

# System.TypeInitializationException Class

```
[ILAsm]
.class public sealed serializable TypeInitializationException
extends System.SystemException

[C#]
public sealed class TypeInitializationException : SystemException
```

## Assembly Info:

- *Name:* mscorlib
- *Public Key:* [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- *Version:* 2.0.x.x
- *Attributes:*
  - CLSCompliantAttribute(true)

## Summary

Represents the error that occurs when an exception is thrown inside the static constructor of a type.

## Inherits From: System.SystemException

**Library:** BCL

**Thread Safety:** All public static members of this type are safe for multithreaded operations. No instance members are guaranteed to be thread safe.

## Description

When a static constructor fails to initialize a type, a `System.TypeInitializationException` instance is created and passed a reference to the exception thrown by the static constructor. The `System.TypeInitializationException.InnerException` property stores the exception that was thrown by the static constructor.

## Example

The following example demonstrates an error that causes a `System.TypeInitializationException` exception.

```
[C#]

using System;
class TypeInit {
    // Static constructor
    static TypeInit () {
```

```
        // Throw an application-defined exception.
        throw new ApplicationException("Error in Class TypeInit");
    }
    public TypeInit() {}
}
class TestTypeInit {
    static public void Main() {
        try {
            TypeInit t = new TypeInit ();
        }
        catch (TypeInitializationException e) {
            Console.WriteLine("Error: {0}",e);
        }
    }
}
```

The output is

```
Error: System.TypeInitializationException: The type initializer for
"TypeInit" threw an exception. ---> System.ApplicationException: Error
in Class TypeInit
    at TypeInit..ctor()
    --- End of inner exception stack trace ---
    at TypeInit..ctor()
    at TestTypeInit.Main()
```

# TypeInitializationException.TypeName Property

```
[ILAsm]  
.property string TypeName { public hidebysig specialname instance  
string get_TypeName() }  
  
[C#]  
public string TypeName { get; }
```

## Summary

Gets the fully qualified name of the type that causes the exception.

## Property Value

A `System.String` that contains the fully qualified name of the type that caused the exception.

## Description

This property is read-only.

The `System.TypeInitializationException.TypeName` property returns the same value as was passed into the constructor.