

Fast Focus: Safer Code Nullability and Null Operators in C#

Jeremy Clark
Developer Betterer
jeremybytes.com

Level: Introductory / Intermediate

Session Survey

- Your feedback is very important to us
- Please take a moment to complete the session survey found in the mobile app
- Use the QR code or search for “Converge360 Events” in your app store
- Find this session on the Agenda tab
- Click “Session Evaluation”
- Thank you!



Fast Focus: Safer Code Nullability and Null Operators in C#

Jeremy Clark
Developer Betterer
jeremybytes.com

Level: Introductory / Intermediate



What Nullability Is

- A way to get *compile-time* warnings about possible null references.
- A way to make the intent of your code more clear.

What Nullability Is Not

- NOT a way to prevent null reference exceptions at runtime.
- NOT a way to prevent someone from passing a null to your method.
- NOT a way to prevent someone from assigning a null to an object.

Enabling Nullability*

- Project Level

```
<Nullable>enable</Nullable>
```

- Code Level

```
#nullable enable
```

* More correctly: enabling non-nullability

Nullability Operators

? . / ?[] (Null Conditional Operators)

```
tokenSource?.Cancel();
```

! (Null Forgiving Operator)

```
task.Exception!.Flatten().InnerExceptions
```

?? / ??= (Null Coalescing Operators)

```
return result ?? new List<Person>();
```

A Note about “var”

- Using “var” results in a nullable type.

```
var people = new List<Person>();
```

 (local variable) List<Person>? people



Session Survey

- Your feedback is very important to us
- Please take a moment to complete the session survey found in the mobile app
- Use the QR code or search for “Converge360 Events” in your app store
- Find this session on the Agenda tab
- Click “Session Evaluation”
- Thank you!





Thank You!

Jeremy Clark

- jeremybytes.com
- jeremy@jeremybytes.com
- [@jeremybytes](https://twitter.com/jeremybytes)

<https://github.com/jeremybytes/vslive2024-microsoft>