

Nullable Reference Types



Matt Honeycutt

HEROIC CONSULTING

@matthoneycutt www.trycatchfail.com



Overview



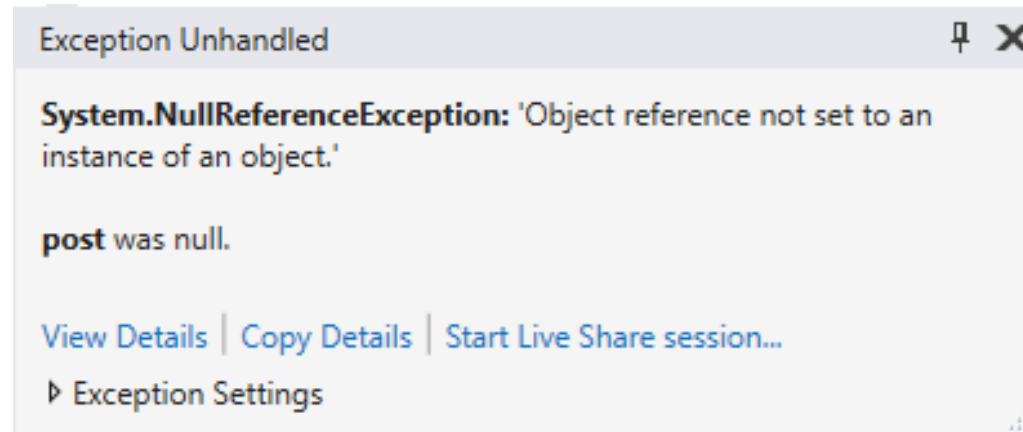
Why nullable reference types matter

Using nullable reference types

Improving existing code



Null Reference Exceptions



Null Reference Exceptions

Simple code...

```
static void PrintTitle(BlogPost post)
{
    Console.WriteLine(post.Title);
}
```

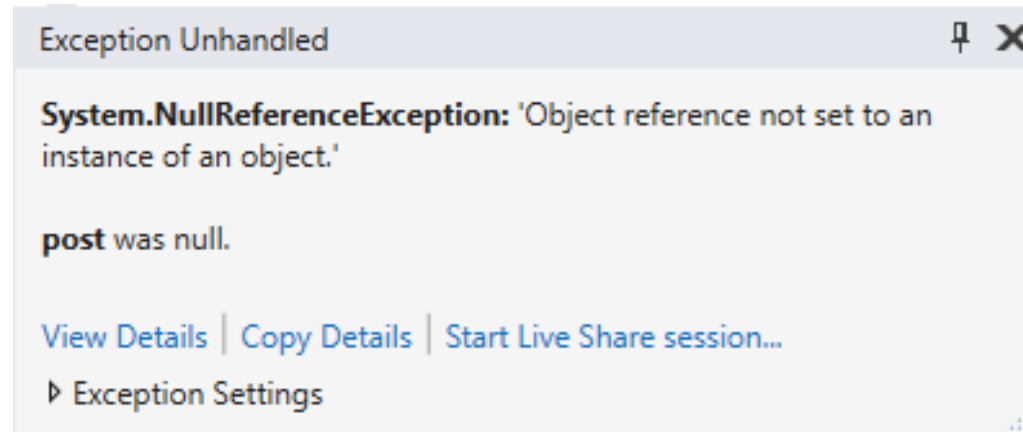
Called the wrong way...

```
//Oops!
post = null;

PrintTitle(post);
```



Null Reference Exceptions



Avoiding null reference
errors in complex
applications is hard



Nullable Reference Types

Avoid null
reference mistakes
at compile-time

Does not remove
null values from
our code

Enables
expressing *intent*



Nullable Reference Types in Action

```
static void PrintTitle(BlogPost post)
{
    Console.WriteLine(post.Title);
}
```



Assumes *post* is not null

```
BlogPost post = null;
PrintTitle(post);
```



C# 7: this compiles with
zero warnings!



Nullable Reference Types in Action

```
static void PrintTitle(BlogPost post)
{
    Console.WriteLine(post.Title);
}
```



Assumes *post* is not null

```
BlogPost post = null;
PrintTitle(post);
```



C# 8: now we get a warning!



[E] (local variable) `BlogPost post`

Possible null reference argument for parameter 'post' in 'void Program.PrintTitle(BlogPost post)'.

[Show potential fixes](#) (Alt+Enter or Ctrl+.)



****BREAK****



Demo



Creating a new project

- Using Visual Studio 2019

Use nullable reference types

Prerequisite: .NET Core 3.0 SDK

- <https://dotnet.microsoft.com/download/dotnet-core/3.0>



Demo



Applying nullable reference types to an existing project

Want to follow along?

- Use the sample project included in the exercise files



Summary



Nullable reference types basics

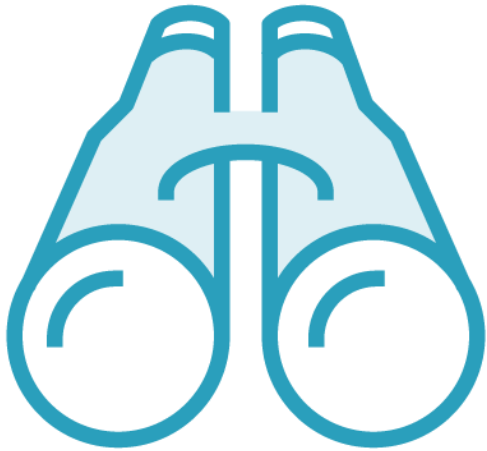
Enabling nullable reference types on an existing project

Finding null reference errors

Remember: you still need to watch out for null-reference errors!



Up Next



Pattern matching improvements

