Applying Functional Principles in C#

INTRODUCTION

Vladimir Khorikov
PROGRAMMER

@vkhorikov www.enterprisecraftsmanship.com

Functional Programming Features

C# 3.0

LINQ, extension methods

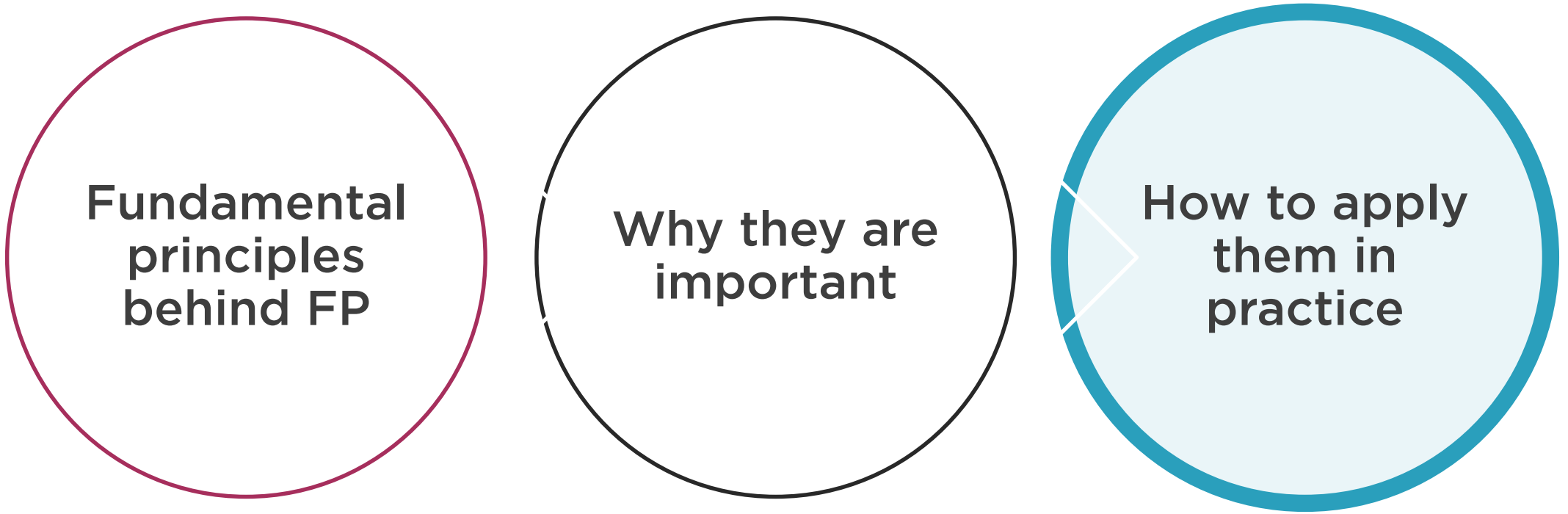
C# 6.0

Read-only properties

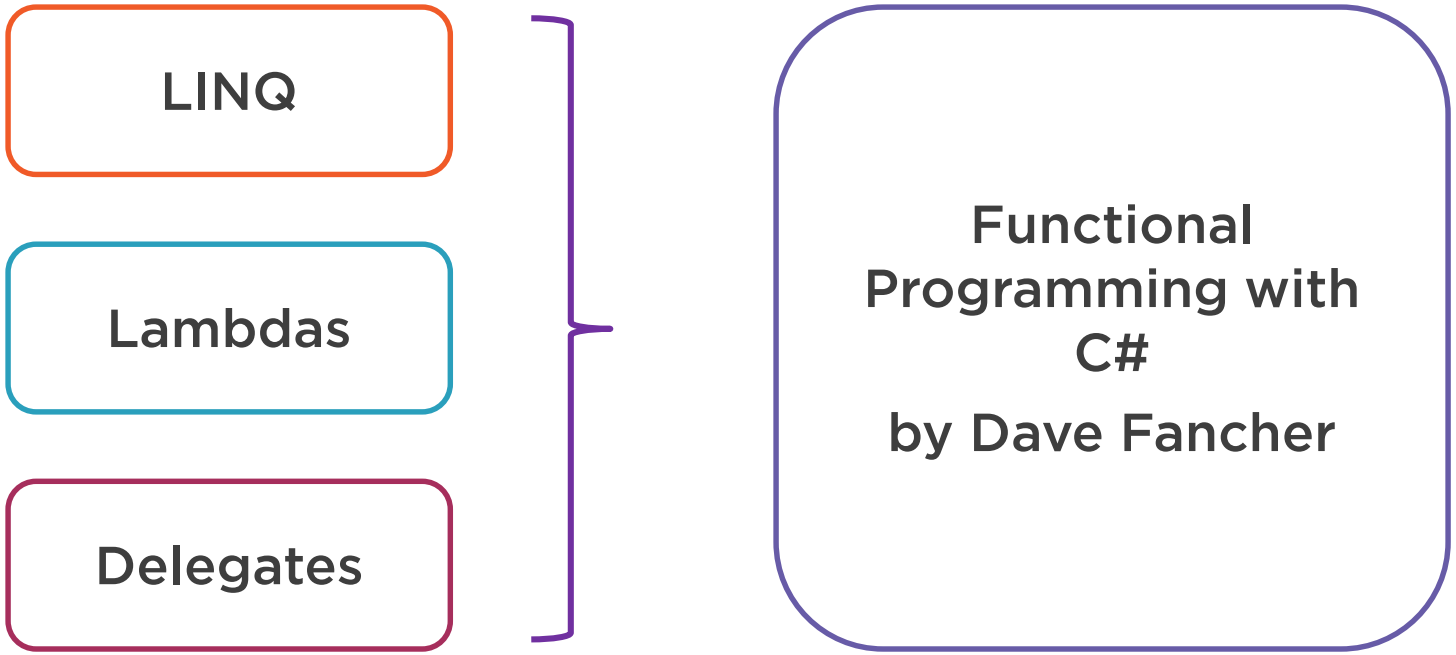
C# 7.0

Pattern matching?

The Purpose of This Course



Prerequisites



Overview

Introduction

Refactoring to an Immutable Architecture

Refactoring Away from Exceptions

Avoiding Primitive Obsession

Avoiding Nulls with the Maybe Type

Handling Failures and Input Errors in a Functional Way

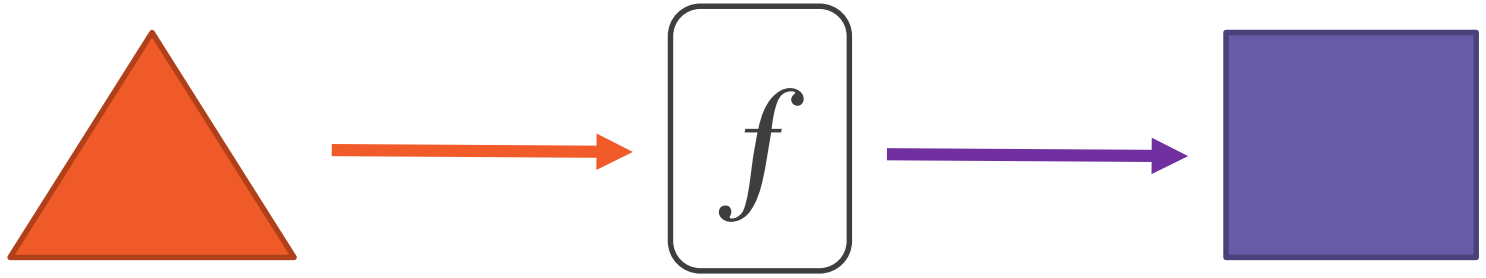
Putting It All Together

Functional programming is programming with mathematical functions.

What Is Functional Programming?

Mathematical function  **Class method**

What Is Functional Programming?



- Same input – same result
- Information about possible inputs and outcomes

What Is Functional Programming?

```
public double Calculate(double x, double y)
{
    return x * x + y * y;
}
```

Same input – same result

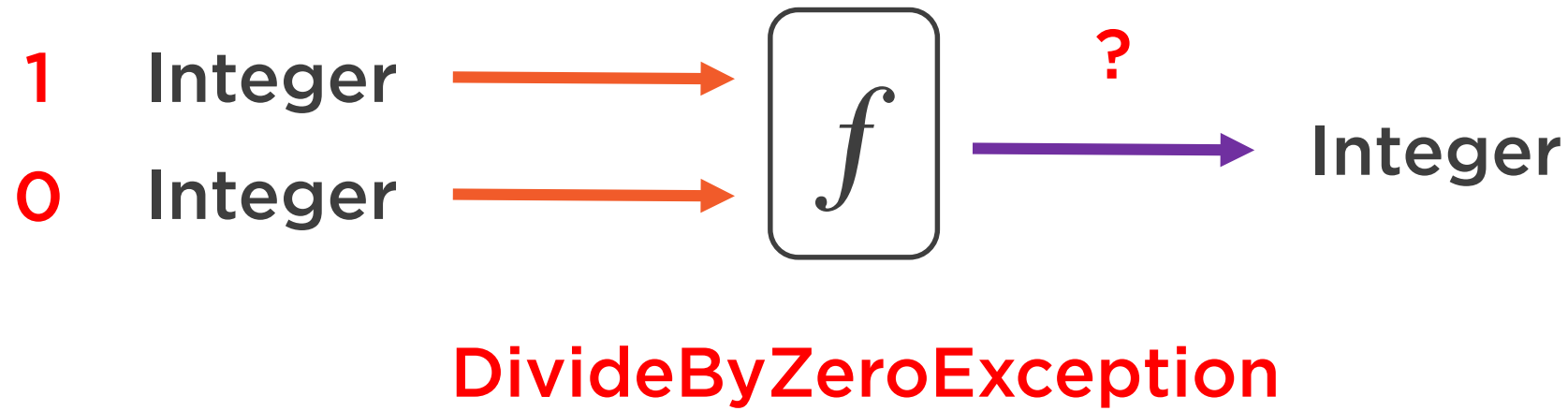
```
public long TicksElapsedFrom(int year)
{
    DateTime now = DateTime.Now;
    DateTime then = new DateTime(year, 1, 1);

    return (now - then).Ticks;
}
```

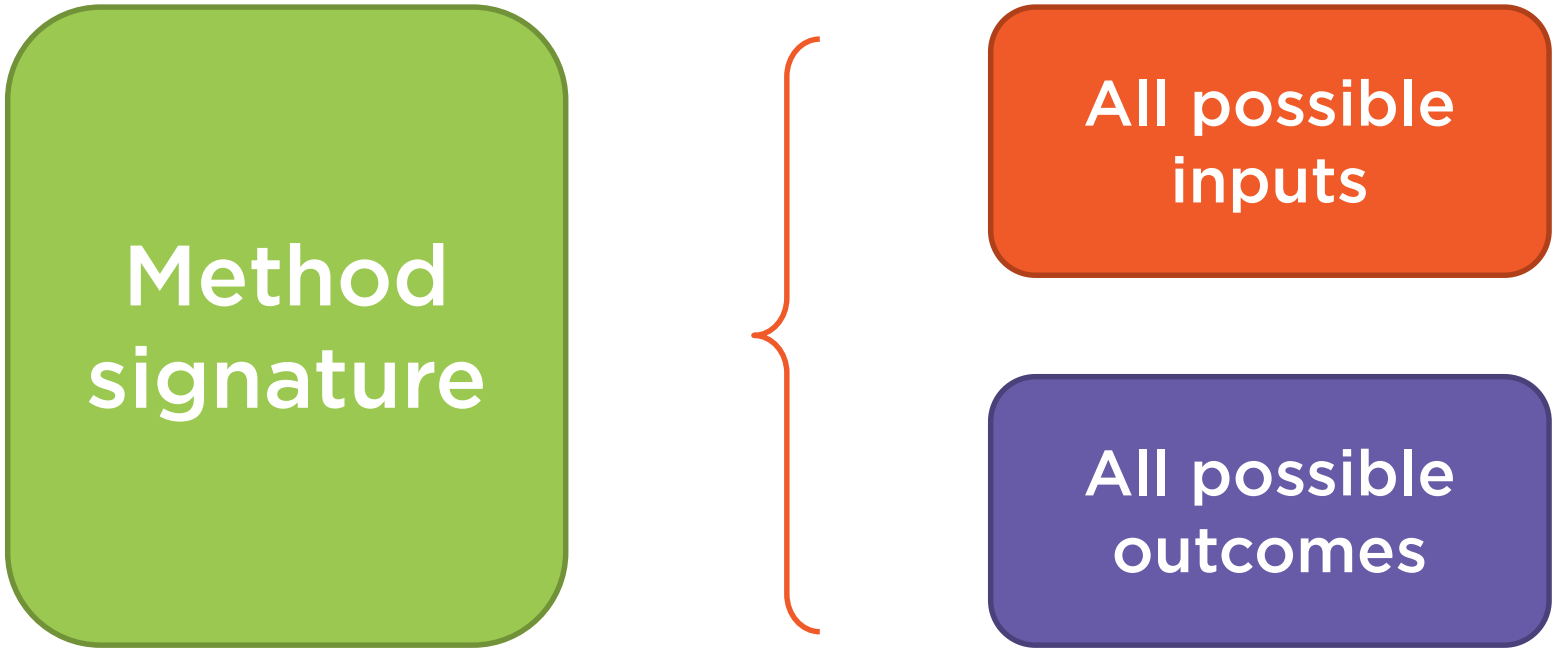
Result is always different

What Is Functional Programming?

```
public static int Divide(int x, int y)
{
    return x / y;
}
```



Method Signature Honesty



Method Signature Honesty

```
public static int Divide(int x, int y)
{
    return x / y;
}
```

Dishonest signature

```
public static int Divide(int x, NonZeroInteger y)
{
    return x / y.Value;
}
```

```
public static int? Divide(int x, int y)
{
    if (y == 0)
        return null;

    return x / y;
}
```

Honest signature

Mathematical Function

Honest

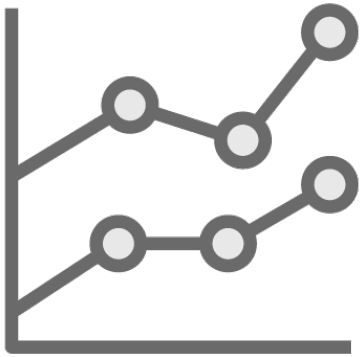
Has precisely defined input and output

Referentially transparent

Doesn't affect or refer to the global state

Why Functional Programming?

Complexity

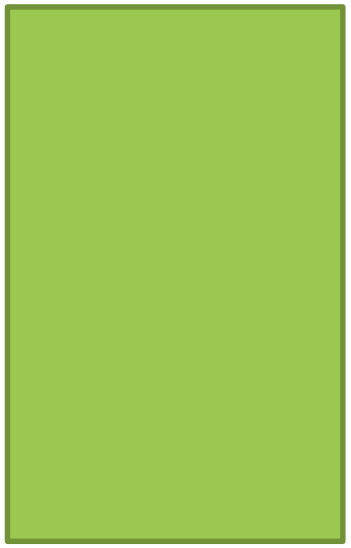


Development speed

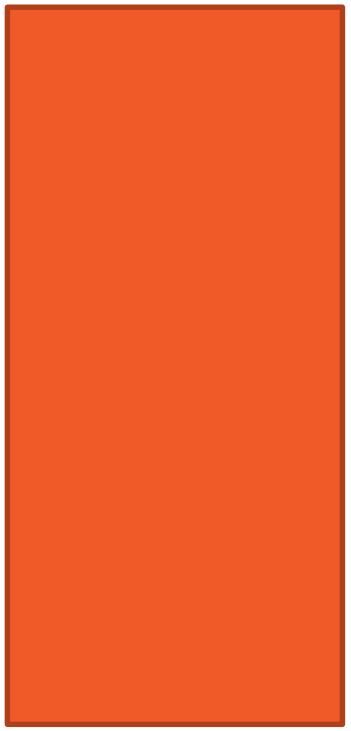
Number of bugs

Agility

Why Functional Programming?

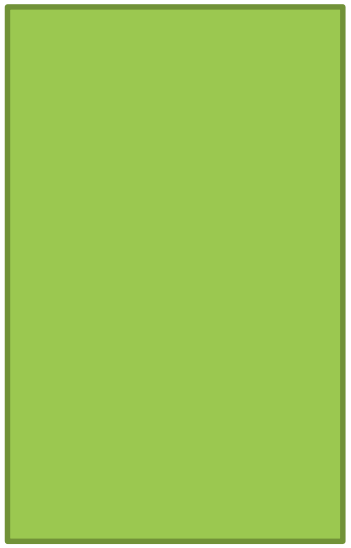


Brain capacity

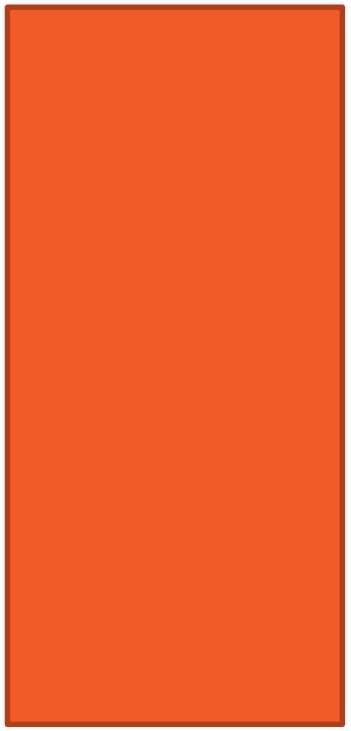


Code complexity

Why Functional Programming?

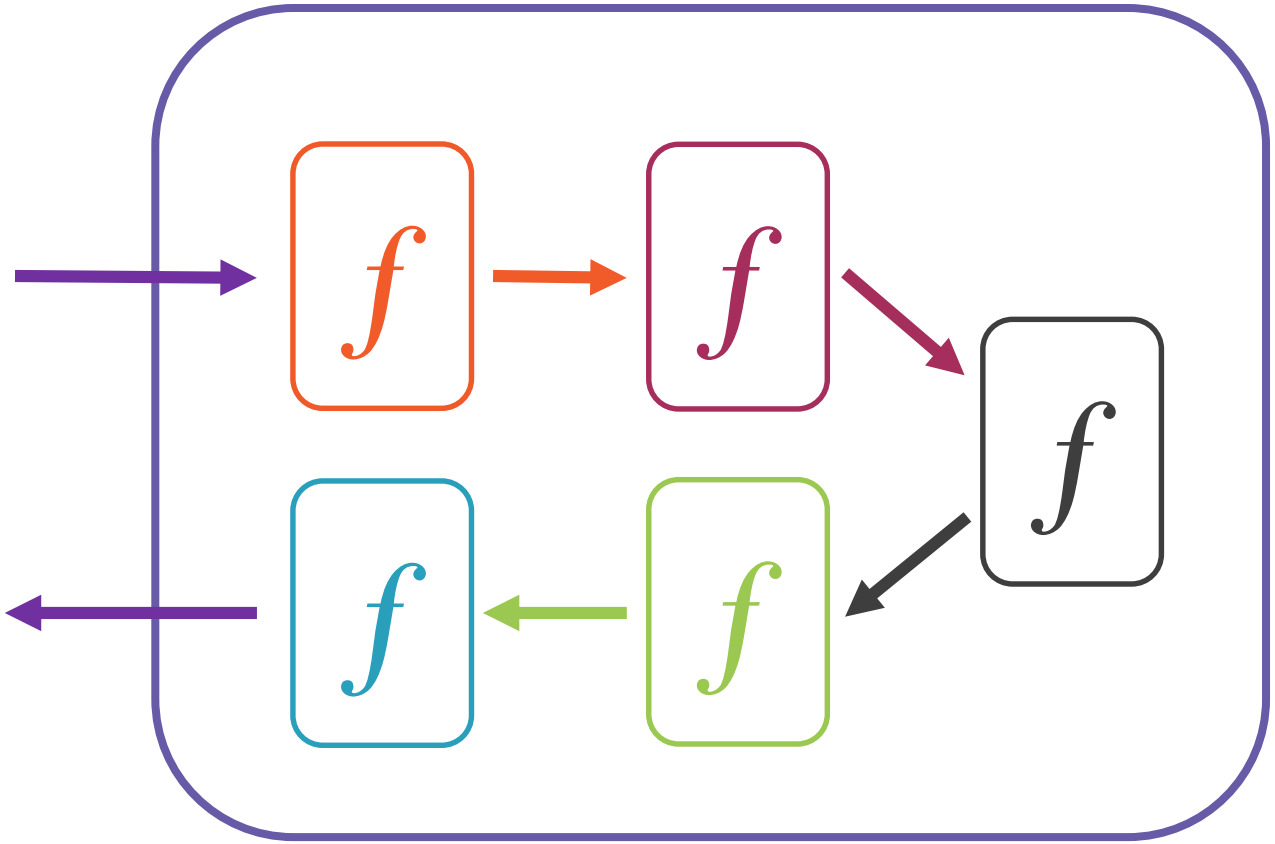


Brain capacity



Code complexity

Why Functional Programming?



- Composable
- Easy to reason about
- Easier to unit test

Summary

Functional programming is programming with mathematical functions

Mathematical functions

- Defined using an honest signature
- Don’t refer to the global state

Functional programming helps reduce code complexity

In the Next Module

Immutability