

# 1 System.ObjectDisposedException Class

2  
3

```
4 [ILASM]  
5 .class public serializable ObjectDisposedException extends  
6 System.InvalidOperationException  
  
7 [C#]  
8 public class ObjectDisposedException:  
9 InvalidOperationException
```

## 10 Assembly Info:

- 11 • Name: mscorlib
- 12 • Public Key: [00 00 00 00 00 00 00 00 04 00 00 00 00 00 00]
- 13 • Version: 1.0.x.x
- 14 • Attributes:
  - 15 ○ CLSCompliantAttribute(true)

## 16 Summary

17

18 Represents the error that occurs when an operation is performed on a  
19 disposed object.

## 20 Inherits From: System.InvalidOperationException

21

22 **Library:** BCL

23

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

26

## 27 Description

28 [Note: For additional information about disposing objects, see the  
29 **System.IDisposable** interface.]

## 30 Example

31

32 The following example demonstrates an error that causes the  
33 **System.ObjectDisposedException** exception to be thrown.

34

35 [C#]

```
36 using System;  
37 using System.IO;  
38  
39 public class ObjectDisposedExceptionTest {  
40     public static void Main() {
```

```
1         MemoryStream ms = new MemoryStream(16);
2         ms.Close();
3         try {
4             ms.ReadByte();
5         }
6         catch (ObjectDisposedException e) {
7             Console.WriteLine("Caught: {0}", e.Message);
8         }
9     }
10 }
```

11 The output is

12  
13 Caught: Cannot access a closed Stream.

14

15

# 1 ObjectDisposedException(System.String) 2 Constructor

```
3 [ILASM]  
4 public rtspecialname specialname instance void .ctor(string  
5 objectName)  
  
6 [C#]  
7 public ObjectDisposedException(string objectName)
```

## 8 Summary

9 Constructs and initializes a new instance of the  
10 **System.ObjectDisposedException** class.

## 11 Parameters

12  
13

Parameter	Description
<i>objectName</i>	A <b>System.String</b> containing the name of the disposed object.

14  
15

## 15 Description

16 This constructor initializes the  
17 **System.ObjectDisposedException.ObjectName** property of the  
18 new instance using *objectName*. The  
19 **System.ObjectDisposedException.Message** property is initialized  
20 to a system-supplied message that describes the error and includes  
21 *objectname*. This message takes into account the current system  
22 culture.

23  
24 The **System.ObjectDisposedException.InnerException** property of  
25 the new instance is initialized to **null**.

26  
27 [Note: If *objectName* is **null**, the  
28 **System.ObjectDisposedException.Message** property contains only  
29 an error message.]

## 30 Example

31

32 The following example displays the error message of a  
33 **System.ObjectDisposedException** instance created using this  
34 constructor.

```
35 [C#]  
36  
37 using System;  
38  
39 public class ExampleDisposableObject: IDisposable {
```

```

1      public static void Main() {
2
3          ExampleDisposableObject obj = new
4      ExampleDisposableObject();
5
6          obj.Close();
7
8          try {
9              Console.WriteLine(obj);
10         } catch (ObjectDisposedException e) {
11             Console.WriteLine("Caught: {0}", e.Message);
12         }
13     }
14
15
16     public ExampleDisposableObject() {
17         isDisposed = false;
18     }
19
20     ~ExampleDisposableObject() {
21         Dispose(true);
22     }
23
24     public void Close() {
25         Dispose(true);
26     }
27
28     public void Dispose() {
29         Dispose(true);
30     }
31
32     public void Dispose(bool disposing) {
33         isDisposed = true;
34     }
35
36     public override String ToString() {
37         if(isDisposed)
38             throw new
39         ObjectDisposedException("ExampleDisposableObject");
40         else
41             return "This is an instance of ExampleDisposableObject.";
42     }
43
44     private bool isDisposed;
45 }
46

```

47 The output is

```

48
49 Caught: Cannot access a disposed object named
50 "ExampleDisposableObject".

```

1  
2  
3  
4  
5

Object name: "ExampleDisposableObject".

# 1 ObjectDisposedException(System.String, 2 System.String) Constructor

```
3 [ILASM]  
4 public rtspecialname specialname instance void .ctor(string  
5 objectName, string message)  
  
6 [C#]  
7 public ObjectDisposedException(string objectName, string  
8 message)
```

## 9 Summary

10 Constructs and initializes a new instance of the  
11 **System.ObjectDisposedException** class.

## 12 Parameters

13  
14

Parameter	Description
<i>objectName</i>	A <b>System.String</b> containing the name of the disposed object.
<i>message</i>	A <b>System.String</b> 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.

15  
16

## Description

17 This constructor initializes the  
18 **System.ObjectDisposedException.Message** property of the new  
19 instance using *message*, and the  
20 **System.ObjectDisposedException.ObjectName** property using  
21 *objectName*. If *message* is **null**, the  
22 **System.ObjectDisposedException.Message** property is initialized  
23 to the system-supplied message provided by the constructor that  
24 takes no arguments.

25  
26  
27

The **System.ObjectDisposedException.InnerException** property of the new instance is initialized to **null**.

## 28 Example

29

30 The following example throws a **System.ObjectDisposedException**  
31 instance created using this constructor.

32  
33

```
[C#]
```

```
34 using System;
```

```

1
2 public class ExampleDisposableObject: IDisposable {
3     public static void Main() {
4
5         ExampleDisposableObject obj = new
6 ExampleDisposableObject();
7
8         obj.Close();
9
10        try {
11            Console.WriteLine(obj);
12        } catch (ObjectDisposedException e) {
13            Console.WriteLine("Caught: {0}", e.Message);
14        }
15    }
16
17
18    public ExampleDisposableObject() {
19        isDisposed = false;
20    }
21
22    ~ExampleDisposableObject() {
23        Dispose(true);
24    }
25
26    public void Close() {
27        Dispose(true);
28    }
29
30    public void Dispose() {
31        Dispose(true);
32    }
33
34    public void Dispose(bool disposing) {
35        isDisposed = true;
36    }
37
38    public override String ToString() {
39        if(isDisposed) {
40            string message = "Oh-oh! This object has been disposed!";
41            string objectName = "ExampleDisposableObject";
42            throw new ObjectDisposedException(objectName, message);
43        }
44        else
45            return "Hello, World!";
46    }
47
48    private bool isDisposed;
49 }

```

50 The output is

51  
52 Caught: Oh-oh! This object has been disposed!

1  
2  
3  
4  
5

Object name: "ExampleDisposableObject".

# 1 ObjectDisposedException.Message 2 Property

```
3 [ILASM]  
4 .property string Message { public hidebysig virtual  
5 specialname string get_Message() }  
6  
7 [C#]  
8 public override string Message { get; }
```

## 8 Summary

9 Gets the message that describes the error.

## 10 Property Value

11

12 A **System.String** that describes the error.

## 13 Description

14 If the **System.ObjectDisposedException.ObjectName** property is  
15 not **null**, the message includes the name of the object.

16

17 This property is read-only.

18

19 [*Note:* This property overrides **System.Exception.Message**.]

20

# 1 ObjectDisposedException.ObjectName 2 Property

```
3 [ILASM]  
4 .property string ObjectName { public hidebysig specialname  
5 instance string get_ObjectName() }  
  
6 [C#]  
7 public string ObjectName { get; }
```

## 8 Summary

9 Gets the name of the disposed object.

## 10 Property Value

11

12 A **System.String** containing the name of the disposed object.

## 13 Description

14 [Note: If this property is not **null** or **System.String.Empty**, the value  
15 of this property is included in the string returned by the  
16 **System.ObjectDisposedException.Message** property.]

17

18 This property is read-only.

19