia



Defensive Coding Helps Us Improve



Code comprehension



Code quality



Code predictability



Competing Goals



Improving code **predictability**
by **writing more** validation and
exception management code



Improving code **comprehension**
by **eliminating** unnecessary code



Protected, but Nimble



Not protected



Just right



Too protected

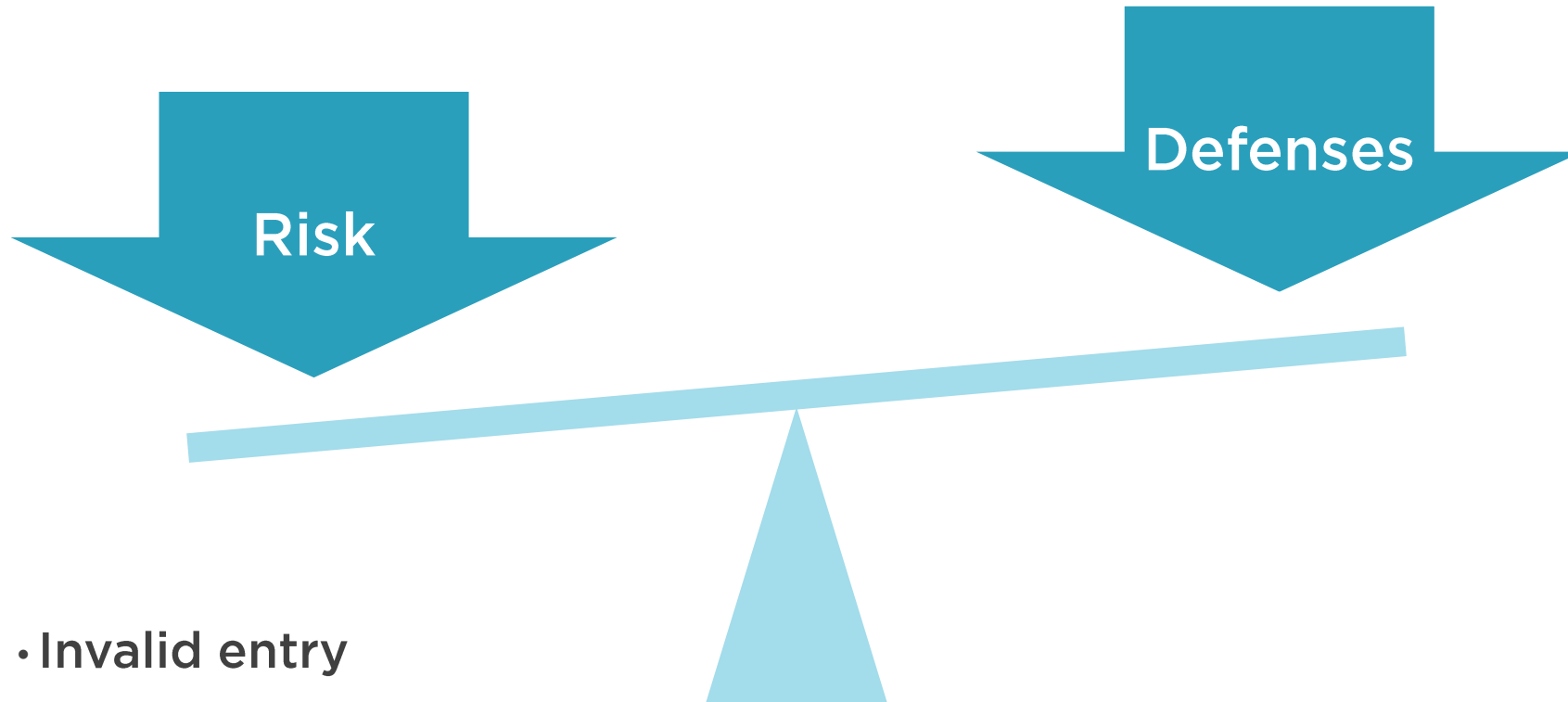
Fitness for a Purpose



Risk vs. Defenses

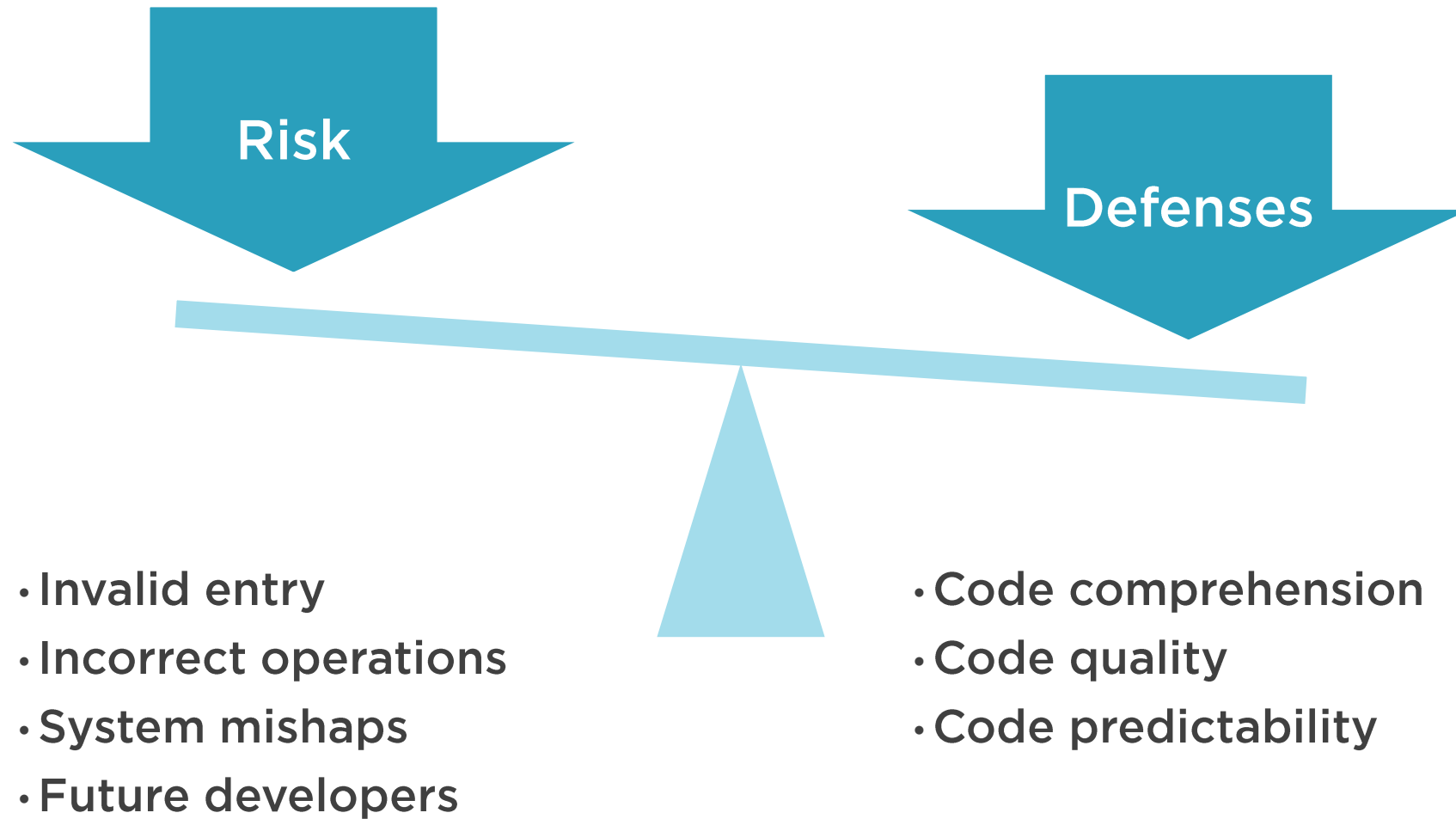


Risk vs. Defenses

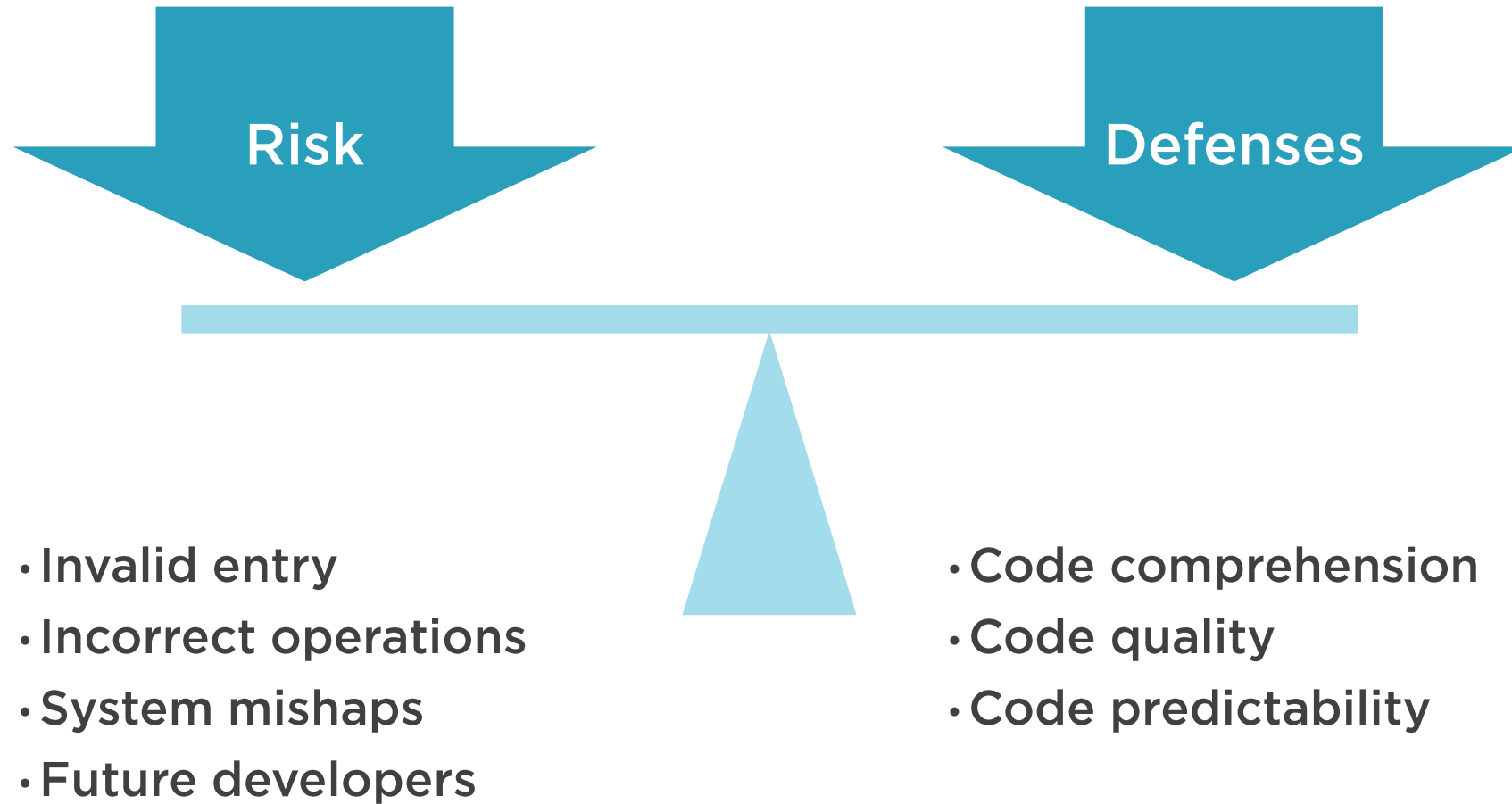


- Invalid entry
- Incorrect operations
- System mishaps
- Future developers

Risk vs. Defenses



Risk vs. Defenses



Prerequisites

Required

- **C# Basics**
- **Visual Studio**

Suggested

- **Unit testing**



DeborahK / CSharp-Defense

Unwatch

2

Star

4

Fork

2

Code

Issues0

Pull requests0

Actions

Projects0

Wiki

Security

Insights

Settings

Materials for the Defensive Coding in C# Pluralsight Course

Edit

Manage topics

20 commits

1 branch

0 packages

0 releases

2 contributors

MIT

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

DeborahK

Changed folder names.

Latest commit 23409b0 2 minutes ago

APM-Begin	Changed folder names.	2 minutes ago
APM-Final	Changed folder names.	2 minutes ago
APM-WithUI	First checkin of completed code with Windows UI	5 minutes ago
.gitignore	Initial commit	2 months ago
LICENSE	Initial commit	2 months ago

<https://github.com/DeborahK/CSharp-Defense>

Course Overview



Strengthening our defenses

- Code comprehension
- Code quality
- Code predictability

Validating method arguments

Handling nulls

Returning predictable results

Managing exceptions

