Accessing and Checking for Null Values

Jason Roberts
.NET DEVELOPER

@robertsjason dontcodetired.com

Overview

More on Nullable<T>

Convenience properties and methods

Comparing Nullable<T> instances

Implicit and explicit conversions to/from Nullable<T>

C# operators for working with nulls

General purpose conditional operator

Null-coalescing operator

The null-coalescing assignment operator

Null-conditional operator

Thread-safe null delegate invocation

More on Nullable<T>

```
.HasValue // false if null, otherwise true
.Value // gets underlying value
.GetValueOrDefault() // underlying value or default
.GetValueOrDefault(default) // value or specified default
```

Comparing Nullable<T> Instances

```
int? i = 42;
int? j = 42;

bool areEqual = i == j; // true
```

Comparing Nullable<T> Instances

```
int? i = 42;
int? j = null;

bool areEqual = i == j; // false
```

Comparing Nullable<T> Instances

```
int? i = null;
int? j = null;

bool areEqual = i == j; // true
```

Nullable<T> Conversions

```
// Implicit conversion from T --> Nullable<T>

int i = 42;
int? j = i; // no explicit casting/conversion required
```

Nullable<T> Conversions

```
// Explicit conversion required from Nullable<T> to T


int? i = 42;

int j = i; // Compiler error, no implicit conversion

int j = (int)i; // explicit cast


int? i = null;

int j = (int)i; // Runtime InvalidOperationException
```

Default Values for Nullable Value Types

```
int? i;
Console.WriteLine(i); // Use of unassigned local variable

int? i = default; // i == null
int? i = default(int); // i == 0
bool? b = default; // b == null
bool? b = default(bool); // b == false
```

Overview of C# Null-related Operators

Conditional operator

?:

Null-coalescing operator

??

Null-coalescing assignment operator

??=

Null-conditional operator

?.   ?[]

Null-forgiving operator

!

The Null-coalescing Assignment Operator

```
string name = Console.ReadLine();

if (name is null) // name == null
{

    name = "No name entered";
}

Console.WriteLine(name);
```

The Null-coalescing Assignment Operator

```
string name = Console.ReadLine();

name ??= "No name entered"; // from C# 8

Console.WriteLine(name);
```

The Null-coalescing Assignment Operator

```
string name = Console.ReadLine();

name ??= "No name entered"; // from C# 8

Console.WriteLine(name);
```

Thread-Safe Null Delegate Invocation

```
public event EventHandler NameChanged;

...

EventHandler eventHandler = NameChanged;

if (eventHandler != null)
{
    eventHandler(this, EventArgs.Empty);
}
```

Thread-Safe Null Delegate Invocation

```
public event EventHandler NameChanged;

...

EventHandler eventHandler = NameChanged;

if (eventHandler != null)
{

    eventHandler(this, EventArgs.Empty);
}
```

Thread-Safe Null Delegate Invocation

```
public event EventHandler NameChanged;

...

NameChanged?.Invoke(this, EventArgs.Empty);
```

Thread-Safe Null Delegate Invocation

```
public event EventHandler NameChanged;

...

NameChanged?.Invoke(this, EventArgs.Empty);
```

Summary

HasValue, Value, & GetValueOrDefault()

Comparing Nullable<T> instances

Conversions to/from Nullable<T>

C# operators for working with nulls

Conditional operator ? :

Null-coalescing operator ??

Null-coalescing assignment operator ??=

Null-conditional operators ?. ?[

NameChanged?.Invoke(...)

Next:

Eliminating Null Reference Exceptions with the Null Object Pattern