

Errors

```
private string? Value
{
    get => this._ myC lass?
        .Value;

    set
    {
        _ = this._ myClass
        ?? Errors
            .Un abl eTo Ass ign Value()
            thi s._ myC las s.Value = value;
    }
}
```

Property Wertübergabe, Errors wirft Exception

Exceptions1

```
try
{
    using var stream = new FileStream(path, FileMode.Open);
    using var reader = new StreamReader(stream);
    ...
}
catch (Exception ex)
{
    ex.Data["exceptionMeta.InvariantMessage.StrValue"] = $"Error on accessing path '{path}'.";
    ex.Data["path"] = path;
    throw;
}
```

throw-catch

ThrowIf

```
value.ThrowIf();
//Bsp
value.T hr owI fNo tDi rec tor -
yPa th();
value.T hr owI fNu llO rWh ite -
space;
//usw.
```

Je nach Datentyp unterschiedliche ThrowIf-Methoden. Erleichtert Exception Throwing

windowspath (copy)

```
Consol e.W rit eli ne( " Hello
World! ");
```

Hello world

Preconditions

```
public string Value {get;}
=Preco ndi tio ns.C he ckI fNu -
ll( value)
```

Übergeben von Konstruktorparametern nach Null-Check ggf. an Propertiers



By **lucasH**

cheatography.com/lucash/

Not published yet.

Last updated 30th October, 2024.

Page 2 of 2.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>

