

# System.ArgumentException Class

```
[ILAsm]
.class public serializable ArgumentException extends
System.SystemException

[C#]
public class ArgumentException: SystemException
```

## Assembly Info:

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

## Summary

Represents the error that occurs when an argument passed to a method is invalid.

## 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

`System.ArgumentException` is thrown when a method is invoked and at least one of the passed arguments does not meet the method's parameter specification.

[*Note:* The Base Class Library includes three derived types:

- `System.ArgumentNullException`
- `System.ArgumentOutOfRangeException`
- `System.DuplicateWaitObjectException`

When appropriate, use these types instead of `System.ArgumentException`.

]

## 1 Example

2  
3 The following example demonstrates an error that causes a `System.ArgumentException`  
4 exception to be thrown by the system.

5  
6 [C#]

```
7 using System;
8 public class MyClass {}
9 public class ArgExceptionExample {
10     public static void Main() {
11         MyClass my = new MyClass();
12         string s = "sometext";
13         try {
14             int i = s.CompareTo(my);
15         }
16         catch (ArgumentException e) {
17             Console.WriteLine("Error: {0}", e);
18         }
19     }
20 }
```

21  
22 The output is

23  
24 Error: System.ArgumentException: Object must be of type String.

25  
26  
27 at System.String.CompareTo(Object value)

28  
29  
30 at ArgExceptionExample.Main()

# ArgumentException() Constructor

```
[ILAsm]  
public rtspecialname specialname instance void .ctor()  
  
[C#]  
public ArgumentException()
```

## Summary

Constructs and initializes a new instance of the `System.ArgumentException` class.

## Description

This constructor initializes the `System.ArgumentException.Message` property of the new instance to a system-supplied message that describes the error, such as "An invalid argument was specified." This message takes into account the current system culture.

The `System.ArgumentException.InnerException` property is initialized to `null`.

# ArgumentException(System.String) Constructor

```
[ILAsm]  
public rtspecialname specialname instance void .ctor(string message)  
  
[C#]  
public ArgumentException(string message)
```

## Summary

Constructs and initializes a new instance of the `System.ArgumentException` class.

## Parameters

Parameter	Description
<i>message</i>	A <code>System.String</code> that describes the error. The content of <i>message</i> is intended to be understood by humans. The caller of this constructor is required to ensure that this string has been localized for the current system culture.

## Description

This constructor initializes the `System.ArgumentException.Message` property of the new instance using *message*. If *message* is null, the `System.ArgumentException.Message` property is initialized to the system-supplied message provided by the constructor that takes no arguments. The `System.ArgumentException.InnerException` and `System.ArgumentException.ParamName` properties are initialized to null.

# ArgumentException(System.String, System.Exception) Constructor

```
[ILAsm]
public rtspecialname specialname instance void .ctor(string message, class
System.Exception innerException)

[C#]
public ArgumentException(string message, Exception innerException)
```

## Summary

Constructs and initializes a new instance of the `System.ArgumentException` class.

## Parameters

Parameter	Description
<i>message</i>	A <code>System.String</code> that describes the error. The content of <i>message</i> is intended to be understood by humans. The caller of this constructor is required to ensure that this string has been localized for the current system culture.
<i>innerException</i>	An instance of <code>System.Exception</code> that is the cause of the current <code>Exception</code> . If <i>innerException</i> is non-null, then the current <code>Exception</code> was raised in a catch block handling <i>innerException</i> .

## Description

This constructor initializes the `System.ArgumentException.Message` property of the new instance using *message*, and the `System.ArgumentException.InnerException` property using *innerException*. If *message* is null, the `System.ArgumentException.Message` property is initialized to the system-supplied message provided by the constructor that takes no arguments. The `System.ArgumentException.ParamName` property is initialized to null.

[*Note:* For information on inner exceptions, see `System.Exception.InnerException`.]

# ArgumentException(System.String, System.String, System.Exception) Constructor

```
[ILAsm]
public rtspecialname specialname instance void .ctor(string message,
string paramName, class System.Exception innerException)

[C#]
public ArgumentException(string message, string paramName, Exception
innerException)
```

## Summary

Constructs and initializes a new instance of the `System.ArgumentException` class.

## Parameters

Parameter	Description
<i>message</i>	A <code>System.String</code> that describes the error. The content of <i>message</i> is intended to be understood by humans. The caller of this constructor is required to ensure that this string has been localized for the current system culture.
<i>paramName</i>	A <code>System.String</code> that contains the name of the parameter that caused the current exception.
<i>innerException</i>	An instance of <code>System.Exception</code> that is the cause of the current <code>Exception</code> . If <i>innerException</i> is non-null, then the current <code>Exception</code> was raised in a catch block handling <i>innerException</i> .

## Description

This constructor initializes the `System.ArgumentException.Message` property of the new instance using *message*, the `System.ArgumentException.ParamName` property using *paramName*, and the `System.ArgumentException.InnerException` property using *innerException*. If *message* is null, the `System.ArgumentException.Message` property is initialized to the system-supplied message provided by the constructor that takes no arguments.

[Note: For information on inner exceptions, see `System.Exception.InnerException`.]

1  
2  
3

# ArgumentException(System.String, System.String) Constructor

```
[ILAsm]
public rtspecialname specialname instance void .ctor(string message,
string paramName)

[C#]
public ArgumentException(string message, string paramName)
```

## Summary

Constructs and initializes a new instance of the `System.ArgumentException` class.

## Parameters

Parameter	Description
<i>message</i>	A <code>System.String</code> that describes the error. The content of <i>message</i> is intended to be understood by humans. The caller of this constructor is required to ensure that this string has been localized for the current system culture.
<i>paramName</i>	A <code>System.String</code> that contains the name of the parameter that caused the exception.

## Description

This constructor initializes the `System.ArgumentException.Message` property of the new instance using *message*, and the `System.ArgumentException.ParamName` property using *paramName*. If *message* is null, the `System.ArgumentException.Message` property is initialized to the system-supplied message provided by the constructor that takes no arguments. The `System.ArgumentException.InnerException` property is initialized to null.



# ArgumentException.Message Property

```
[ILAsm]  
.property string Message { public hidebysig virtual specialname string  
get_Message() }  
  
[C#]  
public override string Message { get; }
```

## Summary

Gets the error message that explains the reason for the exception.

## Property Value

A System.String containing the error message with System.ArgumentException.ParamName appended, if it is not null.

## Description

This property is read-only.

# ArgumentException.ParamName Property

```
[ILAsm]  
.property string ParamName { public hidebysig virtual specialname string  
get_ParamName() }  
  
[C#]  
public virtual string ParamName { get; }
```

## Summary

Gets the name of the parameter that caused the current Exception.

## Property Value

A `System.String` that contains the name of the parameter that caused the current Exception, or `null` if no value was specified to the constructor for the current instance.

## Description

This property is read-only.

## Behaviors

The `System.ArgumentException.ParamName` property returns the same value as was passed into the constructor.

## How and When to Override

Override the `System.ArgumentException.ParamName` property to customize the content or format of the parameter name.

## Usage

Use this property to retrieve the name of the invalid parameter.