C# Generics

Why Generics?

K. Scott Allen
@OdeToCode

pluralsight
hardcore developer training

Generics

- **Generics types allow code reuse with type safety**
  - Defer type specification to client
  - Internal algorithms remain the same

```
public class CircularBuffer<T>
{
    private T[] _buffer;

    // ...
}
```

Terminology

```
public class CircularBuffer<T>
{
    var b1 = new CircularBuffer<string>();
    var b2 = new CircularBuffer<string>();
    var b3 = new CircularBuffer<int>();
    var b4 = new CircularBuffer<object>();

    Console.WriteLine(b1.GetType() == b2.GetType()); // true
    Console.WriteLine(b1.GetType() == b3.GetType()); // false
    Console.WriteLine(b1.GetType() == b4.GetType()); // false
}
        var buffer = new CircularBuffer<double>();
```

Summary

- **Generics types allow code reuse with type safety**

```
public class CircularBuffer<T>
{
    private T[] _buffer;

    // ...
}
```