Visual Studio LIVE!
EXPERT SOLUTIONS FOR ENTERPRISE DEVELOPERS

@ Microsoft HQ

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

Jeremy Clark
Developer Betterer
jeremybytes.com

Level: Introductory / Intermediate

#VSLIVE

Your Code Powers the World.
Our Training Powers You.

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.

©Jeremy Clark 2023

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.

©Jeremy Clark 2023

Enabling Nullability*

- Project Level

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

- Code Level

```
#nullable enable
```

* More correctly: enabling non-nullability

©Jeremy Clark 2023

Nullability Operators

`?.` / `?[]` (Null Conditional Operators)

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

`!` (Null Forgiving Operator)

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

`??` / `??=` (Null Coalescing Operators)

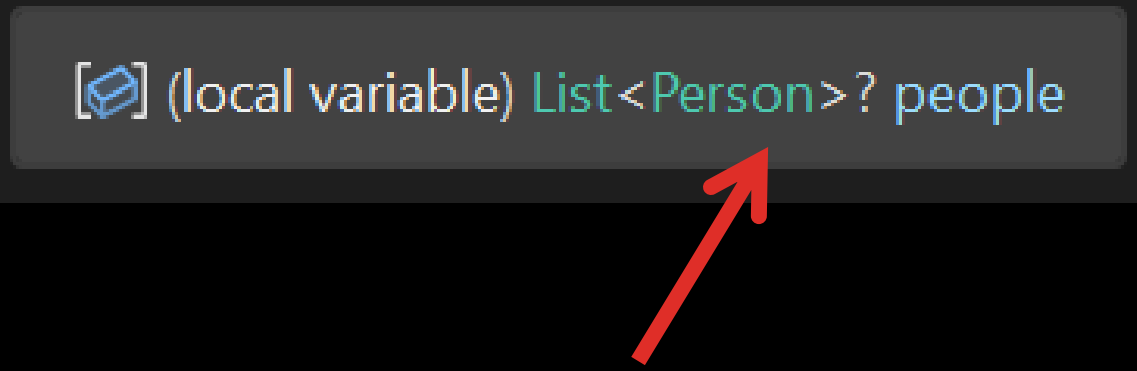
```
return result ?? new List<Person>();
```

©Jeremy Clark 2023

A Note about “var”

- Using “var” results in a nullable type.

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



©Jeremy Clark 2023

Thank You!

Jeremy Clark

- jeremybytes.com
- jeremy@jeremybytes.com
- @jeremybytes

https://github.com/jeremybytes/vslive2024-microsofthq