How to create user-defined exceptions

03/30/2017 • 2 minutes to read • +5

In this article

See also

.NET provides a hierarchy of exception classes ultimately derived from the base class Exception. However, if none of the predefined exceptions meets your needs, you can create your own exception classes by deriving from the Exception class.

When creating your own exceptions, end the class name of the user-defined exception with the word "Exception", and implement the three common constructors, as shown in the following example. The example defines a new exception class named `EmployeeListNotFoundException`. The class is derived from Exception and includes three constructors.

C# Copy

```csharp
using System;

public class EmployeeListNotFoundException : Exception
{
    public EmployeeListNotFoundException()
    {
    }

    public EmployeeListNotFoundException(string message)
        : base(message)
    {
    }

    public EmployeeListNotFoundException(string message, Exception inner)
        : base(message, inner)
    {
    }
}
```

ⓘ Note

In situations where you are using remoting, you must ensure that the metadata for any user-defined exceptions is available at the server (callee) and to the client (the proxy object or caller). For more information, see **Best practices for exceptions**.

See also

- Exceptions

Is this page helpful?

👍 Yes 👎 No

Recommended content

Named and Optional Arguments - C# Programming Guide

Named arguments in C# specify arguments by name, not position. Optional arguments can be omitted.

?? and ??= operators - C# reference

Learn about ?? and ??= which are the C# null-coalescing operators.

How to create user-defined exceptions with localized exception messages

Learn how to create user-defined exceptions with localized exception messages

?: operator - C# reference

Learn about the C# ternary conditional operator that returns the result of one of the two expressions based on a Boolean expression's result.

Show more ⌄