

HttpClient.PostAsync Method

Reference



Definition

Namespace: [System.Net.Http](#)

Assembly: System.Net.Http.dll

Send a POST request to the specified Uri as an asynchronous operation.

In this article

- [Definition](#)
- [Overloads](#)
- [Remarks](#)
- [PostAsync\(String, HttpContent, CancellationTokens\)](#)
- [PostAsync\(Uri, HttpContent, CancellationTokens\)](#)
- [PostAsync\(String, HttpContent\)](#)
- [PostAsync\(Uri, HttpContent\)](#)

Overloads

PostAsync(String, HttpContent, CancellationTokens)	Send a POST request with a cancellation token as an asynchronous operation.
PostAsync(Uri, HttpContent, CancellationTokens)	Send a POST request with a cancellation token as an asynchronous operation.
PostAsync(String, HttpContent)	Send a POST request to the specified Uri as an asynchronous operation.
PostAsync(Uri, HttpContent)	Send a POST request to the specified Uri as an asynchronous operation.

Remarks

This operation will not block.

PostAsync(String, HttpContent, CancellationTokens)

Send a POST request with a cancellation token as an asynchronous operation.

C#	Copy
<pre>public System.Threading.Tasks.Task<System.Net.Http.HttpResponseMessage> PostAsync (string? requestUri, System.Net.Http.HttpContent? content, System.Threading.CancellationToken cancellationToken);</pre>	

Parameters

requestUri [String](#)

The Uri the request is sent to.

content [HttpContent](#)

The HTTP request content sent to the server.

cancellationToken [CancellationToken](#)

A cancellation token that can be used by other objects or threads to receive notice of cancellation.

Returns

[Task](#)<[HttpResponseMessage](#)>

The task object representing the asynchronous operation.

Exceptions

[InvalidOperationException](#)

The `requestUri` must be an absolute URI or [BaseAddress](#) must be set.

[HttpRequestException](#)

The request failed due to an underlying issue such as network connectivity, DNS failure, server certificate validation or timeout.

[TaskCanceledException](#)

.NET Core and .NET 5 and later only: The request failed due to timeout.

Remarks

This operation will not block. The returned [Task<TResult>](#) object will complete after the whole response (including content) is read.

⚠ Note

In case of timeout, different exceptions are thrown on different .NET implementations.


- [HttpRequestException](#) is thrown on all applicable .NET Framework versions.
- [TaskCanceledException](#) without any inner exception is thrown on all applicable .NET Core versions.
- [TaskCanceledException](#) that nests a [TimeoutException](#) is thrown on .NET 5 and later versions.

Applies to

► .NET 6 and other versions

PostAsync(Uri, HttpContent, CancellationTokens)

Send a POST request with a cancellation token as an asynchronous operation.

C#	 Copy
<pre>public System.Threading.Tasks.Task<System.Net.Http.HttpResponseMessage> PostAsync (Uri? requestUri, System.Net.Http.HttpContent? content, System.Threading.CancellationToken cancellationTokens);</pre>	

Parameters

requestUri [Uri](#)

The Uri the request is sent to.

content [HttpContent](#)

The HTTP request content sent to the server.

cancellationToken [CancellationToken](#)

A cancellation token that can be used by other objects or threads to receive notice of cancellation.

Returns

[Task](#)<[HttpResponseMessage](#)>

The task object representing the asynchronous operation.

Exceptions

[InvalidOperationException](#)

The `requestUri` must be an absolute URI or [BaseAddress](#) must be set.

[HttpRequestException](#)

The request failed due to an underlying issue such as network connectivity, DNS failure, server certificate validation or timeout.

[TaskCanceledException](#)

.NET Core and .NET 5 and later only: The request failed due to timeout.

Remarks

This operation will not block. The returned [Task<TResult>](#) object will complete after the whole response (including content) is read.

⚠ Note

In case of timeout, different exceptions are thrown on different .NET implementations.

- [HttpRequestException](#) is thrown on all applicable .NET Framework versions.
- [TaskCanceledException](#) without any inner exception is thrown on all applicable .NET Core versions.
- [TaskCanceledException](#) that nests a [TimeoutException](#) is thrown on .NET 5 and later versions.

Applies to

► .NET 6 and other versions

PostAsync(String, HttpContent)

Send a POST request to the specified Uri as an asynchronous operation.

C#

📄 Copy

```
public System.Threading.Tasks.Task<System.Net.Http.HttpResponseMessage> PostAsync (string? requestUri, System.Net.Http.HttpContent? content);
```

Parameters

requestUri [String](#)

The Uri the request is sent to.

content [HttpContent](#)

The HTTP request content sent to the server.

Returns

[Task<HttpResponseMessage>](#)

The task object representing the asynchronous operation.

Exceptions

[InvalidOperationException](#)

The `requestUri` must be an absolute URI or [BaseAddress](#) must be set.

[HttpRequestException](#)

The request failed due to an underlying issue such as network connectivity, DNS failure, server certificate validation or timeout.

[TaskCanceledException](#)

.NET Core and .NET 5 and later only: The request failed due to timeout.

Remarks

This operation will not block. The returned [Task<TResult>](#) object will complete after the whole response (including content) is read.

ⓘ Note

In case of timeout, different exceptions are thrown on different .NET implementations.

- [HttpRequestException](#) is thrown on all applicable .NET Framework versions.
- [TaskCanceledException](#) without any inner exception is thrown on all applicable .NET Core versions.
- [TaskCanceledException](#) that nests a [TimeoutException](#) is thrown on .NET 5 and later versions.

Applies to

► .NET 6 and other versions

PostAsync(Uri, HttpContent)

Send a POST request to the specified Uri as an asynchronous operation.

--	--

C#

Copy

```
public System.Threading.Tasks.Task<System.Net.Http.HttpResponseMessage> PostAsync (Uri? requestUri,
System.Net.Http.HttpContent? content);
```

Parameters

requestUri [Uri](#)

The Uri the request is sent to.

content [HttpContent](#)

The HTTP request content sent to the server.

Returns

[Task<HttpResponseMessage>](#)

The task object representing the asynchronous operation.

Exceptions

[InvalidOperationException](#)

The `requestUri` must be an absolute URI or [BaseAddress](#) must be set.

[HttpRequestException](#)

The request failed due to an underlying issue such as network connectivity, DNS failure, server certificate validation or timeout.

[TaskCanceledException](#)

.NET Core and .NET 5 and later only: The request failed due to timeout.

Remarks

This operation will not block. The returned [Task<TResult>](#) object will complete after the whole response (including content) is read.

ⓘ Note

In case of timeout, different exceptions are thrown on different .NET implementations.

- [HttpRequestException](#) is thrown on all applicable .NET Framework versions.
- [TaskCanceledException](#) without any inner exception is thrown on all applicable .NET Core versions.
- [TaskCanceledException](#) that nests a [TimeoutException](#) is thrown on .NET 5 and later versions.

Applies to

► .NET 6 and other versions

Recommended content

[HttpClient.GetAsync Method \(System.Net.Http\)](#)

Send a GET request to the specified Uri as an asynchronous operation.

HttpClient.DefaultRequestHeaders Property (Windows.Web.Http) - Windows UWP applications

Gets a collection of headers that should be sent with each request. For programming guidance for the HttpClient class, and code examples, see the HttpClient conceptual topic.

HttpClient Class (System.Net.Http)

Provides a class for sending HTTP requests and receiving HTTP responses from a resource identified by a URI.

HttpContent.ReadAsStringAsync Method (System.Net.Http)

Serialize the HTTP content to a string as an asynchronous operation.

StringContent Class (System.Net.Http)

Provides HTTP content based on a string.

FormUrlEncodedContent Class (System.Net.Http)

A container for name/value tuples encoded using application/x-www-form-urlencoded MIME type.

HttpResponseMessage Class (System.Net.Http)

Represents a HTTP response message including the status code and data.

AuthenticationHeaderValue Constructor (System.Net.Http.Headers)

Initializes a new instance of the AuthenticationHeaderValue class.

Show more ▾