

# System.Net.WebResponse Class

```
[ILAsm]
.class public abstract serializable WebResponse extends
System.MarshalByRefObject implements System.IDisposable

[C#]
public abstract class WebResponse: MarshalByRefObject, IDisposable
```

## Assembly Info:

- *Name:* System
- *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)

## Implements:

- **System.IDisposable**

## Summary

Represents a response received from a Uniform Resource Identifier (URI).

## Inherits From: System.MarshalByRefObject

**Library:** Networking

**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.Net.WebResponse` is the base class from which protocol-specific response classes, such as `System.Net.HttpWebResponse`, are derived.

Classes that derive from `System.Net.WebResponse` are required to override the following members in the `System.Net.WebResponse` class:

- `System.Net.WebResponse.ContentLength`: Gets or sets the content length of the data being received.
- `System.Net.WebResponse.ContentType`: Gets or sets the media type of the data being received.
- `System.Net.WebResponse.GetResponseStream`: Returns a `System.IO.Stream` that contains data from the current host.

- `System.Net.WebResponse.ResponseUri`: Gets a `System.Uri` containing the URI of the resource associated with the current instance.
- `System.Net.WebResponse.Headers`: Gets or sets the collection of header name/value pairs associated with the current instance.

[*Note*: Applications can participate in request/response transactions in a protocol-agnostic manner using instances of the `System.Net.WebResponse` class while protocol-specific classes derived from `System.Net.WebResponse` carry out the details of the request.

Applications do not create `System.Net.WebResponse` objects directly; they are created by calling `System.Net.WebRequest.GetResponse`.

]

## Example

The following example creates a `System.Net.WebResponse` instance from a `System.Net.WebRequest`.

[C#]

```
using System;
using System.Net;

public class WebResponseExample {

    public static void Main() {

        // Initialize the WebRequest.
        WebRequest myRequest =
            WebRequest.Create("http://www.contoso.com");

        // Return the response.
        WebResponse myResponse = myRequest.GetResponse();

        // Code to use the WebResponse goes here.

        // Close the response to free resources.
        myResponse.Close();
    }
}
```

# WebResponse() Constructor

```
[ILAsm]  
family rtspecialname specialname instance void .ctor()
```

```
[C#]  
protected WebResponse()
```

## Summary

Constructs a new instance of the `System.Net.WebResponse` class.

## Description

This constructor is called only by classes that derive from `System.Net.WebResponse`.

[*Note:* To obtain a `System.Net.WebResponse` instance, use the `System.Net.WebRequest.GetResponse` method; do not use this constructor.]

# WebResponse.Close() Method

```
[ILAsm]  
.method public hidebysig virtual void Close()  
  
[C#]  
public virtual void Close()
```

## Summary

Closes the response stream.

## Behaviors

This method is required to free the connection used by the `System.IO.Stream` returned by a call to `System.Net.WebResponse.GetResponseStream`.

## Default

The `System.Net.WebResponse` class is abstract and does not provide an implementation for this method. This method throws `System.NotSupportedException`.

## How and When to Override

This method is required to be overridden in derived classes to close the response stream and free resources allocated by the current instance.

## Usage

Use this method or `System.IO.Stream.Close` to clean up the resources used by a `System.Net.WebResponse`.

## Exceptions

Exception	Condition
<b>System.NotSupportedException</b>	This method is not implemented in the derived class.



# WebResponse.GetResponseStream() Method

```
[ILAsm]  
.method public hidebysig virtual class System.IO.Stream  
GetResponseStream()  
  
[C#]  
public virtual Stream GetResponseStream()
```

## Summary

Returns a `System.IO.Stream` that contains data from the current host.

## Return Value

A `System.IO.Stream` for reading data from the current host.

## Behaviors

This method is required to return a `System.IO.Stream` from the host specified by the `System.Net.WebResponse.ResponseUri` property of the current instance.

## Default

This method throws `System.NotSupportedException`.

## How and When to Override

This method is required to be overridden by classes that inherit from `System.Net.WebResponse`.

## Usage

Use this method to open a connection to the current host. To close the stream when it is no longer needed, in order to prevent the exhaustion of system resources, invoke either the `System.IO.Stream.Close` or `System.Net.WebResponse.Close` methods.

## Exceptions

Exception	Condition
<b>System.NotSupportedException</b>	This method is not implemented in the derived class.

# WebResponse.System.IDisposable.Dispose() Method

```
[ILAsm]  
.method private final hidebysig virtual void  
System.IDisposable.Dispose()
```

```
[C#]  
void IDisposable.Dispose()
```

## Summary

Implemented to support the `System.IDisposable` interface. [Note: For more information, see `System.IDisposable.Dispose.`]

# WebResponse.ContentLength Property

```
[ILAsm]  
.property int64 ContentLength { public hidebysig virtual specialname  
int64 get_ContentLength() public hidebysig virtual specialname void  
set_ContentLength(int64 value) }
```

```
[C#]  
public virtual long ContentLength { get; set; }
```

## Summary

Gets or sets the content length of the data received from the server.

## Property Value

A `System.Int64` containing the number of bytes returned from the resource.

## Behaviors

This property contains the number of bytes of data in the response to the request associated with the current instance. For request methods that contain header information, the `System.Net.WebResponse.ContentLength` does not include the length of the header information. The value of this property is set by the `Content-Length` header received in the response.

## Default

The `System.Net.WebResponse` class is abstract and does not provide an implementation for this property. This property throws `System.NotSupportedException`.

## How and When to Override

This property is required to be overridden by classes that inherit from `System.Net.WebResponse` to set the content length as appropriate for the derived class.

## Usage

Use this property to determine the number of bytes in the response.

## Exceptions

Exception	Condition
<b>System.NotSupportedException</b>	This property is not implemented in the derived class.

# WebResponse.ContentType Property

```
[ILAsm]
.property string ContentType { public hidebysig virtual specialname
string get_ContentType() public hidebysig virtual specialname void
set_ContentType(string value) }

[C#]
public virtual string ContentType { get; set; }
```

## Summary

Gets or sets the media type of the data received.

## Property Value

A `System.String` that contains the media type of the response data.

## Description

[*Note:* The media type is typically the MIME encoding of the content.]

## Behaviors

As described above.

## Default

The `System.Net.WebResponse` class is abstract and does not provide an implementation for this property. This property throws `System.NotSupportedException`.

## How and When to Override

This property is required to be overridden by classes that inherit from `System.Net.WebResponse`.

## Usage

Use this property to get the media type of the response.

## Exceptions

Exception	Condition
<b>System.NotSupportedException</b>	This property is not implemented in the derived class.

# WebResponse.Headers Property

```
[ILAsm]
.property class System.Net.WebHeaderCollection Headers { public
hidebysig virtual specialname class System.Net.WebHeaderCollection
get_Headers() }

[C#]
public virtual WebHeaderCollection Headers { get; }
```

## Summary

Gets the collection of header name/value pairs associated with the current instance.

## Property Value

A `System.Net.WebHeaderCollection` containing the header name/value pairs associated with the current instance.

## Description

This property contains a `System.Net.WebHeaderCollection` instance containing the header information returned in the response from the resource.

## Behaviors

This property is read-only.

## Default

The `System.Net.WebResponse` class is abstract and does not provide an implementation for this property. This property throws `System.NotSupportedException`.

## How and When to Override

This property is required to be overridden by classes that inherit from `System.Net.WebResponse`.

## Usage

Use this property to retrieve the header information received with the response.

## Exceptions

Exception	Condition
<b>System.NotSupportedException</b>	This property is not implemented in the derived class.

# WebResponse.ResponseUri Property

```
[ILAsm]
.property class System.Uri ResponseUri { public hidebysig virtual
specialname class System.Uri get_ResponseUri() }

[C#]
public virtual Uri ResponseUri { get; }
```

## Summary

Gets the URI of the resource associated with the current instance.

## Property Value

A `System.Uri` containing the URI of the resource associated with the current response.

## Behaviors

This property is read-only.

This property returns the URI that provided the response information in the current instance. [*Note:* If the method protocol used in the associated `System.Net.WebRequest` permits redirection, the value of this property might differ from the `System.Net.WebRequest.RequestUri` property of the request.]

## Default

The `System.Net.WebResponse` class is abstract and does not provide an implementation for this property. This property throws `System.NotSupportedException`.

## How and When to Override

This property is required to be overridden by classes that inherit from `System.Net.WebResponse`.

## Usage

Use this property to determine the URI that originated the response.

## Exceptions

Exception	Condition
<b>System.NotSupportedException</b>	This property is not implemented in the derived class.