

# Namespace OllamaSharp

## Classes

### [Chat](#)

A chat helper that handles the chat logic internally and automatically extends the message history.

A simple interactive chat can be implemented in just a handful of lines:

```
var ollama = new OllamaApiClient("http://localhost:11434", "llama3.2-vision:latest");
var chat = new Chat(ollama);
// ...
while (true)
{
    Console.Write("You: ");
    var message = Console.ReadLine()!;
    Console.Write("Ollama: ");
    await foreach (var answerToken in chat.SendAsync(message))
        Console.Write(answerToken);
    // ...
    Console.WriteLine();
}
// ...
// Output:
// You: Write a haiku about AI models
// Ollama: Code whispers secrets
// Intelligent designs unfold
// Minds beyond our own
```

### [ChatOptionsExtensions](#)

Extension methods to stream IAsyncEnumerable to its end and return one single result value

### [ConversationContext](#)

Represents a conversation context containing context data.

### [IAsyncEnumerableExtensions](#)

Extension methods to stream IAsyncEnumerable to its end and return one single result value

### [OllamaApiClient](#)

The default client to use the Ollama API conveniently.

<https://github.com/jmorganca/ollama/blob/main/docs/api.md> ↗

### [OllamaApiClient.Configuration](#)

The configuration for the Ollama API client.

## [OllamaApiClientExtensions](#)

Extension methods to simplify the usage of the [I OllamaApiClient](#).

## Interfaces

### [I OllamaApiClient](#)

Interface for the Ollama API client.

# Class Chat

Namespace: [OllamaSharp](#)

Assembly: OllamaSharp.dll

A chat helper that handles the chat logic internally and automatically extends the message history.

A simple interactive chat can be implemented in just a handful of lines:

```
var ollama = new OllamaApiClient("http://localhost:11434", "llama3.2-vision:latest");
var chat = new Chat(ollama);
// ...
while (true)
{
    Console.Write("You: ");
    var message = Console.ReadLine()!;
    Console.Write("Ollama: ");
    await foreach (var answerToken in chat.SendAsync(message))
        Console.Write(answerToken);
    // ...
    Console.WriteLine();
}
// ...
// Output:
// You: Write a haiku about AI models
// Ollama: Code whispers secrets
// Intelligent designs unfold
// Minds beyond our own

public class Chat
```

## Inheritance

[object](#) ← Chat

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Constructors

## Chat(IllamaApiClient, string)

Creates a new chat instance

```
public Chat(IllamaApiClient client, string systemPrompt = "")
```

### Parameters

**client** [IllumaApiClient](#)

The Ollama client to use for the chat

**systemPrompt** [string](#)

An optional system prompt to define the behavior of the chat assistant

### Examples

Setting up a chat with a system prompt:

```
var client = new OllamaApiClient("http://localhost:11434", "llama3.2-vision:latest");
var prompt = "You are a helpful assistant that will answer any question you are asked.";
var chat = new Chat(client, prompt);
```

### Exceptions

[ArgumentNullException](#)

If the client is null, an [ArgumentNullException](#) is thrown.

## Properties

### Client

Gets the Ollama API client

```
public IllamaApiClient Client { get; }
```

### Property Value

## Messages

Gets or sets the messages of the chat history

```
public List<Message> Messages { get; set; }
```

### Property Value

[List](#) <[Message](#)>

## Model

Gets or sets the AI model to chat with

```
public string Model { get; set; }
```

### Property Value

[string](#)

## Options

Gets or sets the RequestOptions to chat with

```
public RequestOptions? Options { get; set; }
```

### Property Value

[RequestOptions](#)

## Methods

## SendAsAsync(ChatRole, string, IEnumerable<IEnumerable<byte>>?, CancellationToken)

Sends a message in a given role to the currently selected model and streams its response

```
public IAsyncEnumerable<string> SendAsAsync(ChatRole role, string message,
IEnumerable<IEnumerable<byte>>? imagesAsBytes, CancellationToken cancellationToken
= default)
```

Parameters

**role** [ChatRole](#)

The role in which the message should be sent

**message** [string](#)

The message to send

**imagesAsBytes** [IEnumerable](#)<[IEnumerable](#)<[byte](#)>>

Images in byte representation to send to the model

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with

Returns

[IAsyncEnumerable](#)<[string](#)>

## SendAsAsync(ChatRole, string, IEnumerable<string>?, CancellationToken)

Sends a message in a given role to the currently selected model and streams its response

```
public IAsyncEnumerable<string> SendAsAsync(ChatRole role, string message,
IEnumerable<string>? imagesAsBase64, CancellationToken cancellationToken = default)
```

Parameters

### `role` [ChatRole](#)

The role in which the message should be sent

### `message` [string](#)

The message to send

### `imagesAsBase64` [IEnumerable](#)<[string](#)>

Base64 encoded images to send to the model

### `cancellationToken` [CancellationToken](#)

The token to cancel the operation with

Returns

### [IAsyncEnumerable](#)<[string](#)>

## `SendAsAsync(ChatRole, string, IReadOnlyCollection<Tool>?, IEnumerable<string>?, CancellationToken)`

Sends a message in a given role to the currently selected model and streams its response

```
public IAsyncEnumerable<string> SendAsAsync(ChatRole role, string message,
IReadOnlyCollection<Tool>? tools, IEnumerable<string>? imagesAsBase64 = null,
CancellationToken cancellationToken = default)
```

Parameters

### `role` [ChatRole](#)

The role in which the message should be sent

### `message` [string](#)

The message to send

### `tools` [IReadOnlyCollection](#)<[Tool](#)>

Tools that the model can make use of, see <https://ollama.com/blog/tool-support>. By using tools, response streaming is automatically turned off

`imagesAsBase64` `IEnumerable<string>`

Base64 encoded images to send to the model

`cancellationToken` `CancellationToken`

The token to cancel the operation with

Returns

`IAsyncEnumerable<string>`

## SendAsAsync(ChatRole, string, CancellationToken)

Sends a message in a given role to the currently selected model and streams its response

```
public IAsyncEnumerable<string> SendAsAsync(ChatRole role, string message, CancellationToken  
cancellationToken = default)
```

Parameters

`role` `ChatRole`

The role in which the message should be sent

`message` `string`

The message to send

`cancellationToken` `CancellationToken`

The token to cancel the operation with

Returns

`IAsyncEnumerable<string>`

## SendAsync(string, IEnumerable<IEnumerable<byte>>?, CancellationToken)

Sends a message to the currently selected model and streams its response

```
public IAsyncEnumerable<string> SendAsync(string message, IEnumerable<IEnumerable<byte>>? imagesAsBytes, CancellationToken cancellationToken = default)
```

## Parameters

**message** [string](#)

The message to send

**imagesAsBytes** [IEnumerable](#)<[IEnumerable](#)<[byte](#)>>

Images in byte representation to send to the model

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with

## Returns

[IAsyncEnumerable](#)<[string](#)>

An [IAsyncEnumerable](#)<T> that streams the response.

## Examples

Getting a response from the model with an image:

```
var client = new HttpClient();
var cat = await client.GetByteArrayAsync("https://cataas.com/cat");
var ollama = new OllamaApiClient("http://localhost:11434", "llama3.2-vision:latest");
var chat = new Chat(ollama);
var response = chat.SendAsync("What do you see?", [cat]);
await foreach (var answerToken in response) Console.WriteLine(answerToken);

// Output: The image shows a white kitten with black markings on its
//         head and tail, sitting next to an orange tabby cat. The kitten
//         is looking at the camera while the tabby cat appears to be
//         sleeping or resting with its eyes closed. The two cats are
//         lying in a blanket that has been rumpled up.
```

## SendAsync(string, IEnumerable<string>?, CancellationToken)

Sends a message to the currently selected model and streams its response

```
public IAsyncEnumerable<string> SendAsync(string message, IEnumerable<string>? imagesAsBase64, CancellationToken cancellationToken = default)
```

## Parameters

**message** [string](#)

The message to send

**imagesAsBase64** [IEnumerable](#)<[string](#)>

Base64 encoded images to send to the model

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with

## Returns

[IAsyncEnumerable](#)<[string](#)>

An [IAsyncEnumerable](#)<T> that streams the response.

## Examples

Getting a response from the model with an image:

```
var client = new HttpClient();
var cat = await client.GetByteArrayAsync("https://cataas.com/cat");
var base64Cat = Convert.ToBase64String(cat);
var ollama = new OllamaApiClient("http://localhost:11434", "llama3.2-vision:latest");
var chat = new Chat(ollama);
var response = chat.SendAsync("What do you see?", [base64Cat]);
await foreach (var answerToken in response) Console.WriteLine(answerToken);

// Output:
// The image shows a cat lying on the floor next to an iPad. The cat is looking
// at the screen, which displays a game with fish and other sea creatures. The
// cat's paw is touching the screen, as if it is playing the game. The background
// of the image is a wooden floor.
```

## SendAsync(string, IReadOnlyCollection<Tool>?, IEnumerable<string>?, CancellationToken)

Sends a message to the currently selected model and streams its response

```
public IAsyncEnumerable<string> SendAsync(string message, IReadOnlyCollection<Tool>? tools, IEnumerable<string>? imagesAsBase64 = null, CancellationToken cancellationToken = default)
```

### Parameters

**message** [string](#)

The message to send

**tools** [IReadOnlyCollection](#)<[Tool](#)>

Tools that the model can make use of, see <https://ollama.com/blog/tool-support>. By using tools, response streaming is automatically turned off

**imagesAsBase64** [IEnumerable](#)<[string](#)>

Base64 encoded images to send to the model

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with

### Returns

[IAsyncEnumerable](#)<[string](#)>

## SendAsync(string, CancellationToken)

Sends a message to the currently selected model and streams its response

```
public IAsyncEnumerable<string> SendAsync(string message, CancellationToken cancellationToken = default)
```

### Parameters

**message** [string](#)

The message to send

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with

Returns

[IAsyncEnumerable](#) <[string](#)>

An [IAsyncEnumerable<T>](#) that streams the response.

Examples

Getting a response from the model:

```
var response = await chat.SendAsync("Write a haiku about AI models");
await foreach (var answerToken in response)
    Console.WriteLine(answerToken);
```

# Class ChatOptionsExtensions

Namespace: [OllamaSharp](#)

Assembly: OllamaSharp.dll

Extension methods to stream IAsyncEnumerable to its end and return one single result value

```
public static class ChatOptionsExtensions
```

## Inheritance

[object](#) ← ChatOptionsExtensions

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Methods

### AddOllamaOption(ChatOptions, OllamaOption, object)

Adds Ollama specific options to the additional properties of ChatOptions. These can be interpreted and sent to the Ollama API by OllamaSharp.

```
public static ChatOptions AddOllamaOption(this ChatOptions chatOptions, OllamaOption option,  
object value)
```

## Parameters

**chatOptions** ChatOptions

The chat options to set Ollama options on

**option** [OllamaOption](#)

The Ollama option to set, like OllamaOption.NumCtx for the option 'num\_ctx'

**value** [object](#)

The value for the option

## Returns

### ChatOptions

The Microsoft.Extensions.AI.ChatOptions with the Ollama option set

# Class ConversationContext

Namespace: [OllamaSharp](#)

Assembly: OllamaSharp.dll

Represents a conversation context containing context data.

```
public record ConversationContext : IEquatable<ConversationContext>
```

Inheritance

[object](#) ← ConversationContext

Implements

[IEquatable](#)<[ConversationContext](#)>

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### ConversationContext(long[])

Represents a conversation context containing context data.

```
public ConversationContext(long[] Context)
```

Parameters

Context [long](#)[]

## Properties

### Context

```
public long[] Context { get; init; }
```

## Property Value

[long ↗ \[\]](#)

# Class IAsyncEnumerableExtensions

Namespace: [OllamaSharp](#)

Assembly: OllamaSharp.dll

Extension methods to stream IAsyncEnumerable to its end and return one single result value

```
public static class IAsyncEnumerableExtensions
```

## Inheritance

[object](#) ← IAsyncEnumerableExtensions

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Methods

**StreamToEndAsync(IAsyncEnumerable<StreamingChatCompletionUpdate?>, Action<StreamingChatCompletionUpdate?>?)**

Streams a given [IAsyncEnumerable<T>](#) of response chunks to its end and builds one single Microsoft.Extensions.AI.StreamingChatCompletionUpdate out of them.

```
public static Task<StreamingChatCompletionUpdate?> StreamToEndAsync(this  
IAsyncEnumerable<StreamingChatCompletionUpdate?> stream,  
Action<StreamingChatCompletionUpdate?>? itemCallback = null)
```

## Parameters

**stream** [IAsyncEnumerable](#)<StreamingChatCompletionUpdate>

The [IAsyncEnumerable<T>](#) to stream

**itemCallback** [Action](#)<StreamingChatCompletionUpdate>

An optional callback to additionally process every single item from the IAsyncEnumerable

Returns

[Task](#) <StreamingChatCompletionUpdate>

A single Microsoft.Extensions.AI.StreamingChatCompletionUpdate built up from every single IAsyncEnumerable item

## StreamToEndAsync(IAsyncEnumerable<ChatResponseStream?>, Action<ChatResponseStream?>?)

Streams a given IAsyncEnumerable of response chunks to its end and builds one single ChatDoneResponseStream out of them.

```
public static Task<ChatDoneResponseStream?> StreamToEndAsync(this  
IAsyncEnumerable<ChatResponseStream?> stream, Action<ChatResponseStream?>? itemCallback  
= null)
```

Parameters

**stream** [IAsyncEnumerable](#) <ChatResponseStream>

The IAsyncEnumerable to stream

**itemCallback** [Action](#) <ChatResponseStream>

An optional callback to additionally process every single item from the IAsyncEnumerable

Returns

[Task](#) <ChatDoneResponseStream>

A single ChatDoneResponseStream built up from every single IAsyncEnumerable item

## StreamToEndAsync(IAsyncEnumerable<GenerateResponseStream?>, Action<GenerateResponseStream?>?)

Streams a given IAsyncEnumerable of response chunks to its end and builds one single GenerateDoneResponseStream out of them.

```
public static Task<GenerateDoneResponseStream?> StreamToEndAsync(this  
IAsyncEnumerable<GenerateResponseStream?> stream, Action<GenerateResponseStream?>?  
itemCallback = null)
```

## Parameters

**stream** [IAsyncEnumerable](#)<[GenerateResponseStream](#)>

The IAsyncEnumerable to stream

**itemCallback** [Action](#)<[GenerateResponseStream](#)>

An optional callback to additionally process every single item from the IAsyncEnumerable

## Returns

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

A single GenerateDoneResponseStream built up from every single IAsyncEnumerable item

## StreamToEndAsync(IAsyncEnumerable<string>, Action<string>?)

Streams a given IAsyncEnumerable to its end and appends its items to a single response string

```
public static Task<string> StreamToEndAsync(this IAsyncEnumerable<string> stream,  
Action<string>? itemCallback = null)
```

## Parameters

**stream** [IAsyncEnumerable](#)<[string](#)>

The IAsyncEnumerable to stream

**itemCallback** [Action](#)<[string](#)>

An optional callback to additionally process every single item from the IAsyncEnumerable

## Returns

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

A single response stream appended from every IAsyncEnumerable item

# Interface IOllamaApiClient

Namespace: [OllamaSharp](#)

Assembly: OllamaSharp.dll

Interface for the Ollama API client.

```
public interface IOllamaApiClient
```

## Extension Methods

[OllamaApiClientExtensions.CopyModelAsync\(IOllamaApiClient, string, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.CreateModelAsync\(IOllamaApiClient, string, string, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.CreateModelAsync\(IOllamaApiClient, string, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.DeleteModelAsync\(IOllamaApiClient, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.EmbedAsync\(IOllamaApiClient, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.GenerateAsync\(IOllamaApiClient, string, ConversationContext?, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.PullModelAsync\(IOllamaApiClient, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.PushModelAsync\(IOllamaApiClient, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions>ShowModelAsync\(IOllamaApiClient, string, CancellationToken\)](#).

## Properties

### SelectedModel

Gets or sets the name of the model to run requests on.

```
string SelectedModel { get; set; }
```

### Property Value

[string](#) ↗

### Uri

Gets the endpoint URI used by the API client.

```
Uri Uri { get; }
```

## Property Value

[Uri](#)

## Methods

### ChatAsync(ChatRequest, CancellationToken)

Sends a request to the /api/chat endpoint and streams the response of the chat.

```
IAsyncEnumerable<ChatResponseStream?> ChatAsync(ChatRequest request, CancellationToken cancellationToken = default)
```

#### Parameters

request [ChatRequest](#)

The request to send to Ollama.

cancellationToken [CancellationToken](#)

The token to cancel the operation with.

#### Returns

[IAsyncEnumerable](#) <[ChatResponseStream](#)>

An asynchronous enumerable that yields [ChatResponseStream](#). Each item represents a message in the chat response stream. Returns null when the stream is completed.

#### Remarks

This is the method to call the Ollama endpoint /api/chat. You might not want to do this manually. To implement a fully interactive chat, you should make use of the Chat class with "new Chat(...)"

### CopyModelAsync(CopyModelRequest, CancellationToken)

Sends a request to the /api/copy endpoint to copy a model.

Task `CopyModelAsync(CopyModelRequest request, CancellationToken cancellationToken = default)`

## Parameters

`request` [CopyModelRequest](#)

The parameters required to copy a model.

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

## Returns

[Task](#)

## CreateModelAsync(CreateModelRequest, CancellationToken)

Sends a request to the /api/create endpoint to create a model.

IAsyncEnumerable<CreateModelResponse?> `CreateModelAsync(CreateModelRequest request, CancellationToken cancellationToken = default)`

## Parameters

`request` [CreateModelRequest](#)

The request object containing the model details.

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

## Returns

[IAsyncEnumerable](#)<[CreateModelResponse](#)>

An asynchronous enumerable of the model creation status.

## DeleteModelAsync(DeleteModelRequest, CancellationToken)

Sends a request to the /api/delete endpoint to delete a model.

```
Task DeleteModelAsync(DeleteModelRequest request, CancellationToken cancellationToken  
= default)
```

### Parameters

**request** [DeleteModelRequest](#)

The request containing the model to delete.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

[Task](#)

## EmbedAsync(EmbedRequest, CancellationToken)

Sends a request to the /api/embed endpoint to generate embeddings.

```
Task<EmbedResponse> EmbedAsync(EmbedRequest request, CancellationToken cancellationToken  
= default)
```

### Parameters

**request** [EmbedRequest](#)

The parameters to generate embeddings for.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

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

A task that represents the asynchronous operation. The task result contains the [EmbedResponse](#).

## GenerateAsync(GenerateRequest, CancellationToken)

Streams completion responses from the /api/generate endpoint on the Ollama API based on the provided request.

```
IAsyncEnumerable<GenerateResponseStream?> GenerateAsync(GenerateRequest request,  
CancellationToken cancellationToken = default)
```

### Parameters

**request** [GenerateRequest](#)

The request containing the parameters for the completion.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

[IAsyncEnumerable](#) <[GenerateResponseStream](#)>

An asynchronous enumerable of [GenerateResponseStream](#).

## GetVersionAsync(CancellationToken)

Gets the version of Ollama.

```
Task<Version> GetVersionAsync(CancellationToken cancellationToken = default)
```

### Parameters

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

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

A task that represents the asynchronous operation. The task result contains the [Version](#).

## IsRunningAsync(CancellationToken)

Sends a query to check whether the Ollama API is running or not.

```
Task<bool> IsRunningAsync(CancellationToken cancellationToken = default)
```

### Parameters

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

### Returns

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

A task that represents the asynchronous operation. The task result contains a boolean indicating whether the API is running.

## ListLocalModelsAsync(CancellationToken)

Sends a request to the /api/tags endpoint to get all models that are available locally.

```
Task<IEnumerable<Model>> ListLocalModelsAsync(CancellationToken cancellationToken = default)
```

### Parameters

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

### Returns

[Task](#) <[IEnumerable](#)<[Model](#)>>

A task that represents the asynchronous operation. The task result contains a collection of [Model](#).

## ListRunningModelsAsync(CancellationToken)

Sends a request to the /api/ps endpoint to get the running models.

```
Task<IEnumerable<RunningModel>> ListRunningModelsAsync(CancellationToken cancellationToken  
= default)
```

### Parameters

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

[Task](#) <IEnumerable <[RunningModel](#)>>

A task that represents the asynchronous operation. The task result contains a collection of [Running Model](#).

## PullModelAsync(PullModelRequest, CancellationToken)

Sends a request to the /api/pull endpoint to pull a new model.

```
IAsyncEnumerable<PullModelResponse?> PullModelAsync(PullModelRequest request,  
CancellationToken cancellationToken = default)
```

### Parameters

**request** [PullModelRequest](#)

The request specifying the model name and whether to use an insecure connection.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

Returns

#### [IAsyncEnumerable<PullModelResponse>](#)

An asynchronous enumerable of [PullModelResponse](#) objects representing the status of the model pull operation.

## PushModelAsync(PushModelRequest, CancellationToken)

Pushes a model to the Ollama API endpoint.

```
IAsyncEnumerable<PushModelResponse?> PushModelAsync(PushModelRequest request,  
CancellationToken cancellationToken = default)
```

Parameters

#### **request** [PushModelRequest](#)

The request containing the model information to push.

#### **cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

Returns

#### [IAsyncEnumerable<PushModelResponse>](#)

An asynchronous enumerable of push status updates. Use the enumerator to retrieve the push status updates.

## ShowModelAsync(ShowModelRequest, CancellationToken)

Sends a request to the /api/show endpoint to show the information of a model.

```
Task<ShowModelResponse> ShowModelAsync(ShowModelRequest request, CancellationToken  
cancellationToken = default)
```

Parameters

`request` [ShowModelRequest](#)

The request containing the name of the model to get the information for.

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

Returns

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

A task that represents the asynchronous operation. The task result contains the [ShowModelResponse](#).

# Class OllamaApiClient

Namespace: [OllamaSharp](#)

Assembly: OllamaSharp.dll

The default client to use the Ollama API conveniently.

<https://github.com/jmorganca/ollama/blob/main/docs/api.md>

```
public class OllamaApiClient : IOllamaApiClient, IChatClient, IEmbeddingGenerator<string,  
Embedding<float>>, IDisposable
```

## Inheritance

[object](#) ← OllamaApiClient

## Implements

[IOllamaApiClient](#), IChatClient, IEmbeddingGenerator<[string](#)>, Embedding<[float](#)>>, [IDisposable](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Extension Methods

[OllamaApiClientExtensions.CopyModelAsync\(IOllamaApiClient, string, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.CreateModelAsync\(IOllamaApiClient, string, string, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.CreateModelAsync\(IOllamaApiClient, string, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.DeleteModelAsync\(IOllamaApiClient, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.EmbedAsync\(IOllamaApiClient, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.GenerateAsync\(IOllamaApiClient, string, ConversationContext?, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.PullModelAsync\(IOllamaApiClient, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions.PushModelAsync\(IOllamaApiClient, string, CancellationToken\)](#) ,  
[OllamaApiClientExtensions>ShowModelAsync\(IOllamaApiClient, string, CancellationToken\)](#).

## Constructors

### OllamaApiClient(Configuration)

Creates a new instance of the Ollama API client.

```
public OllamaApiClient(OllamaApiClient.Configuration config)
```

## Parameters

**config** [OllamaApiClient.Configuration](#)

The configuration for the Ollama API client.

## OllamaApiClient(HttpClient, string)

Creates a new instance of the Ollama API client.

```
public OllamaApiClient(HttpClient client, string defaultModel = "")
```

## Parameters

**client** [HttpClient](#)

The HTTP client to access the Ollama API with.

**defaultModel** [string](#)

The default model that should be used with Ollama.

## Exceptions

[ArgumentNullException](#)

## OllamaApiClient(string, string)

Creates a new instance of the Ollama API client.

```
public OllamaApiClient(string uriString, string defaultModel = "")
```

## Parameters

**uriString** [string](#)

The URI of the Ollama API endpoint.

**defaultModel** [string](#)

The default model that should be used with Ollama.

## OllamaApiClient(Uri, string)

Creates a new instance of the Ollama API client.

```
public OllamaApiClient(Uri uri, string defaultModel = "")
```

### Parameters

**uri** [Uri](#)

The URI of the Ollama API endpoint.

**defaultModel** [string](#)

The default model that should be used with Ollama.

# Properties

## Config

Gets the current configuration of the API client.

```
public OllamaApiClient.Configuration Config { get; }
```

### Property Value

[OllamaApiClient.Configuration](#)

## DefaultRequestHeaders

Gets the default request headers that are sent to the Ollama API.

```
public Dictionary<string, string> DefaultRequestHeaders { get; }
```

## Property Value

[Dictionary](#)<[string](#), [string](#)>

## IncomingJsonSerializerOptions

Gets the serializer options used for deserializing HTTP responses.

```
public JsonSerializerOptions IncomingJsonSerializerOptions { get; }
```

## Property Value

[JsonSerializerOptions](#)

## OutgoingJsonSerializerOptions

Gets the serializer options for outgoing web requests like Post or Delete.

```
public JsonSerializerOptions OutgoingJsonSerializerOptions { get; }
```

## Property Value

[JsonSerializerOptions](#)

## SelectedModel

Gets or sets the name of the model to run requests on.

```
public string SelectedModel { get; set; }
```

## Property Value

[string](#)

# Uri

Gets the endpoint URI used by the API client.

```
public Uri Uri { get; }
```

Property Value

[Uri](#)

## Methods

### ChatAsync(ChatRequest, CancellationToken)

Sends a request to the /api/chat endpoint and streams the response of the chat.

```
public IAsyncEnumerable<ChatResponseStream?> ChatAsync(ChatRequest request,  
CancellationToken cancellationToken = default)
```

Parameters

**request** [ChatRequest](#)

The request to send to Ollama.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

Returns

[IAsyncEnumerable](#) <[ChatResponseStream](#)>

An asynchronous enumerable that yields [ChatResponseStream](#). Each item represents a message in the chat response stream. Returns null when the stream is completed.

Remarks

This is the method to call the Ollama endpoint /api/chat. You might not want to do this manually. To implement a fully interactive chat, you should make use of the Chat class with "new Chat(...)"

## CopyModelAsync(CopyModelRequest, CancellationToken)

Sends a request to the /api/copy endpoint to copy a model.

```
public Task CopyModelAsync(CopyModelRequest request, CancellationToken cancellationToken = default)
```

### Parameters

**request** [CopyModelRequest](#)

The parameters required to copy a model.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

[Task](#)

## CreateModelAsync(CreateModelRequest, CancellationToken)

Sends a request to the /api/create endpoint to create a model.

```
public IAsyncEnumerable<CreateModelResponse?> CreateModelAsync(CreateModelRequest request, CancellationToken cancellationToken = default)
```

### Parameters

**request** [CreateModelRequest](#)

The request object containing the model details.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

[IAsyncEnumerable](#) <[CreateModelResponse](#)>

An asynchronous enumerable of the model creation status.

## DeleteModelAsync(DeleteModelRequest, CancellationToken)

Sends a request to the /api/delete endpoint to delete a model.

```
public Task DeleteModelAsync(DeleteModelRequest request, CancellationToken cancellationToken  
= default)
```

### Parameters

**request** [DeleteModelRequest](#)

The request containing the model to delete.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

[Task](#)

## EmbedAsync(EmbedRequest, CancellationToken)

Sends a request to the /api/embed endpoint to generate embeddings.

```
public Task<EmbedResponse> EmbedAsync(EmbedRequest request, CancellationToken  
cancellationToken = default)
```

### Parameters

**request** [EmbedRequest](#)

The parameters to generate embeddings for.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

## Returns

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

A task that represents the asynchronous operation. The task result contains the [EmbedResponse](#).

## GenerateAsync(GenerateRequest, CancellationToken)

Streams completion responses from the /api/generate endpoint on the Ollama API based on the provided request.

```
public IAsyncEnumerable<GenerateResponseStream?> GenerateAsync(GenerateRequest request,  
CancellationToken cancellationToken = default)
```

## Parameters

**request** [GenerateRequest](#)

The request containing the parameters for the completion.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

## Returns

[IAsyncEnumerable](#) <[GenerateResponseStream](#)>

An asynchronous enumerable of [GenerateResponseStream](#).

## GetVersionAsync(CancellationToken)

Gets the version of Ollama.

```
public Task<Version> GetVersionAsync(CancellationToken cancellationToken = default)
```

## Parameters

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

Returns

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

A task that represents the asynchronous operation. The task result contains the [Version](#).

## IsRunningAsync(CancellationToken)

Sends a query to check whether the Ollama API is running or not.

```
public Task<bool> IsRunningAsync(CancellationToken cancellationToken = default)
```

Parameters

cancellationToken [CancellationToken](#)

The token to cancel the operation with.

Returns

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

A task that represents the asynchronous operation. The task result contains a boolean indicating whether the API is running.

## ListLocalModelsAsync(CancellationToken)

Sends a request to the /api/tags endpoint to get all models that are available locally.

```
public Task<IEnumerable<Model>> ListLocalModelsAsync(CancellationToken cancellationToken = default)
```

Parameters

cancellationToken [CancellationToken](#)

The token to cancel the operation with.

Returns

[Task](#) <IEnumerable<[Model](#)>>

A task that represents the asynchronous operation. The task result contains a collection of [Model](#).

## ListRunningModelsAsync(CancellationToken)

Sends a request to the /api/ps endpoint to get the running models.

```
public Task<IEnumerable<RunningModel>> ListRunningModelsAsync(CancellationToken  
cancellationToken = default)
```

Parameters

[cancellationToken](#) [CancellationToken](#)

The token to cancel the operation with.

Returns

[Task](#) <IEnumerable<[RunningModel](#)>>

A task that represents the asynchronous operation. The task result contains a collection of [Running Model](#).

## PullModelAsync(PullModelRequest, CancellationToken)

Sends a request to the /api/pull endpoint to pull a new model.

```
public IAsyncEnumerable<PullModelResponse?> PullModelAsync(PullModelRequest request,  
CancellationToken cancellationToken = default)
```

Parameters

[request](#) [PullModelRequest](#)

The request specifying the model name and whether to use an insecure connection.

[cancellationToken](#) [CancellationToken](#)

The token to cancel the operation with.

Returns

[IAsyncEnumerable](#)<[PullModelResponse](#)>

An asynchronous enumerable of [PullModelResponse](#) objects representing the status of the model pull operation.

## PushModelAsync(PushModelRequest, CancellationToken)

Pushes a model to the Ollama API endpoint.

```
public IAsyncEnumerable<PushModelResponse?> PushModelAsync(PushModelRequest request,  
CancellationToken cancellationToken = default)
```

Parameters

**request** [PushModelRequest](#)

The request containing the model information to push.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

Returns

[IAsyncEnumerable](#)<[PushModelResponse](#)>

An asynchronous enumerable of push status updates. Use the enumerator to retrieve the push status updates.

## SendToOllamaAsync(HttpRequestMessage, OllamaRequest?, HttpCompletionOption, CancellationToken)

Sends an HTTP request message to the Ollama API.

```
protected virtual Task<HttpResponseMessage> SendToOllamaAsync(HttpRequestMessage  
requestMessage, OllamaRequest? ollamaRequest, HttpCompletionOption completionOption,
```

```
CancellationToken cancellationToken)
```

## Parameters

**requestMessage** [HttpRequestMessage](#)

The HTTP request message to send.

**ollamaRequest** [OllamaRequest](#)

The request containing custom HTTP request headers.

**completionOption** [HttpCompletionOption](#)

When the operation should complete (as soon as a response is available or after reading the whole response content).

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

## Returns

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

## ShowModelAsync(ShowModelRequest, CancellationToken)

Sends a request to the /api/show endpoint to show the information of a model.

```
public Task<ShowModelResponse> ShowModelAsync(ShowModelRequest request, CancellationToken  
cancellationToken = default)
```

## Parameters

**request** [ShowModelRequest](#)

The request containing the name of the model to get the information for.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

## Returns

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

A task that represents the asynchronous operation. The task result contains the [ShowModelResponse](#).

# Class OllamaApiClient.Configuration

Namespace: [OllamaSharp](#)

Assembly: OllamaSharp.dll

The configuration for the Ollama API client.

```
public class OllamaApiClient.Configuration
```

## Inheritance

[object](#) ← OllamaApiClient.Configuration

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Model

Gets or sets the model that should be used.

```
public string Model { get; set; }
```

### Property Value

[string](#)

## Uri

Gets or sets the URI of the Ollama API endpoint.

```
public Uri Uri { get; set; }
```

### Property Value



# Class OllamaApiClientExtensions

Namespace: [OllamaSharp](#)

Assembly: OllamaSharp.dll

Extension methods to simplify the usage of the [IOllamaApiClient](#).

```
public static class OllamaApiClientExtensions
```

## Inheritance

[object](#) ← OllamaApiClientExtensions

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Methods

### CopyModelAsync([IOllamaApiClient](#), string, string, CancellationToken)

Sends a request to the /api/copy endpoint to copy a model.

```
public static Task CopyModelAsync(this IOllamaApiClient client, string source, string destination, CancellationToken cancellationToken = default)
```

#### Parameters

**client** [IOllamaApiClient](#)

The client used to execute the command.

**source** [string](#)

The name of the existing model to copy.

**destination** [string](#)

The name the copied model should get.

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

Returns

[Task](#)

A task that represents the asynchronous operation.

## CreateModelAsync(IllamaApiClient, string, string, string, CancellationToken)

Sends a request to the /api/create endpoint to create a model.

```
public static IAsyncEnumerable<CreateModelResponse?> CreateModelAsync(this IllamaApiClient  
client, string name, string modelFileContent, string path, CancellationToken  
cancellationToken = default)
```

Parameters

`client` [IllamaApiClient](#)

The client used to execute the command.

`name` [string](#)

The name for the new model.

`modelFileContent` [string](#)

The file content for the model file the new model should be built with. See <https://github.com/jmorganca/ollama/blob/main/docs/modelfile.md>.

`path` [string](#)

The name path to the model file.

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

## Returns

[IAsyncEnumerable](#)<[CreateModelResponse](#)>

An async enumerable that can be used to iterate over the streamed responses. See [CreateModel Response](#).

## CreateModelAsync(IllamaApiClient, string, string, CancellationToken)

Sends a request to the /api/create endpoint to create a model.

```
public static IAsyncEnumerable<CreateModelResponse?> CreateModelAsync(this IllamaApiClient client, string name, string modelFileContent, CancellationToken cancellationToken = default)
```

### Parameters

**client** [IllamaApiClient](#)

The client used to execute the command.

**name** [string](#)

The name for the new model.

**modelFileContent** [string](#)

The file content for the model file the new model should be built with. See <https://github.com/jmorganca/ollama/blob/main/docs/modelfile.md>.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

## Returns

[IAsyncEnumerable](#)<[CreateModelResponse](#)>

An async enumerable that can be used to iterate over the streamed responses. See [CreateModel Response](#).

## DeleteModelAsync(IllamaApiClient, string, CancellationToken)

Sends a request to the /api/delete endpoint to delete a model.

```
public static Task DeleteModelAsync(this IllamaApiClient client, string model,  
CancellationToken cancellationToken = default)
```

### Parameters

**client** [IllamaApiClient](#)

The client used to execute the command.

**model** [string](#)

The name of the model to delete.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

### Returns

[Task](#)

A task that represents the asynchronous operation.

## EmbedAsync(IllamaApiClient, string, CancellationToken)

Sends a request to the /api/embed endpoint to generate embeddings for the currently selected model.

```
public static Task<EmbedResponse> EmbedAsync(this IllamaApiClient client, string input,  
CancellationToken cancellationToken = default)
```

### Parameters

**client** [IllamaApiClient](#)

The client used to execute the command.

**input** [string](#)

The input text to generate embeddings for.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

Returns

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

A [EmbedResponse](#) containing the embeddings.

## GenerateAsync(IllamaApiClient, string, ConversationContext?, CancellationToken)

Sends a request to the /api/generate endpoint to get a completion and streams the returned chunks to a given streamer that can be used to update the user interface in real-time.

```
public static IAsyncEnumerable<GenerateResponseStream?> GenerateAsync(this IllamaApiClient  
client, string prompt, ConversationContext? context = null, CancellationToken  
cancellationToken = default)
```

Parameters

**client** [IllamaApiClient](#)

The client used to execute the command.

**prompt** [string](#)

The prompt to generate a completion for.

**context** [ConversationContext](#)

The context that keeps the conversation for a chat-like history. Should reuse the result from earlier calls if these calls belong together. Can be null initially.

**cancellationToken** [CancellationToken](#)

The token to cancel the operation with.

Returns

## [IAsyncEnumerable](#) <[GenerateResponseStream](#)>

An async enumerable that can be used to iterate over the streamed responses. See [GenerateResponse Stream](#).

## PullModelAsync(IllamaApiClient, string, CancellationToken)

Sends a request to the /api/pull endpoint to pull a new model.

```
public static IAsyncEnumerable<PullModelResponse?> PullModelAsync(this IllamaApiClient  
client, string model, CancellationToken cancellationToken = default)
```

### Parameters

#### [client](#) [IllamaApiClient](#)

The client used to execute the command.

#### [model](#) [string](#)

The name of the model to pull.

#### [cancellationToken](#) [CancellationToken](#)

The token to cancel the operation with.

### Returns

## [IAsyncEnumerable](#) <[PullModelResponse](#)>

An async enumerable that can be used to iterate over the streamed responses. See [PullModel Response](#).

## PushModelAsync(IllamaApiClient, string, CancellationToken)

Sends a request to the /api/push endpoint to push a new model.

```
public static IAsyncEnumerable<PushModelResponse?> PushModelAsync(this IllamaApiClient  
client, string name, CancellationToken cancellationToken = default)
```

## Parameters

`client` [IOllamaApiClient](#)

The client used to execute the command.

`name` [string](#)

The name of the model to push.

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

## Returns

[IAsyncEnumerable](#) <[PushModelResponse](#)>

An async enumerable that can be used to iterate over the streamed responses. See [PullModelResponse](#).

## ShowModelAsync(IOllamaApiClient, string, CancellationToken)

Sends a request to the /api/show endpoint to show the information of a model.

```
public static Task<ShowModelResponse> ShowModelAsync(this IOllamaApiClient client, string model, CancellationToken cancellationToken = default)
```

## Parameters

`client` [IOllamaApiClient](#)

The client used to execute the command.

`model` [string](#)

The name of the model to get the information for.

`cancellationToken` [CancellationToken](#)

The token to cancel the operation with.

## Returns

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

A task that represents the asynchronous operation. The task result contains the [ShowModelResponse](#) with the model information.

# Namespace OllamaSharp.Models

## Classes

### [CopyModelRequest](#)

Copy a model. Creates a model with another name from an existing model. [Ollama API docs](#)

### [CreateModelRequest](#)

Create a model from a Modelfile. It is recommended to set [ModelFileContent](#) to the content of the Modelfile rather than just set path. This is a requirement for remote create. Remote model creation must also create any file blobs, fields such as FROM and ADAPTER, explicitly with the server using Create a Blob and the value to the path indicated in the response.

[Ollama API docs](#)

### [CreateModelResponse](#)

Represents the response from the /api/create endpoint

### [DeleteModelRequest](#)

Delete a model and its data.

[Ollama API docs](#)

### [Details](#)

Represents additional details about a model.

### [EmbedRequest](#)

Generate embeddings from a model.

[Ollama API docs](#)

### [EmbedResponse](#)

The response from the /api/embed endpoint

### [GenerateDoneResponseStream](#)

Represents the final response from the /api/generate endpoint

### [GenerateRequest](#)

Generate a response for a given prompt with a provided model. This is a streaming endpoint, so there will be a series of responses. The final response object will include statistics and additional data from the request.

[Ollama API docs](#)

### [GenerateResponseStream](#)

The response from the /api/generate endpoint when streaming is enabled

## [ListModelsResponse](#)

List models that are available locally.

[Ollama API docs](#)

## [ListRunningModelsResponse](#)

List models that are currently loaded into memory.

[Ollama API docs](#)

## [Model](#)

Represents a model with its associated metadata.

## [ModelInfo](#)

Represents additional model information.

## [OllamaOption](#)

Collection of options available to Ollama

## [OllamaRequest](#)

Represents the base class for requests to the Ollama API.

## [ProjectorInfo](#)

Represents projector-specific information.

## [PullModelRequest](#)

Download a model from the ollama library. Cancelled pulls are resumed from where they left off, and multiple calls will share the same download progress.

[Ollama API docs](#)

## [PullModelResponse](#)

Represents the streamed response from the /api/pull endpoint.

## [PushModelRequest](#)

Upload a model to a model library. Requires registering for ollama.ai and adding a public key first.

[Ollama API docs](#)

## [PushModelResponse](#)

Represents the response from the /api/push endpoint.

## [RequestOptions](#)

The configuration information used for a chat completions request.

## [RunningModel](#)

Represents a running model.

## [ShowModelRequest](#)

Show information about a model including details, modelfile, template, parameters, license, system prompt.

[Ollama API docs](#) ↗

## [ShowModelResponse](#)

Represents the response containing detailed model information.

# Class CopyModelRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Copy a model. Creates a model with another name from an existing model. [Ollama API docs](#)

```
public class CopyModelRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← CopyModelRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Destination

The destination model name

```
[JsonPropertyName("destination")]
public string Destination { get; set; }
```

### Property Value

[string](#)

### Source

The source model name

```
[JsonPropertyName("source")]
public string Source { get; set; }
```

## Property Value

[string](#) ↗

# Class CreateModelRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Create a model from a Modelfile. It is recommended to set [ModelFileContent](#) to the content of the Modelfile rather than just set path. This is a requirement for remote create. Remote model creation must also create any file blobs, fields such as FROM and ADAPTER, explicitly with the server using Create a Blob and the value to the path indicated in the response.

[Ollama API docs](#)

```
[JsonUnmappedMemberHandling(JsonUnmappedMemberHandling.Skip)]  
public class CreateModelRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← CreateModelRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Model

Name of the model to create

```
[JsonPropertyName("model")]  
public string? Model { get; set; }
```

### Property Value

[string](#)

### ModelFileContent

Contents of the Modelfile See <https://github.com/jmorganca/ollama/blob/main/docs/modelfile.md>

```
[JsonPropertyName("modelfile")]
public string ModelFileContent { get; set; }
```

## Property Value

[string](#)

## Path

Path to the Modelfile (optional)

```
[JsonPropertyName("path")]
public string? Path { get; set; }
```

## Property Value

[string](#)

## Quantize

Set the quantization level for quantize model when importing (e.g. q4\_0, optional)

```
[JsonPropertyName("quantize")]
public string? Quantize { get; set; }
```

## Property Value

[string](#)

## Stream

If false the response will be returned as a single response object, rather than a stream of objects (optional)

```
[JsonPropertyName("stream")]
public bool Stream { get; set; }
```

Property Value

bool ↗

# Class CreateModelResponse

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents the response from the /api/create endpoint

```
public class CreateModelResponse
```

## Inheritance

[object](#) ← CreateModelResponse

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Status

Represents the status of a model creation.

```
[JsonPropertyName("status")]
public string Status { get; set; }
```

## Property Value

[string](#)

# Class DeleteModelRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Delete a model and its data.

[Ollama API docs](#)

```
[JsonUnmappedMemberHandling(JsonUnmappedMemberHandling.Skip)]
```

```
public class DeleteModelRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← DeleteModelRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Model

The name of the model to delete

```
[JsonPropertyName("model")]
public string? Model { get; set; }
```

### Property Value

[string](#)

# Class Details

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents additional details about a model.

```
public class Details
```

## Inheritance

[object](#) ← Details

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Families

Gets or sets the families of the model.

```
[JsonPropertyName("families")]
public string[]? Families { get; set; }
```

## Property Value

[string](#)[]

## Family

Gets or sets the family of the model.

```
[JsonPropertyName("family")]
public string Family { get; set; }
```

## Property Value

[string](#) ↗

## Format

Gets or sets the format of the model file.

```
[JsonPropertyName("format")]
public string Format { get; set; }
```

## Property Value

[string](#) ↗

## ParameterSize

Gets or sets the number of parameters in the model.

```
[JsonPropertyName("parameter_size")]
public string ParameterSize { get; set; }
```

## Property Value

[string](#) ↗

## ParentModel

Gets or sets the name of the parent model on which the model is based.

```
[JsonPropertyName("parent_model")]
public string? ParentModel { get; set; }
```

## Property Value

[string](#) ↗

## QuantizationLevel

Gets or sets the quantization level of the model.

```
[JsonPropertyName("quantization_level")]
public string QuantizationLevel { get; set; }
```

Property Value

[string](#)

# Class EmbedRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Generate embeddings from a model.

[Ollama API docs](#)

```
public class EmbedRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← EmbedRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Input

The text to generate embeddings for

```
[JsonPropertyName("input")]
public List<string> Input { get; set; }
```

### Property Value

[List](#) <[string](#)>

### KeepAlive

Gets or sets the KeepAlive property, which decides how long a given model should stay loaded.

```
[JsonPropertyName("keep_alive")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public long? KeepAlive { get; set; }
```

## Property Value

[long](#)?

## Model

The name of the model to generate embeddings from

```
[JsonPropertyName("model")]
public string Model { get; set; }
```

## Property Value

[string](#)

## Options

Additional model parameters listed in the documentation for the Modelfile such as temperature.

```
[JsonPropertyName("options")]
public RequestOptions? Options { get; set; }
```

## Property Value

[RequestOptions](#)

## Truncate

Truncates the end of each input to fit within context length. Returns error if false and context length is exceeded. Defaults to true

```
[JsonPropertyName("truncate")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? Truncate { get; set; }
```

Property Value

bool?

# Class EmbedResponse

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

The response from the /api/embed endpoint

```
public class EmbedResponse
```

## Inheritance

[object](#) ← EmbedResponse

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Embeddings

An array of embeddings for the input text

```
[JsonPropertyName("embeddings")]
public List<float> Embeddings { get; set; }
```

### Property Value

[List](#)<[float](#)[]>

### LoadDuration

The time spent in nanoseconds loading the model

```
[JsonPropertyName("load_duration")]
public long? LoadDuration { get; set; }
```

Property Value

[long](#)?

## PromptEvalCount

The number of tokens in the input text

```
[JsonPropertyName("prompt_eval_count")]
public int? PromptEvalCount { get; set; }
```

Property Value

[int](#)?

## TotalDuration

The time spent generating the response

```
[JsonPropertyName("total_duration")]
public long? TotalDuration { get; set; }
```

Property Value

[long](#)?

# Class GenerateDoneResponseStream

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents the final response from the /api/generate endpoint

```
public class GenerateDoneResponseStream : GenerateResponseStream
```

## Inheritance

[object](#) ← [GenerateResponseStream](#) ← GenerateDoneResponseStream

## Inherited Members

[GenerateResponseStream.Model](#) , [GenerateResponseStream.CreatedAtString](#) ,  
[GenerateResponseStream.CreatedAt](#) , [GenerateResponseStream.Response](#) ,  
[GenerateResponseStream.Done](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Context

An encoding of the conversation used in this response, this can be sent in the next request to keep a conversational memory

```
[JsonPropertyName("context")]
public long[] Context { get; set; }
```

### Property Value

[long](#)[]

### EvalCount

The number of tokens in the response

```
[JsonPropertyName("eval_count")]
public int EvalCount { get; set; }
```

Property Value

[int](#)

## EvalDuration

The time in nanoseconds spent generating the response

```
[JsonPropertyName("eval_duration")]
public long EvalDuration { get; set; }
```

Property Value

[long](#)

## LoadDuration

The time spent in nanoseconds loading the model

```
[JsonPropertyName("load_duration")]
public long LoadDuration { get; set; }
```

Property Value

[long](#)

## PromptEvalCount

The number of tokens in the prompt

```
[JsonPropertyName("prompt_eval_count")]
public int PromptEvalCount { get; set; }
```

Property Value

[int](#) ↗

## PromptEvalDuration

The time spent in nanoseconds evaluating the prompt

```
[JsonPropertyName("prompt_eval_duration")]
public long PromptEvalDuration { get; set; }
```

Property Value

[long](#) ↗

## TotalDuration

The time spent generating the response

```
[JsonPropertyName("total_duration")]
public long TotalDuration { get; set; }
```

Property Value

[long](#) ↗

# Class GenerateRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Generate a response for a given prompt with a provided model. This is a streaming endpoint, so there will be a series of responses. The final response object will include statistics and additional data from the request.

[Ollama API docs](#)

```
public class GenerateRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← GenerateRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Context

The context parameter returned from a previous request to /generate, this can be used to keep a short conversational memory

```
[JsonPropertyName("context")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public long[]? Context { get; set; }
```

## Property Value

[long](#)[]

## Format

The format to return a response in. Currently only accepts "json" or null.

```
[JsonPropertyName("format")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string? Format { get; set; }
```

## Property Value

[string](#)

## Images

Base64-encoded images (for multimodal models such as llava)

```
[JsonPropertyName("images")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string[]? Images { get; set; }
```

## Property Value

[string](#)[]

## KeepAlive

Gets or sets the KeepAlive property, which decides how long a given model should stay loaded.

```
[JsonPropertyName("keep_alive")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string? KeepAlive { get; set; }
```

## Property Value

[string](#)

## Model

The model name (required)

```
[JsonPropertyName("model")]
public string Model { get; set; }
```

## Property Value

[string](#)

## Options

Additional model parameters listed in the documentation for the Modelfile such as temperature

```
[JsonPropertyName("options")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public RequestOptions? Options { get; set; }
```

## Property Value

[RequestOptions](#)

## Prompt

The prompt to generate a response for

```
[JsonPropertyName("prompt")]
public string Prompt { get; set; }
```

## Property Value

[string](#)

## Raw

In some cases you may wish to bypass the templating system and provide a full prompt. In this case, you can use the raw parameter to disable formatting.

```
[JsonPropertyName("raw")]
public bool Raw { get; set; }
```

## Property Value

[bool](#) ↗

## Stream

If false the response will be returned as a single response object, rather than a stream of objects

```
[JsonPropertyName("stream")]
public bool Stream { get; set; }
```

## Property Value

[bool](#) ↗

## Suffix

Suffix for Fill-In-the-Middle generate

```
[JsonPropertyName("suffix")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string Suffix { get; set; }
```

## Property Value

[string](#) ↗

## System

System prompt to (overrides what is defined in the Modelfile)

```
[JsonPropertyName("system")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
```

```
public string? System { get; set; }
```

Property Value

[string ↗](#)

## Template

The full prompt or prompt template (overrides what is defined in the Modelfile)

```
[JsonPropertyName("template")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string? Template { get; set; }
```

Property Value

[string ↗](#)

# Class GenerateResponseStream

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

The response from the /api/generate endpoint when streaming is enabled

```
public class GenerateResponseStream
```

## Inheritance

[object](#) ← GenerateResponseStream

## Derived

[GenerateDoneResponseStream](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## CreatedAt

Gets or sets the time the response was generated.

```
[JsonIgnore]  
public DateTimeOffset? CreatedAt { get; set; }
```

## Property Value

[DateTimeOffset](#)?

## CreatedAtString

Gets or sets the time the response was generated.

```
[JsonPropertyName("created_at")]
public string? CreatedAtString { get; set; }
```

Property Value

string ↗

Done

Whether the response is complete

```
[JsonPropertyName("done")]
public bool Done { get; set; }
```

Property Value

bool ↗

Model

The model that generated the response

```
[JsonPropertyName("model")]
public string Model { get; set; }
```

Property Value

string ↗

Response

The response generated by the model

```
[JsonPropertyName("response")]
public string Response { get; set; }
```

Property Value

[string](#) ↗

# Class ListModelsResponse

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

List models that are available locally.

[Ollama API docs](#)

```
public class ListModelsResponse
```

## Inheritance

[object](#) ← ListModelsResponse

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Models

Gets or sets the array of models returned by the API.

```
[JsonPropertyName("models")]
public Model[] Models { get; set; }
```

## Property Value

[Model\[\]](#)

# Class ListRunningModelsResponse

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

List models that are currently loaded into memory.

[Ollama API docs](#)

```
[JsonUnmappedMemberHandling(JsonUnmappedMemberHandling.Skip)]  
public class ListRunningModelsResponse
```

## Inheritance

[object](#) ← ListRunningModelsResponse

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## RunningModels

An array of running models.

```
[JsonPropertyName("models")]  
public RunningModel[] RunningModels { get; set; }
```

## Property Value

[RunningModel\[\]](#)

# Class Model

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents a model with its associated metadata.

```
public class Model
```

## Inheritance

[object](#) ← Model

## Derived

[RunningModel](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Details

Gets or sets additional details about the model.

```
[JsonPropertyName("details")]
public Details Details { get; set; }
```

## Property Value

[Details](#)

## Digest

Gets or sets a cryptographic hash of the model file.

```
[JsonPropertyName("digest")]
public string Digest { get; set; }
```

## Property Value

[string](#)

## ModifiedAt

Gets or sets the time the model was created or last modified.

```
[JsonPropertyName("modified_at")]
public DateTime ModifiedAt { get; set; }
```

## Property Value

[DateTime](#)

## Name

Gets or sets the name of the model.

```
[JsonPropertyName("name")]
public string Name { get; set; }
```

## Property Value

[string](#)

## Size

Gets or sets the size of the model file in bytes.

```
[JsonPropertyName("size")]
public long Size { get; set; }
```

## Property Value

[long↗](#)

# Class ModelInfo

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents additional model information.

```
public class ModelInfo
```

## Inheritance

[object](#) ← ModelInfo

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Architecture

Gets or sets the architecture of the model.

```
[JsonPropertyName("general.architecture")]
public string? Architecture { get; set; }
```

## Property Value

[string](#)

## ExtraInfo

Gets or sets additional information as a dictionary.

```
[JsonExtensionData]
public IDictionary<string, object>? ExtraInfo { get; set; }
```

## Property Value

[IDictionary](#) <[string](#), [object](#)>

## FileType

Gets or sets the file type of the model.

```
[JsonPropertyName("general.file_type")]
public int? FileType { get; set; }
```

## Property Value

[int](#)?

## ParameterCount

Gets or sets the parameter count of the model.

```
[JsonPropertyName("general.parameter_count")]
public long? ParameterCount { get; set; }
```

## Property Value

[long](#)?

## QuantizationVersion

Gets or sets the quantization version of the model.

```
[JsonPropertyName("general.quantization_version")]
public int? QuantizationVersion { get; set; }
```

## Property Value

[int](#)?

# Class OllamaOption

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Collection of options available to Ollama

```
public class OllamaOption
```

Inheritance

[object](#) ← OllamaOption

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### OllamaOption(string)

Collection of options available to Ollama

```
public OllamaOption(string name)
```

Parameters

**name** [string](#)

The name of the setting like defined in the Ollama api docs

## Properties

### F16kv

Enable f16 key/value. (Default: False)

```
public static OllamaOption F16kv { get; }
```

Property Value

[OllamaOption](#)

## FrequencyPenalty

The penalty to apply to tokens based on their frequency in the prompt. (Default: 0.0)

```
public static OllamaOption FrequencyPenalty { get; }
```

Property Value

[OllamaOption](#)

## LogitsAll

Return logits for all the tokens, not just the last one. (Default: False)

```
public static OllamaOption LogitsAll { get; }
```

Property Value

[OllamaOption](#)

## LowVram

Enable low VRAM mode. (Default: False)

```
public static OllamaOption LowVram { get; }
```

Property Value

[OllamaOption](#)

## MainGpu

This option controls which GPU is used for small tensors. The overhead of splitting the computation across all GPUs is not worthwhile. The GPU will use slightly more VRAM to store a scratch buffer for temporary results. By default, GPU 0 is used.

```
public static OllamaOption MainGpu { get; }
```

### Property Value

[OllamaOption](#)

## MinP

Alternative to the top\_p, and aims to ensure a balance of quality and variety. min\_p represents the minimum probability for a token to be considered, relative to the probability of the most likely token. For example, with min\_p=0.05 and the most likely token having a probability of 0.9, logits with a value less than  $0.05 \times 0.9 = 0.045$  are filtered out. (Default: 0.0)

```
public static OllamaOption MinP { get; }
```

### Property Value

[OllamaOption](#)

## MiroStat

Enable Mirostat sampling for controlling perplexity. (default: 0, 0 = disabled, 1 = Mirostat, 2 = Mirostat 2.0)

```
public static OllamaOption MiroStat { get; }
```

### Property Value

[OllamaOption](#)

## MiroStatEta

Influences how quickly the algorithm responds to feedback from the generated text. A lower learning rate will result in slower adjustments, while a higher learning rate will make the algorithm more responsive. (Default: 0.1)

```
public static OllamaOption MiroStatEta { get; }
```

Property Value

[OllamaOption](#)

## MiroStatTau

Controls the balance between coherence and diversity of the output. A lower value will result in more focused and coherent text. (Default: 5.0)

```
public static OllamaOption MiroStatTau { get; }
```

Property Value

[OllamaOption](#)

## Name

Gets the name of the Ollama setting

```
public string Name { get; }
```

Property Value

[string](#) ↗

## NumBatch

Prompt processing maximum batch size. (Default: 512)

```
public static OllamaOption NumBatch { get; }
```

## Property Value

[OllamaOption](#)

## NumCtx

Sets the size of the context window used to generate the next token. (Default: 2048)

```
public static OllamaOption NumCtx { get; }
```

## Property Value

[OllamaOption](#)

## NumGpu

The number of layers to send to the GPU(s). On macOS it defaults to 1 to enable metal support, 0 to disable.

```
public static OllamaOption NumGpu { get; }
```

## Property Value

[OllamaOption](#)

## NumGqa

The number of GQA groups in the transformer layer. Required for some models, for example it is 8 for llama2:70b

```
public static OllamaOption NumGqa { get; }
```

## Property Value

## [OllamaOption](#)

### NumKeep

Number of tokens to keep from the initial prompt. (Default: 4, -1 = all)

```
public static OllamaOption NumKeep { get; }
```

### Property Value

#### [OllamaOption](#)

### NumPredict

Maximum number of tokens to predict when generating text. (Default: 128, -1 = infinite generation, -2 = fill context)

```
public static OllamaOption NumPredict { get; }
```

### Property Value

#### [OllamaOption](#)

### NumThread

Sets the number of threads to use during computation. By default, Ollama will detect this for optimal performance. It is recommended to set this value to the number of physical CPU cores your system has (as opposed to the logical number of cores).

```
public static OllamaOption NumThread { get; }
```

### Property Value

#### [OllamaOption](#)

## Numa

Enable NUMA support. (Default: False)

```
public static OllamaOption Numa { get; }
```

Property Value

[OllamaOption](#)

## PenalizeNewline

Penalize newline tokens (Default: True)

```
public static OllamaOption PenalizeNewline { get; }
```

Property Value

[OllamaOption](#)

## PresencePenalty

The penalty to apply to tokens based on their presence in the prompt. (Default: 0.0)

```
public static OllamaOption PresencePenalty { get; }
```

Property Value

[OllamaOption](#)

## RepeatLastN

Sets how far back for the model to look back to prevent repetition. (Default: 64, 0 = disabled, -1 = num\_ctx)

```
public static OllamaOption RepeatLastN { get; }
```

## Property Value

[OllamaOption](#)

## RepeatPenalty

Sets how strongly to penalize repetitions. A higher value (e.g., 1.5) will penalize repetitions more strongly, while a lower value (e.g., 0.9) will be more lenient. (Default: 1.1)

```
public static OllamaOption RepeatPenalty { get; }
```

## Property Value

[OllamaOption](#)

## Seed

Sets the random number seed to use for generation. Setting this to a specific number will make the model generate the same text for the same prompt. (Default: 0)

```
public static OllamaOption Seed { get; }
```

## Property Value

[OllamaOption](#)

## Stop

Sets the stop sequences to use. When this pattern is encountered the LLM will stop generating text and return. Multiple stop patterns may be set by specifying multiple separate stop parameters in a modelfile.

```
public static OllamaOption Stop { get; }
```

## Property Value

[OllamaOption](#)

## Temperature

The temperature of the model. Increasing the temperature will make the model answer more creatively.  
(Default: 0.8)

```
public static OllamaOption Temperature { get; }
```

Property Value

[OllamaOption](#)

## TfsZ

Tail free sampling is used to reduce the impact of less probable tokens from the output. A higher value (e.g., 2.0) will reduce the impact more, while a value of 1.0 disables this setting. (default: 1)

```
public static OllamaOption TfsZ { get; }
```

Property Value

[OllamaOption](#)

## TopK

Reduces the probability of generating nonsense. A higher value (e.g. 100) will give more diverse answers, while a lower value (e.g. 10) will be more conservative. (Default: 40)

```
public static OllamaOption TopK { get; }
```

Property Value

[OllamaOption](#)

## TopP

Works together with top-k. A higher value (e.g., 0.95) will lead to more diverse text, while a lower value (e.g., 0.5) will generate more focused and conservative text. (Default: 0.9)

```
public static OllamaOption TopP { get; }
```

Property Value

[OllamaOption](#)

## TypicalP

The typical-p value to use for sampling. Locally Typical Sampling implementation described in the paper <https://arxiv.org/abs/2202.00666>. (Default: 1.0)

```
public static OllamaOption TypicalP { get; }
```

Property Value

[OllamaOption](#)

## UseMlock

Lock the model in memory to prevent swapping. This can improve performance, but it uses more RAM and may slow down loading. (Default: False)

```
public static OllamaOption UseMlock { get; }
```

Property Value

[OllamaOption](#)

## UseMmap

Models are mapped into memory by default, which allows the system to load only the necessary parts as needed. Disabling mmap makes loading slower but reduces pageouts if you're not using mlock. If the model is bigger than your RAM, turning off mmap stops it from loading. (Default: True)

```
public static OllamaOption UseMmap { get; }
```

Property Value

[OllamaOption](#)

## VocabOnly

Load only the vocabulary, not the weights. (Default: False)

```
public static OllamaOption VocabOnly { get; }
```

Property Value

[OllamaOption](#)

# Class OllamaRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents the base class for requests to the Ollama API.

```
public abstract class OllamaRequest
```

## Inheritance

[object](#) ← OllamaRequest

## Derived

[ChatRequest](#), [CopyModelRequest](#), [CreateModelRequest](#), [DeleteModelRequest](#), [EmbedRequest](#),  
[GenerateRequest](#), [PullModelRequest](#), [PushModelRequest](#), [ShowModelRequest](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## CustomHeaders

Gets the custom headers to include with the request.

```
public Dictionary<string, string> CustomHeaders { get; }
```

## Property Value

[Dictionary](#)<[string](#), [string](#)>

# Class ProjectorInfo

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents projector-specific information.

```
public class ProjectorInfo
```

## Inheritance

[object](#) ← ProjectorInfo

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## ExtraInfo

Gets or sets additional projector information as a dictionary.

```
[JsonExtensionData]  
public IDictionary<string, object>? ExtraInfo { get; set; }
```

## Property Value

[IDictionary](#)<[string](#), [object](#)>

# Class PullModelRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Download a model from the ollama library. Cancelled pulls are resumed from where they left off, and multiple calls will share the same download progress.

[Ollama API docs](#)

```
public class PullModelRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← PullModelRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Insecure

Gets or sets a value indicating whether to allow insecure connections to the library. Only use this if you are pulling from your own library during development.

```
[JsonPropertyName("insecure")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? Insecure { get; set; }
```

### Property Value

[bool](#)?

## Model

Gets or sets the name of the model to pull.

```
[JsonPropertyName("model")]
public string? Model { get; set; }
```

Property Value

[string](#) ↗

## Stream

Gets or sets a value indicating whether to stream the response.

```
[JsonPropertyName("stream")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? Stream { get; set; }
```

Property Value

[bool](#) ↗?

# Class PullModelResponse

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents the streamed response from the /api/pull endpoint.

```
public class PullModelResponse
```

## Inheritance

[object](#) ← PullModelResponse

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Completed

Gets or sets the number of bytes pulled so far.

```
[JsonPropertyName("completed")]
public long Completed { get; set; }
```

### Property Value

[long](#)

### Digest

Gets or sets the hash of the model file.

```
[JsonPropertyName("digest")]
public string Digest { get; set; }
```

## Property Value

[string](#) ↗

## Percent

Gets the percentage of the pull operation that has been completed.

```
[JsonIgnore]  
public double Percent { get; }
```

## Property Value

[double](#) ↗

## Status

Gets or sets the status of the pull operation.

```
[JsonPropertyName("status")]  
public string Status { get; set; }
```

## Property Value

[string](#) ↗

## Total

Gets or sets the total number of bytes to pull.

```
[JsonPropertyName("total")]  
public long Total { get; set; }
```

## Property Value

[long](#) ↗

# Class PushModelRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Upload a model to a model library. Requires registering for ollama.ai and adding a public key first.

[Ollama API docs](#)

```
public class PushModelRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← PushModelRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Insecure

Gets or sets a value indicating whether to allow insecure connections to the library. Only use this if you are pulling from your own library during development.

```
[JsonPropertyName("insecure")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? Insecure { get; set; }
```

### Property Value

[bool](#)?

### Model

Gets or sets the name of the model to push in the form of namespace/model:tag.

```
[JsonPropertyName("model")]
public string? Model { get; set; }
```

## Property Value

string ↗

## Stream

Gets or sets a value indicating whether to stream the response.

```
[JsonPropertyName("stream")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? Stream { get; set; }
```

## Property Value

bool ↗?

# Class PushModelResponse

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents the response from the /api/push endpoint.

```
public class PushModelResponse
```

## Inheritance

[object](#) ← PushModelResponse

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Digest

Gets or sets the hash of the model file.

```
[JsonPropertyName("digest")]
public string Digest { get; set; }
```

### Property Value

[string](#)

### Status

Gets or sets the status of the push operation.

```
[JsonPropertyName("status")]
public string Status { get; set; }
```

## Property Value

[string](#)

## Total

Gets or sets the total number of bytes to push.

```
[JsonPropertyName("total")]
public int Total { get; set; }
```

## Property Value

[int](#)

# Class RequestOptions

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

The configuration information used for a chat completions request.

```
public class RequestOptions
```

## Inheritance

[object](#) ← RequestOptions

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### F16kv

Enable f16 key/value. (Default: False)

```
[JsonPropertyName("f16_kv")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? F16kv { get; set; }
```

### Property Value

[bool](#)?

### FrequencyPenalty

The penalty to apply to tokens based on their frequency in the prompt. (Default: 0.0)

```
[JsonPropertyName("frequency_penalty")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? FrequencyPenalty { get; set; }
```

## Property Value

[float](#)?

## LogitsAll

Return logits for all the tokens, not just the last one. (Default: False)

```
[JsonPropertyName("logits_all")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? LogitsAll { get; set; }
```

## Property Value

[bool](#)?

## LowVram

Enable low VRAM mode. (Default: False)

```
[JsonPropertyName("low_vram")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? LowVram { get; set; }
```

## Property Value

[bool](#)?

## MainGpu

This option controls which GPU is used for small tensors. The overhead of splitting the computation across all GPUs is not worthwhile. The GPU will use slightly more VRAM to store a scratch buffer for temporary results. By default, GPU 0 is used.

```
[JsonPropertyName("main_gpu")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? MainGpu { get; set; }
```

## Property Value

[int](#)?

## MinP

Alternative to the top\_p, and aims to ensure a balance of quality and variety. min\_p represents the minimum probability for a token to be considered, relative to the probability of the most likely token. For example, with min\_p=0.05 and the most likely token having a probability of 0.9, logits with a value less than 0.05\*0.9=0.045 are filtered out. (Default: 0.0)

```
[JsonPropertyName("min_p")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? MinP { get; set; }
```

## Property Value

[float](#)?

## MiroStat

Enable Mirostat sampling for controlling perplexity. (default: 0, 0 = disabled, 1 = Mirostat, 2 = Mirostat 2.0)

```
[JsonPropertyName("mirostat")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? MiroStat { get; set; }
```

## Property Value

[int](#)?

## MiroStatEta

Influences how quickly the algorithm responds to feedback from the generated text. A lower learning rate will result in slower adjustments, while a higher learning rate will make the algorithm more responsive. (Default: 0.1)

```
[JsonPropertyName("mirostat_eta")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? MiroStatEta { get; set; }
```

## Property Value

[float](#)?

## MiroStatTau

Controls the balance between coherence and diversity of the output. A lower value will result in more focused and coherent text. (Default: 5.0)

```
[JsonPropertyName("mirostat_tau")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? MiroStatTau { get; set; }
```

## Property Value

[float](#)?

## NumBatch

Prompt processing maximum batch size. (Default: 512)

```
[JsonPropertyName("num_batch")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? NumBatch { get; set; }
```

## Property Value

[int](#)?

## NumCtx

Sets the size of the context window used to generate the next token. (Default: 2048)

```
[JsonPropertyName("num_ctx")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? NumCtx { get; set; }
```

## Property Value

[int](#)?

## NumGpu

The number of layers to send to the GPU(s). On macOS it defaults to 1 to enable metal support, 0 to disable.

```
[JsonPropertyName("num_gpu")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? NumGpu { get; set; }
```

## Property Value

[int](#)?

## NumGqa

The number of GQA groups in the transformer layer. Required for some models, for example it is 8 for llama2:70b

```
[JsonPropertyName("num_gqa")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? NumGqa { get; set; }
```

## Property Value

[int](#)?

## NumKeep

Number of tokens to keep from the initial prompt. (Default: 4, -1 = all)

```
[JsonPropertyName("num_keep")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? NumKeep { get; set; }
```

## Property Value

[int](#)?

## NumPredict

Maximum number of tokens to predict when generating text. (Default: 128, -1 = infinite generation, -2 = fill context)

```
[JsonPropertyName("num_predict")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? NumPredict { get; set; }
```

## Property Value

[int](#)?

## NumThread

Sets the number of threads to use during computation. By default, Ollama will detect this for optimal performance. It is recommended to set this value to the number of physical CPU cores your system has (as opposed to the logical number of cores).

```
[JsonPropertyName("num_thread")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? NumThread { get; set; }
```

## Property Value

[int](#)?

## Numa

Enable NUMA support. (Default: False)

```
[JsonPropertyName("numa")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? Numa { get; set; }
```

Property Value

[bool](#)?

## PenalizeNewline

Penalize newline tokens (Default: True)

```
[JsonPropertyName("penalize_newline")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? PenalizeNewline { get; set; }
```

Property Value

[bool](#)?

## PresencePenalty

The penalty to apply to tokens based on their presence in the prompt. (Default: 0.0)

```
[JsonPropertyName("presence_penalty")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? PresencePenalty { get; set; }
```

Property Value

[float](#)?

## RepeatLastN

Sets how far back for the model to look back to prevent repetition. (Default: 64, 0 = disabled, -1 = num\_ctx)

```
[JsonPropertyName("repeat_last_n")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? RepeatLastN { get; set; }
```

Property Value

[int](#)?

## RepeatPenalty

Sets how strongly to penalize repetitions. A higher value (e.g., 1.5) will penalize repetitions more strongly, while a lower value (e.g., 0.9) will be more lenient. (Default: 1.1)

```
[JsonPropertyName("repeat_penalty")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? RepeatPenalty { get; set; }
```

Property Value

[float](#)?

## Seed

Sets the random number seed to use for generation. Setting this to a specific number will make the model generate the same text for the same prompt. (Default: 0)

```
[JsonPropertyName("seed")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? Seed { get; set; }
```

Property Value

[int](#)?

## Stop

Sets the stop sequences to use. When this pattern is encountered the LLM will stop generating text and return. Multiple stop patterns may be set by specifying multiple separate stop parameters in a modelfile.

```
[JsonPropertyName("stop")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string[]? Stop { get; set; }
```

## Property Value

[string](#)[]

## Temperature

The temperature of the model. Increasing the temperature will make the model answer more creatively.  
(Default: 0.8)

```
[JsonPropertyName("temperature")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? Temperature { get; set; }
```

## Property Value

[float](#)?

## TfsZ

Tail free sampling is used to reduce the impact of less probable tokens from the output. A higher value (e.g., 2.0) will reduce the impact more, while a value of 1.0 disables this setting. (default: 1)

```
[JsonPropertyName("tfs_z")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? TfsZ { get; set; }
```

## Property Value

[float](#)?

## TopK

Reduces the probability of generating nonsense. A higher value (e.g. 100) will give more diverse answers, while a lower value (e.g. 10) will be more conservative. (Default: 40)

```
[JsonPropertyName("top_k")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public int? TopK { get; set; }
```

### Property Value

[int](#)?

## TopP

Works together with top-k. A higher value (e.g., 0.95) will lead to more diverse text, while a lower value (e.g., 0.5) will generate more focused and conservative text. (Default: 0.9)

```
[JsonPropertyName("top_p")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? TopP { get; set; }
```

### Property Value

[float](#)?

## TypicalP

The typical-p value to use for sampling. Locally Typical Sampling implementation described in the paper <https://arxiv.org/abs/2202.00666>. (Default: 1.0)

```
[JsonPropertyName("typical_p")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public float? TypicalP { get; set; }
```

### Property Value

[float](#)?

## UseMlock

Lock the model in memory to prevent swapping. This can improve performance, but it uses more RAM and may slow down loading. (Default: False)

```
[JsonPropertyName("use_mlock")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? UseMlock { get; set; }
```

### Property Value

[bool](#)?

## UseMmap

Models are mapped into memory by default, which allows the system to load only the necessary parts as needed. Disabling mmap makes loading slower but reduces pageouts if you're not using mlock. If the model is bigger than your RAM, turning off mmap stops it from loading. (Default: True)

```
[JsonPropertyName("use_mmap")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? UseMmap { get; set; }
```

### Property Value

[bool](#)?

## VocabOnly

Load only the vocabulary, not the weights. (Default: False)

```
[JsonPropertyName("vocab_only")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public bool? VocabOnly { get; set; }
```

Property Value

bool?

# Class RunningModel

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents a running model.

```
[JsonUnmappedMemberHandling(JsonUnmappedMemberHandling.Skip)]  
public class RunningModel : Model
```

## Inheritance

[object](#) ← [Model](#) ← RunningModel

## Inherited Members

[Model.Name](#) , [Model.ModifiedAt](#) , [Model.Size](#) , [Model.Digest](#) , [Model.Details](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## ExpiresAt

The time the model will be unloaded from memory.

```
[JsonPropertyName("expires_at")]  
public DateTime ExpiresAt { get; set; }
```

## Property Value

[DateTime](#)

## SizeVram

The amount of vram (in bytes) used by the model.

```
[JsonPropertyName("size_vram")]  
public long SizeVram { get; set; }
```

## Property Value

[long ↗](#)

# Class ShowModelRequest

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Show information about a model including details, modelfile, template, parameters, license, system prompt.

[Ollama API docs](#)

```
[JsonUnmappedMemberHandling(JsonUnmappedMemberHandling.Skip)]  
public class ShowModelRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← ShowModelRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Model

Gets or sets the name of the model to show.

```
[JsonPropertyName("model")]  
public string? Model { get; set; }
```

## Property Value

[string](#)

# Class ShowModelResponse

Namespace: [OllamaSharp.Models](#)

Assembly: OllamaSharp.dll

Represents the response containing detailed model information.

```
public class ShowModelResponse
```

## Inheritance

[object](#) ← ShowModelResponse

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Details

Gets or sets additional details about the model.

```
[JsonPropertyName("details")]
public Details Details { get; set; }
```

## Property Value

[Details](#)

## Info

Gets or sets extra information about the model.

```
[JsonPropertyName("model_info")]
public ModelInfo Info { get; set; }
```

## Property Value

[ModellInfo](#)

## License

Gets or sets the license for the model.

```
[JsonPropertyName("license")]
public string? License { get; set; }
```

## Property Value

[string](#)

## Modelfile

Gets or sets the Modelfile for the model.

```
[JsonPropertyName("modelfile")]
public string? Modelfile { get; set; }
```

## Property Value

[string](#)

## Parameters

Gets or sets the parameters for the model.

```
[JsonPropertyName("parameters")]
public string? Parameters { get; set; }
```

## Property Value

[string](#)

## Projector

Gets or sets extra information about the projector.

```
[JsonPropertyName("projector_info")]
public ProjectorInfo? Projector { get; set; }
```

## Property Value

[ProjectorInfo](#)

## System

Gets or sets the system prompt for the model.

```
[JsonPropertyName("system")]
public string? System { get; set; }
```

## Property Value

[string](#)

## Template

Gets or sets the template for the model.

```
[JsonPropertyName("template")]
public string? Template { get; set; }
```

## Property Value

[string](#)

# Namespace OllamaSharp.Models.Chat

## Classes

### [ChatDoneResponseStream](#)

Represents the final message in a stream of responses from the /api/chat endpoint.

### [ChatRequest](#)

Represents a request to generate a chat completion using the specified model and parameters.

### [ChatResponseStream](#)

Represents a streamed response from a chat model in the Ollama API.

### [Function](#)

Represents a function that can be executed by a tool.

### [Message](#)

Represents a message in a chat.

### [Message.Function](#)

Represents a function that can be called by a tool.

### [Message.ToolCall](#)

Represents a tool call within a message.

### [MessageBuilder](#)

A message builder that can build messages from streamed chunks

### [Parameters](#)

Represents the parameters required by a function, including their properties and required fields.

### [Properties](#)

Represents a property within a function's parameters, including its type, description, and possible values.

### [Tool](#)

Represents a tool that the model can use, if supported.

## Structs

### [ChatRole](#)

Represents a role within a chat completions interaction, describing the intended purpose of a message.

# Class ChatDoneResponseStream

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents the final message in a stream of responses from the /api/chat endpoint.

```
public class ChatDoneResponseStream : ChatResponseStream
```

## Inheritance

[object](#) ← [ChatResponseStream](#) ← ChatDoneResponseStream

## Inherited Members

[ChatResponseStream.Model](#) , [ChatResponseStream.CreatedAtString](#) , [ChatResponseStream.CreatedAt](#) ,  
[ChatResponseStream.Message](#) , [ChatResponseStream.Done](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### DoneReason

The reason for the completion of the chat

```
[JsonPropertyName("done_reason")]
public string? DoneReason { get; set; }
```

### Property Value

[string](#)

### EvalCount

The number of tokens in the response

```
[JsonPropertyName("eval_count")]
```

```
public int EvalCount { get; set; }
```

Property Value

[int](#)

## EvalDuration

The time in nanoseconds spent generating the response

```
[JsonPropertyName("eval_duration")]
public long EvalDuration { get; set; }
```

Property Value

[long](#)

## LoadDuration

The time spent in nanoseconds loading the model

```
[JsonPropertyName("load_duration")]
public long LoadDuration { get; set; }
```

Property Value

[long](#)

## PromptEvalCount

The number of tokens in the prompt

```
[JsonPropertyName("prompt_eval_count")]
public int PromptEvalCount { get; set; }
```

Property Value

[int](#) ↗

## PromptEvalDuration

The time spent in nanoseconds evaluating the prompt

```
[JsonPropertyName("prompt_eval_duration")]
public long PromptEvalDuration { get; set; }
```

Property Value

[long](#) ↗

## TotalDuration

The time spent generating the response

```
[JsonPropertyName("total_duration")]
public long TotalDuration { get; set; }
```

Property Value

[long](#) ↗

# Class ChatRequest

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a request to generate a chat completion using the specified model and parameters.

```
public class ChatRequest : OllamaRequest
```

## Inheritance

[object](#) ← [OllamaRequest](#) ← ChatRequest

## Inherited Members

[OllamaRequest.CustomHeaders](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,  
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Format

Gets or sets the format to return a response in. Currently only accepts "json" or null.

```
[JsonPropertyName("format")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string? Format { get; set; }
```

### Property Value

[string](#)

### KeepAlive

Gets or sets the KeepAlive property, which decides how long a given model should stay loaded.

```
[JsonPropertyName("keep_alive")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
```

```
public string? KeepAlive { get; set; }
```

## Property Value

[string](#)

## Messages

Gets or sets the messages of the chat, this can be used to keep a chat memory.

```
[JsonPropertyName("messages")]
public IEnumerable<Message>? Messages { get; set; }
```

## Property Value

[IEnumerable](#) <[Message](#)>

## Model

Gets or sets the model name (required).

```
[JsonPropertyName("model")]
public string Model { get; set; }
```

## Property Value

[string](#)

## Options

Gets or sets additional model parameters listed in the documentation for the Modelfile such as temperature.

```
[JsonPropertyName("options")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public RequestOptions? Options { get; set; }
```

## Property Value

[RequestOptions](#)

## Stream

Gets or sets a value indicating whether the response will be returned as a single response object rather than a stream of objects.

```
[JsonPropertyName("stream")]
public bool Stream { get; set; }
```

## Property Value

[bool](#) ↗

## Template

Gets or sets the full prompt or prompt template (overrides what is defined in the Modelfile).

```
[JsonPropertyName("template")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string? Template { get; set; }
```

## Property Value

[string](#) ↗

## Tools

Gets or sets the tools for the model to use if supported. Requires stream to be set to false.

```
[JsonPropertyName("tools")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public IEnumerable<Tool>? Tools { get; set; }
```

## Property Value

[IEnumerable](#) <Tool>

# Class ChatResponseStream

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a streamed response from a chat model in the Ollama API.

```
public class ChatResponseStream
```

## Inheritance

[object](#) ← ChatResponseStream

## Derived

[ChatDoneResponseStream](#)

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## CreatedAt

Gets or sets the time the response was generated.

```
[JsonIgnore]  
public DateTimeOffset? CreatedAt { get; set; }
```

## Property Value

[DateTimeOffset](#)?

## CreatedAtString

Gets or sets the time the response was generated.

```
[JsonPropertyName("created_at")]
public string? CreatedAtString { get; set; }
```

## Property Value

[string](#)

## Done

Gets or sets a value indicating whether the response is complete.

```
[JsonPropertyName("done")]
public bool Done { get; set; }
```

## Property Value

[bool](#)

## Message

Gets or sets the message returned by the model.

```
[JsonPropertyName("message")]
public Message Message { get; set; }
```

## Property Value

[Message](#)

## Model

Gets or sets the model that generated the response.

```
[JsonPropertyName("model")]
public string Model { get; set; }
```

## Property Value

[string](#) ↗

# Struct ChatRole

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a role within a chat completions interaction, describing the intended purpose of a message.

```
[JsonConverter(typeof(ChatRoleConverter))]  
public readonly struct ChatRole : IEquatable<ChatRole>
```

Implements

[IEquatable](#) <[ChatRole](#)>

Inherited Members

[ValueType.Equals\(object\)](#) , [ValueType.GetHashCode\(\)](#) , [object.Equals\(object, object\)](#) ,  
[object.GetType\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

## Constructors

### ChatRole(object)

Initializes a new instance of [ChatRole](#) using a JSON constructor.

```
[JsonConstructor]  
public ChatRole(object _)
```

Parameters

\_ [object](#)

The placeholder parameter for JSON constructor.

### ChatRole(string?)

Initializes a new instance of [ChatRole](#) with the specified role.

```
public ChatRole(string? role)
```

## Parameters

### `role` [string](#)

The role to initialize with.

## Exceptions

### [ArgumentNullException](#)

Thrown when `role` is null.

## Properties

### Assistant

Gets the role that provides responses to system-instructed, user-prompted input.

```
public static ChatRole Assistant { get; }
```

### Property Value

#### [ChatRole](#)

### System

Gets the role that instructs or sets the behavior of the assistant.

```
public static ChatRole System { get; }
```

### Property Value

#### [ChatRole](#)

### Tool

Gets the role that is used to input the result from an external tool.

```
public static ChatRole Tool { get; }
```

Property Value

[ChatRole](#)

## User

Gets the role that provides input for chat completions.

```
public static ChatRole User { get; }
```

Property Value

[ChatRole](#)

## Methods

### Equals(ChatRole)

```
public bool Equals(ChatRole other)
```

Parameters

**other** [ChatRole](#)

Returns

[bool](#)

### ToString()

```
public override string ToString()
```

Returns

[string](#)

## Operators

### operator ==(ChatRole, ChatRole)

Determines if two [ChatRole](#) instances are equal.

```
public static bool operator ==(ChatRole left, ChatRole right)
```

Parameters

**left** [ChatRole](#)

The first [ChatRole](#) to compare.

**right** [ChatRole](#)

The second [ChatRole](#) to compare.

Returns

[bool](#)

**true** if both instances are equal; otherwise, **false**.

### implicit operator ChatRole(string)

Implicitly converts a string to a [ChatRole](#).

```
public static implicit operator ChatRole(string value)
```

Parameters

**value** [string](#)

The string value to convert.

Returns

[ChatRole](#)

## operator !=(ChatRole, ChatRole)

Determines if two [ChatRole](#) instances are not equal.

```
public static bool operator !=(ChatRole left, ChatRole right)
```

Parameters

**left** [ChatRole](#)

The first [ChatRole](#) to compare.

**right** [ChatRole](#)

The second [ChatRole](#) to compare.

Returns

[bool](#) ↗

**true** if both instances are not equal; otherwise, **false**.

# Class Function

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a function that can be executed by a tool.

```
public class Function
```

## Inheritance

[object](#) ← Function

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Description

Gets or sets the description of the function.

```
[JsonPropertyName("description")]
public string? Description { get; set; }
```

## Property Value

[string](#)

## Name

Gets or sets the name of the function.

```
[JsonPropertyName("name")]
public string? Name { get; set; }
```

## Property Value

[string](#) ↗

## Parameters

Gets or sets the parameters required by the function.

```
[JsonPropertyName("parameters")]
public Parameters? Parameters { get; set; }
```

## Property Value

[Parameters](#)

# Class Message

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a message in a chat.

```
public class Message
```

## Inheritance

[object](#) ← Message

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Constructors

### Message()

Initializes a new instance of the [Message](#) class. Required for JSON deserialization.

```
public Message()
```

### Message(ChatRole, string, string[]?)

Initializes a new instance of the [Message](#) class with the specified role, content, and images.

```
public Message(ChatRole role, string content, string[]? images)
```

#### Parameters

##### role [ChatRole](#)

The role of the message, either system, user, or assistant.

##### content [string](#)

The content of the message.

**images** [string](#)[]

An array of base64-encoded images.

## Message(ChatRole, string[])

Initializes a new instance of the [Message](#) class with the specified role and images.

```
public Message(ChatRole role, string[] images)
```

### Parameters

**role** [ChatRole](#)

The role of the message, either system, user, or assistant.

**images** [string](#)[]

An array of base64-encoded images.

## Message(ChatRole?, string)

Initializes a new instance of the [Message](#) class with the specified role and content.

```
public Message(ChatRole? role, string content)
```

### Parameters

**role** [ChatRole?](#)

The role of the message, either system, user, or assistant.

**content** [string](#)[]

The content of the message.

# Properties

## Content

Gets or sets the content of the message.

```
[JsonPropertyName("content")]
public string? Content { get; set; }
```

### Property Value

[string](#)

## Images

Gets or sets an array of base64-encoded images (for multimodal models such as llava).

```
[JsonPropertyName("images")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public string[]? Images { get; set; }
```

### Property Value

[string](#)[]

## Role

Gets or sets the role of the message, either system, user, or assistant.

```
[JsonPropertyName("role")]
public ChatRole? Role { get; set; }
```

### Property Value

[ChatRole](#)?

## ToolCalls

Gets or sets a list of tools the model wants to use (for models that support function calls, such as `llama3.1`).

```
[JsonPropertyName("tool_calls")]
[JsonIgnore(Condition = JsonIgnoreCondition.WhenWritingNull)]
public IEnumerable<Message.ToolCall>? ToolCalls { get; set; }
```

Property Value

[IEnumerable](#) <[Message.ToolCall](#)>

# Class Message.Function

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a function that can be called by a tool.

```
public class Message.Function
```

## Inheritance

[object](#) ← Message.Function

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Arguments

Gets or sets the arguments for the function, represented as a dictionary of argument names and values.

```
[JsonPropertyName("arguments")]
public IDictionary<string, object>? Arguments { get; set; }
```

## Property Value

[IDictionary](#)<[string](#), [object](#)>

## Name

Gets or sets the name of the function.

```
[JsonPropertyName("name")]
public string? Name { get; set; }
```

## Property Value

[string](#) ↗

# Class Message.ToolCall

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a tool call within a message.

```
public class Message.ToolCall
```

## Inheritance

[object](#) ← Message.ToolCall

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Function

Gets or sets the function to be called by the tool.

```
[JsonPropertyName("function")]
public Message.Function? Function { get; set; }
```

## Property Value

[Message.Function](#)

# Class MessageBuilder

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

A message builder that can build messages from streamed chunks

```
public class MessageBuilder
```

## Inheritance

[object](#) ← MessageBuilder

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## HasValue

Gets whether the message builder received message chunks yet

```
public bool HasValue { get; }
```

## Property Value

[bool](#)

## Images

Base64-encoded images (for multimodal models such as llava)

```
public List<string> Images { get; set; }
```

## Property Value

[List](#) <[string](#)>

## Role

The role of the message, either system, user or assistant

```
public ChatRole? Role { get; set; }
```

### Property Value

[ChatRole?](#)

## ToolCalls

A list of tools the model wants to use (for models that support function calls, such as llama3.1)

```
public List<Message.ToolCall> ToolCalls { get; set; }
```

### Property Value

[List](#) <[Message.ToolCall](#)>

## Methods

### Append(ChatResponseStream?)

Appends a message chunk to build the final message

```
public void Append(ChatResponseStream? chunk)
```

### Parameters

chunk [ChatResponseStream](#)

The message chunk to append to the final message

## ToMessage()

Builds the message out of the streamed chunks that were appended before

```
public Message ToMessage()
```

Returns

[Message](#)

# Class Parameters

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents the parameters required by a function, including their properties and required fields.

```
public class Parameters
```

## Inheritance

[object](#) ← Parameters

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

# Properties

## Properties

Gets or sets the properties of the parameters with their respective types and descriptions.

```
[JsonPropertyName("properties")]
public Dictionary<string, Properties>? Properties { get; set; }
```

## Property Value

[Dictionary](#)<[string](#), [Properties](#)>

## Required

Gets or sets a list of required fields within the parameters.

```
[JsonPropertyName("required")]
public IEnumerable<string>? Required { get; set; }
```

## Property Value

[IEnumerable](#) <[string](#)>

# Class Properties

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a property within a function's parameters, including its type, description, and possible values.

```
public class Properties
```

Inheritance

[object](#) ← Properties

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Description

Gets or sets the description of the property.

```
[JsonPropertyName("description")]
public string? Description { get; set; }
```

### Property Value

[string](#)

## Enum

Gets or sets an enumeration of possible values for the property.

```
[JsonPropertyName("enum")]
public IEnumerable<string>? Enum { get; set; }
```

## Property Value

[IEnumerable](#)<[string](#)>

## Type

Gets or sets the type of the property.

```
[JsonPropertyName("type")]
public string? Type { get; set; }
```

## Property Value

[string](#)

# Class Tool

Namespace: [OllamaSharp.Models.Chat](#)

Assembly: OllamaSharp.dll

Represents a tool that the model can use, if supported.

```
public class Tool
```

Inheritance

[object](#) ← Tool

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Properties

### Function

Gets or sets the function definition associated with this tool.

```
[JsonPropertyName("function")]
public Function? Function { get; set; }
```

Property Value

[Function](#)

# Namespace OllamaSharp.Models.Chat. Converter

## Classes

### [ChatRoleConverter](#)

Converts a [ChatRole](#) to and from JSON.

# Class ChatRoleConverter

Namespace: [OllamaSharp.Models.Chat.Converter](#)

Assembly: OllamaSharp.dll

Converts a [ChatRole](#) to and from JSON.

```
public class ChatRoleConverter : JsonConverter<ChatRole>
```

## Inheritance

[object](#) ← [JsonConverter](#) ← [JsonConverter](#)<[ChatRole](#)> ← ChatRoleConverter

## Inherited Members

[JsonConverter<ChatRole>.CanConvert\(Type\)](#) ,  
[JsonConverter<ChatRole>.ReadAsPropertyName\(ref Utf8JsonReader, Type, JsonSerializerOptions\)](#) ,  
[JsonConverter<ChatRole>.WriteAsPropertyName\(Utf8JsonWriter, ChatRole, JsonSerializerOptions\)](#) ,  
[JsonConverter<ChatRole>.HandleNull](#) , [JsonConverter<ChatRole>.Type](#) , [object.Equals\(object\)](#) ,  
[object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

## Methods

### Read(ref Utf8JsonReader, Type, JsonSerializerOptions)

Reads and converts the JSON representation of a [ChatRole](#).

```
public override ChatRole Read(ref Utf8JsonReader reader, Type typeToConvert,  
JsonSerializerOptions options)
```

#### Parameters

reader [Utf8JsonReader](#)

The reader to read from.

typeToConvert [Type](#)

The type of the object to convert.

`options` [JsonSerializerOptions](#)

Options to control the conversion.

Returns

[ChatRole](#)

The [ChatRole](#) value.

## Write(Utf8JsonWriter, ChatRole, JsonSerializerOptions)

Writes a [ChatRole](#) as a JSON string.

```
public override void Write(Utf8JsonWriter writer, ChatRole value,  
JsonSerializerOptions options)
```

Parameters

`writer` [Utf8JsonWriter](#)

The writer to write to.

`value` [ChatRole](#)

The [ChatRole](#) value to write.

`options` [JsonSerializerOptions](#)

Options to control the conversion.

# Namespace OllamaSharp.Models.Exceptions

## Classes

### [ModelDoesNotSupportToolsException](#)

Represents an exception thrown when a model does not support the requested tools.

### [OllamaException](#)

Represents errors that occur during Ollama API operations.

# Class ModelDoesNotSupportToolsException

Namespace: [OllamaSharp.Models.Exceptions](#)

Assembly: OllamaSharp.dll

Represents an exception thrown when a model does not support the requested tools.

```
public class ModelDoesNotSupportToolsException : OllamaException, ISerializable
```

## Inheritance

[object](#) ← [Exception](#) ← [OllamaException](#) ← ModelDoesNotSupportToolsException

## Implements

[ISerializable](#)

## Inherited Members

[Exception.GetBaseException\(\)](#) , [Exception.GetObjectData\(SerializationInfo, StreamingContext\)](#) ,  
[Exception.GetType\(\)](#) , [Exception.ToString\(\)](#) , [Exception.Data](#) , [Exception.HelpLink](#) ,  
[Exception.HResult](#) , [Exception.InnerException](#) , [Exception.Message](#) , [Exception.Source](#) ,  
[Exception.StackTrace](#) , [Exception.TargetSite](#) , [Exception.SerializeObjectState](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

## Constructors

### ModelDoesNotSupportToolsException()

Initializes a new instance of the [ModelDoesNotSupportToolsException](#) class.

```
public ModelDoesNotSupportToolsException()
```

### ModelDoesNotSupportToolsException(string)

Initializes a new instance of the [ModelDoesNotSupportToolsException](#) class with a specified error message.

```
public ModelDoesNotSupportToolsException(string message)
```

## Parameters

**message** [string](#)

The message that describes the error.

## ModelDoesNotSupportToolsException(string, Exception)

Initializes a new instance of the [ModelDoesNotSupportToolsException](#) class with a specified error message and a reference to the inner exception that is the cause of this exception.

```
public ModelDoesNotSupportToolsException(string message, Exception innerException)
```

## Parameters

**message** [string](#)

The error message that explains the reason for the exception.

**innerException** [Exception](#)

The exception that is the cause of the current exception, or a null reference if no inner exception is specified.

# Class OllamaException

Namespace: [OllamaSharp.Models.Exceptions](#)

Assembly: OllamaSharp.dll

Represents errors that occur during Ollama API operations.

```
public class OllamaException : Exception, ISerializable
```

## Inheritance

[object](#) ← [Exception](#) ← OllamaException

## Implements

[ISerializable](#)

## Derived

[ModelDoesNotSupportToolsException](#)

## Inherited Members

[Exception.GetBaseException\(\)](#) , [Exception.GetObjectData\(SerializationInfo, StreamingContext\)](#) ,  
[Exception.GetType\(\)](#) , [Exception.ToString\(\)](#) , [Exception.Data](#) , [Exception.HelpLink](#) ,  
[Exception.HResult](#) , [Exception.InnerException](#) , [Exception.Message](#) , [Exception.Source](#) ,  
[Exception.StackTrace](#) , [Exception.TargetSite](#) , [Exception.SerializeObjectState](#) ,  
[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#)

## Constructors

### OllamaException()

Initializes a new instance of the [OllamaException](#) class.

```
public OllamaException()
```

### OllamaException(string)

Initializes a new instance of the [OllamaException](#) class with a specified error message.

```
public OllamaException(string message)
```

## Parameters

**message** [string](#)

The message that describes the error.

## OllamaException(string, Exception)

Initializes a new instance of the [OllamaException](#) class with a specified error message and a reference to the inner exception that is the cause of this exception.

```
public OllamaException(string message, Exception innerException)
```

## Parameters

**message** [string](#)

The error message that explains the reason for the exception.

**innerException** [Exception](#)

The exception that is the cause of the current exception, or a null reference if no inner exception is specified.