Defending Your Methods Part 3: Returning Predictable Results

Deborah Kurata
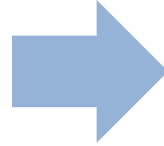http://msmvps.com/blogs/deborahk/
@DeborahKurata
deborahk@insteptech.com
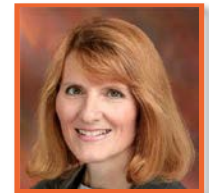
pluralsight
hardcore dev and IT training

Defensive Coding

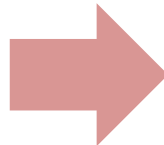| Clean Code	➜	• Improves Comprehension • Simplifies Maintenance • Reduces Bugs
Testable Code + Unit Tests	➜	• Improves Quality • Confirms Maintenance • Reduces Bugs
Validation + Exception Handling	➜	• Improves Predictability • More Consistent • Reduces Bugs
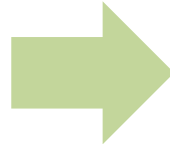
Method Results

Value

Exception(s)

Multiple Values

Null

Garbage In / Garbage Out

Examples

```
var result = customer.CalculatePercentOfGoalSteps(this.GoalTextBox.Text,
                                                   this.StepsTextBox.Text);


customer.ValidateEmail();


customerRepository.Add(customer);


var customer = customerRepository.Find(this.IdTextBox.Text);


var customerList = customerRepository.Find(this.NameTextBox.Text);
```

| Value | Exception(s) | Multiple Values | Null |

Returning Null

```
String message = null;


Nullable<decimal> result = null;
decimal? result = null;


Customer customer = null;


List<Customer> customerList = null;
```

Returning Null

```
var customer = customerRepository.Find(this.IdTextBox.Text);


var customerList = customerRepository.Find(this.NameTextBox.Text);
```

Summary

```
var result = customer.CalculatePercentOfGoalSteps(this.GoalTextBox.Text,
                                                  this.StepsTextBox.Text);


customer.ValidateEmail();


customerRepository.Add(customer);


var customer = customerRepository.Find(this.IdTextBox.Text);


var customerList = customerRepository.Find(this.NameTextBox.Text);
```

| Value | Exception(s) | Multiple Values | Null |