

# Namespace GoogleMapsWrapper.Api

## Classes

### [GeocodeApi](#)

Provides methods for retrieving data from Google's Geocoding API.

### [GoogleMapsApi](#)

The entry point of the wrapper. Exposes API objects to interact with the endpoint:

### [StaticMapCustomIcon](#)

An object that represents a custom icon for a marker. The source must be a Uri of an image.

### [StaticMapsApi](#)

Provides methods for retrieving a static image of a map from Google's Static API. This object leverages an [IApiEngine](#) to send [IRequests](#) and return data in form of [IResponse<TResponse>](#)s.

# Class GeocodeApi

Namespace: [GoogleMapsWrapper.Api](#)

Assembly: GoogleMapsWrapper.dll

Provides methods for retrieving data from Google's Geocoding API.

```
public class GeocodeApi
```

## Inheritance

[object](#) ← GeocodeApi

## Inherited Members

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

## Constructors

### GeocodeApi(IApiEngine)

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

```
public GeocodeApi(IApiEngine engine)
```

## Parameters

### engine [IApiEngine](#)

An instance of the API engine.

## Properties

### ApiType

```
public ApiTypes ApiType { get; }
```

## Property Value

[ApiTypes](#)

## Methods

### GeocodeAsync(GpsCoordinate, string?)

Makes a request to the API endpoint for geocoding a lat/lng coordinate.

```
public Task<IResponse<JsonDocument>> GeocodeAsync(GpsCoordinate coordinate, string?  
identifier = null)
```

#### Parameters

**coordinate** [GpsCoordinate](#)

GpsCoordinate to target the request.

**identifier** [string](#)

A string appended to the request object for tracking or identification.

#### Returns

[Task](#)<[IResponse](#)<[JsonDocument](#)>>

An [IResponse<TResponse>](#) containing the raw JSON response from the Google API.

### GeocodeAsync(string, string, string?)

Makes a request to the API endpoint for geocoding a lat/lng coordinate.

```
public Task<IResponse<JsonDocument>> GeocodeAsync(string latitude, string longitude, string?  
identifier = null)
```

#### Parameters

**latitude** [string](#)

Decimal degrees gps latitude.

**longitude** [string](#)

Decimal degrees gps longitude.

**identifier** [string](#)

A string appended to the request object for tracking or identification.

Returns

[Task](#)<[IResponse](#)<[JsonDocument](#)>>

An [IResponse<TResponse>](#) containing the raw JSON response from the Google API.

## GeocodeParseAsync(GpsCoordinate, string?)

Makes a request to the API endpoint for geocoding a lat/lng coordinate.

```
public Task<GeocodeContainer> GeocodeParseAsync(GpsCoordinate coordinate, string? identifier = null)
```

Parameters

**coordinate** [GpsCoordinate](#)

GpsCoordinate to target the request.

**identifier** [string](#)

A string appended to the request object for tracking or identification.

Returns

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

An [IResponse<TResponse>](#) containing the raw JSON response from the Google API.

## GeocodeParseAsync(string, string, string?)

Makes a request to the API endpoint for geocoding a lat/lng coordinate.

```
public Task<GeocodeContainer> GeocodeParseAsync(string latitude, string longitude, string?  
identifier = null)
```

## Parameters

**latitude** [string](#)

Decimal degrees gps latitude.

**longitude** [string](#)

Decimal degrees gps longitude.

**identifier** [string](#)

A string appended to the request object for tracking or identification.

## Returns

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

An [IResponse<TResponse>](#) containing the raw JSON response from the Google API.

## GetElevationAsync(GpsCoordinate, string?)

Makes a request to the API endpoint for acquiring elevation data about a lat/lng coordinate.

```
public Task<IResponse<JsonDocument>> GetElevationAsync(GpsCoordinate coordinate, string?  
identifier = null)
```

## Parameters

**coordinate** [GpsCoordinate](#)

GpsCoordinate to target the request about.

**identifier** [string](#)

A string appended to the request object for tracking or identification.

## Returns

[Task](#)<[IResponse](#)<[JsonDocument](#)>>

An [IResponse<TResponse>](#) containing the raw JSON response from the Google API.

## GetElevationAsync(string, string, string?)

Makes a request to the API endpoint for acquiring elevation data about a lat/lng coordinate.

```
public Task<IResponse<JsonDocument>> GetElevationAsync(string latitude, string longitude, string? identifier = null)
```

## Parameters

**latitude** [string](#)

Decimal degrees gps latitude.

**longitude** [string](#)

Decimal degrees gps longitude.

**identifier** [string](#)

A string appended to the request object for tracking or identification.

## Returns

[Task](#)<[IResponse](#)<[JsonDocument](#)>>

An [IResponse<TResponse>](#) containing the raw JSON response from the Google API.

## GetElevationParsedAsync(GpsCoordinate, string?)

Makes a request to the API endpoint for acquiring elevation data about a lat/lng coordinate.

```
public Task<ElevationContainer> GetElevationParsedAsync(GpsCoordinate coordinate, string? identifier = null)
```

## Parameters

### `coordinate` [GpsCoordinate](#)

GpsCoordinate to target the request about.

### `identifier` [string](#)

A string appended to the request object for tracking or identification.

## Returns

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

An [IResponse<TResponse>](#) containing the raw JSON response from the Google API.

## GetElevationParsedAsync(string, string, string?)

Makes a request to the API endpoint for acquiring elevation data about a lat/lng coordinate.

```
public Task<ElevationContainer> GetElevationParsedAsync(string latitude, string longitude,  
string? identifier = null)
```

## Parameters

### `latitude` [string](#)

Decimal degrees gps latitude.

### `longitude` [string](#)

Decimal degrees gps longitude.

### `identifier` [string](#)

A string appended to the request object for tracking or identification.

## Returns

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

An [IResponse<TResponse>](#) containing the raw JSON response from the Google API.

# Class GoogleMapsApi

Namespace: [GoogleMapsWrapper.Api](#)

Assembly: GoogleMapsWrapper.dll

The entry point of the wrapper. Exposes API objects to interact with the endpoint:

```
public class GoogleMapsApi
```

## Inheritance

[object](#) ← GoogleMapsApi

## Inherited Members

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

## Constructors

### GoogleMapsApi(HttpClient, IConfiguration)

Initializes a new instance of the class, using the provided [HttpClient](#) and configuration settings.

```
public GoogleMapsApi(HttpClient httpClient, IConfiguration config)
```

## Parameters

**httpClient** [HttpClient](#)

An [HttpClient](#) to be used to process web requests.

**config** [IConfiguration](#)

An [IConfiguration](#) which contains the API key.

## Properties

### GeocodeApi

An instance of the GeocodeApi.

```
public GeocodeApi GeocodeApi { get; }
```

Property Value

[GeocodeApi](#)

## StaticMapsApi

An instance of the StaticMapsApi.

```
public StaticMapsApi StaticMapsApi { get; }
```

Property Value

[StaticMapsApi](#)

# Class StaticMapCustomIcon

Namespace: [GoogleMapsWrapper.Api](#)

Assembly: GoogleMapsWrapper.dll

An object that represents a custom icon for a marker. The source must be a Uri of an image.

```
public record StaticMapCustomIcon : IEquatable<StaticMapCustomIcon>
```

## Inheritance

[object](#) ← StaticMapCustomIcon

## Implements

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

## Inherited Members

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

# Constructors

## StaticMapCustomIcon(string, MarkerIconAnchorTypes)

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

```
public StaticMapCustomIcon(string uri, MarkerIconAnchorTypes anchorType  
= MarkerIconAnchorTypes.Center)
```

## Parameters

uri [string](#)

anchorType [MarkerIconAnchorTypes](#)

How the icon is anchored.

## Exceptions

[ArgumentException](#)

Uri provided contains invalid image format.

## Properties

### AnchorType

```
public MarkerIconAnchorTypes AnchorType { get; init; }
```

#### Property Value

[MarkerIconAnchorTypes](#)

### Uri

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

#### Property Value

[Uri↗](#)

# Class StaticMapsApi

Namespace: [GoogleMapsWrapper.Api](#)

Assembly: GoogleMapsWrapper.dll

Provides methods for retrieving a static image of a map from Google's Static API. This object leverages an [IApiEngine](#) to send [IRequests](#) and return data in form of [IResponses<TResponse>](#)s.

```
public class StaticMapsApi
```

## Inheritance

[object](#) ← StaticMapsApi

## Inherited Members

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

## Constructors

### StaticMapsApi(IApiEngine)

Initializes a new instance of the StaticMapsAPI.

```
public StaticMapsApi(IApiEngine engine)
```

## Parameters

engine [IApiEngine](#)

An [IApiEngine](#) to process API requests and responses./>

## Methods

### GetMapAsync(Map, IEnumerable<Marker>?, IEnumerable<Polyline>?, string?)

Makes a request to the API endpoint for a static map image.

```
public Task<IResponse<byte[]>> GetMapAsync(Map mapSettings, IEnumerable<Marker>? markers = null, IEnumerable<Polyline>? paths = null, string? identifier = null)
```

## Parameters

### mapSettings [Map](#)

A [Map](#) element that defines key features about the map.>

### markers [IEnumerable](#)<[Marker](#)>

Optional. [Markers](#) to be included in the map.

### paths [IEnumerable](#)<[Polyline](#)>

Optional. [Polylines](#) to be included in the map.

### identifier [string](#)

A string appended to the request object for tracking or identification.

## Returns

### [Task](#)<[IResponse](#)<[byte](#)[]>>

An [IResponse](#) containing the raw byte[] response from the Google API.

## GetMapBytesAsync(Map, IEnumerable<Marker>?, IEnumerable<Polyline>?, string?)

Makes a request to the API endpoint for a static map image.

```
public Task<byte[]?> GetMapBytesAsync(Map mapSettings, IEnumerable<Marker>? markers = null, IEnumerable<Polyline>? paths = null, string? identifier = null)
```

## Parameters

### mapSettings [Map](#)

A [Map](#) element that defines key features about the map.>

**markers** [IEnumerable<Marker>](#)

Optional. [Markers](#) to be included in the map.

**paths** [IEnumerable<Polyline>](#)

Optional. [Polylines](#) to be included in the map.

**identifier** [string](#)

A string appended to the request object for tracking or identification.

Returns

[Task<byte\[\]>](#)

An [IResponse<TResponse>](#) containing the raw byte[] response from the Google API.

# Namespace GoogleMapsWrapper.Containers

## Classes

### [ElevationContainer](#)

An object to hold parsed JSON data from Elevation API responses.

### [GeocodeContainer](#)

A container to hold parsed JSON information from a [GeocodeApi](#) response.

## Interfaces

### [.IContainer](#)

Describes a container object used to hold parsed data from an API response.

# Class ElevationContainer

Namespace: [GoogleMapsWrapper.Containers](#)

Assembly: GoogleMapsWrapper.dll

An object to hold parsed JSON data from Elevation API responses.

```
public record ElevationContainer :.IContainer, IEquatable<ElevationContainer>
```

## Inheritance

[object](#) ← ElevationContainer

## Implements

[.IContainer](#), [IEquatable](#)<[ElevationContainer](#)>

## Inherited Members

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

## Constructors

### ElevationContainer(double, double, double, GpsCoordinate?, object?)

An object to hold parsed JSON data from Elevation API responses.

```
public ElevationContainer(double ElevationMeters, double Resolution, double ElevationFeet,  
GpsCoordinate? Coordinates, object? AssociatedData)
```

## Parameters

**ElevationMeters** [double](#)

The elevation in meters.

**Resolution** [double](#)

Indicates the maximum distance between data points from which the elevation was interpolated, in meters. This property is acquired from the JSON response's 'administrative\_area\_level\_1' property.

## **ElevationFeet** [double](#)

The elevation in feet. This property is mathematically converted and not intrinsic to the API.

## **Coordinates** [GpsCoordinate?](#)

The GPS coordinate for the request.

## **AssociatedData** [object](#)

Associated information about the data or it's source. This contains the IResponse.

# Properties

## AssociatedData

Associated information about the data or it's source. This contains the IResponse.

```
public object? AssociatedData { get; init; }
```

## Property Value

[object](#)

## Coordinates

The GPS coordinate for the request.

```
public GpsCoordinate? Coordinates { get; init; }
```

## Property Value

[GpsCoordinate?](#)

## ElevationFeet

The elevation in feet. This property is mathematically converted and not intrinsic to the API.

```
public double ElevationFeet { get; init; }
```

Property Value

[double](#) ↗

## ElevationMeters

The elevation in meters.

```
public double ElevationMeters { get; init; }
```

Property Value

[double](#) ↗

## Resolution

Indicates the maximum distance between data points from which the elevation was interpolated, in meters. This property is acquired from the JSON response's 'administrative\_area\_level\_1' property.

```
public double Resolution { get; init; }
```

Property Value

[double](#) ↗

# Class GeocodeContainer

Namespace: [GoogleMapsWrapper.Containers](#)

Assembly: GoogleMapsWrapper.dll

A container to hold parsed JSON information from a [GeocodeApi](#) response.

```
public record GeocodeContainer :.IContainer, IEquatable<GeocodeContainer>
```

## Inheritance

[object](#) ← GeocodeContainer

## Implements

[.IContainer](#), [IEquatable](#)<[GeocodeContainer](#)>

## Inherited Members

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

## Constructors

GeocodeContainer(string, string, string, string, string, int, string,  
GpsCoordinate?, object?)

A container to hold parsed JSON information from a [GeocodeApi](#) response.

```
public GeocodeContainer(string Country, string State, string County, string City,  
string Municipality, int ZipCode, string Address, GpsCoordinate? Coordinates,  
object? AssociatedData)
```

## Parameters

Country [string](#)

This property is acquired from the JSON response's 'country' property.

State [string](#)

This property is acquired from the JSON response's 'administrative\_area\_level\_1' property.

## County [string](#)

This property is acquired from the JSON response's 'administrative\_area\_level\_2' property.

## City [string](#)

This property is acquired from the JSON response's 'locality' property.

## Municipality [string](#)

This property is acquired from the JSON response's 'administrative\_area\_level\_3' property.

## ZipCode [int](#)

This property is acquired from the JSON response's 'postal\_code' property.

## Address [string](#)

This property is acquired from the JSON response's 'formatted\_address' property.

## Coordinates [GpsCoordinate](#)?

The GPS coordinate for the request.

## AssociatedData [object](#)

Associated information about the data or it's source. This contains the IResponse.

# Properties

## Address

This property is acquired from the JSON response's 'formatted\_address' property.

```
public string Address { get; init; }
```

## Property Value

### [string](#)

## AssociatedData

Associated information about the data or it's source. This contains the IResponse.

```
public object? AssociatedData { get; init; }
```

Property Value

[object](#)

## City

This property is acquired from the JSON response's 'locality' property.

```
public string City { get; init; }
```

Property Value

[string](#)

## Coordinates

The GPS coordinate for the request.

```
public GpsCoordinate? Coordinates { get; init; }
```

Property Value

[GpsCoordinate?](#)

## Country

This property is acquired from the JSON response's 'country' property.

```
public string Country { get; init; }
```

Property Value

[string](#)

## County

This property is acquired from the JSON response's 'administrative\_area\_level\_2' property.

```
public string County { get; init; }
```

### Property Value

[string](#)

## Municipality

This property is acquired from the JSON response's 'administrative\_area\_level\_3' property.

```
public string Municipality { get; init; }
```

### Property Value

[string](#)

## State

This property is acquired from the JSON response's 'administrative\_area\_level\_1' property.

```
public string State { get; init; }
```

### Property Value

[string](#)

## ZipCode

This property is acquired from the JSON response's 'postal\_code' property.

```
public int ZipCode { get; init; }
```

Property Value

[int](#)

## Methods

### ToString()

Converts to string.

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

Returns

[string](#)

A [string](#) that represents this instance.

# Interface IContainer

Namespace: [GoogleMapsWrapper.Containers](#)

Assembly: GoogleMapsWrapper.dll

Describes a container object used to hold parsed data from an API response.

```
public interface IContainer
```

## Properties

### AssociatedData

A property to append data to the container.

```
object? AssociatedData { get; }
```

Property Value

[object](#)

# Namespace GoogleMapsWrapper.Elements

## Classes

### [GoogleMapElement](#)

An abstract base class that contains properties from which map elements derive. A GoogleMapElement can be defined as a geo-locational data type, such as a map, marker (placemark), or polyline (path) that is leveraged by the wrapper.

### [Map](#)

Defines a 'map' object and it's characteristics. It is primarily used as a parameter for methods within the StaticMapsApi.

### [Marker](#)

Defines a marker (placemark) object and it's characteristics. It is primarily used as parameter to methods within the StaticMapsApi.

### [Polyline](#)

# Class GoogleMapElement

Namespace: [GoogleMapsWrapper.Elements](#)

Assembly: GoogleMapsWrapper.dll

An abstract base class that contains properties from which map elements derive. A GoogleMapElement can be defined as a geo-locational data type, such as a map, marker (placemark), or polyline (path) that is leveraged by the wrapper.

```
public abstract record GoogleMapElement : IEquatable<GoogleMapElement>
```

**Inheritance**

[object](#) ← GoogleMapElement

**Implements**

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

**Derived**

[Map](#), [Marker](#), [Polyline](#)

**Inherited Members**

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

## Properties

### AssociatedData

```
public object? AssociatedData { get; init; }
```

Property Value

[object](#)

### Color

```
public Color Color { get; init; }
```

Property Value

[Color](#) ↗

## Height

```
public int Height { get; init; }
```

Property Value

[int](#) ↗

## Id

```
public string? Id { get; init; }
```

Property Value

[string](#) ↗

## Name

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

Property Value

[string](#) ↗

## SecondaryColor

```
public Color SecondaryColor { get; init; }
```

Property Value

[Color](#) ↗

## Visible

```
public bool Visible { get; init; }
```

Property Value

[bool](#) ↗

## Width

```
public int Width { get; init; }
```

Property Value

[int](#) ↗

# Class Map

Namespace: [GoogleMapsWrapper.Elements](#)

Assembly: GoogleMapsWrapper.dll

Defines a 'map' object and it's characteristics. It is primarily used as a parameter for methods within the StaticMapsApi.

```
public record Map : GoogleMapElement, IEquatable<GoogleMapElement>, IEquatable<Map>
```

## Inheritance

[object](#) ← [GoogleMapElement](#) ← Map

## Implements

[IEquatable](#)<[GoogleMapElement](#)>, [IEquatable](#)<[Map](#)>

## Inherited Members

[GoogleMapElement.Name](#), [GoogleMapElement.Id](#), [GoogleMapElement.AssociatedData](#),  
[GoogleMapElement.Color](#), [GoogleMapElement.SecondaryColor](#), [GoogleMapElement.Visible](#),  
[GoogleMapElement.Height](#), [GoogleMapElement.Width](#), [object.Equals\(object\)](#),  
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),  
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

# Constructors

## Map()

Constructs a new instance of a map.

```
public Map()
```

# Properties

## Center

Sets the center of the map when rendered on the StaticMapsAPI.

```
public GpsCoordinate? Center { get; init; }
```

Property Value

[GpsCoordinate?](#)

## Dimensions

A string describing the size dimensions of the map. Format is: {Height}x{Width}

```
public string Dimensions { get; init; }
```

Property Value

[string](#)

## ImageFormat

Desired file format for an image of the map.///

```
public MapImageFormats ImageFormat { get; init; }
```

Property Value

[MapImageFormats](#)

## MapType

The type or style of map to be rendered by the API.///

```
public MapTypes MapType { get; init; }
```

Property Value

[MapTypes](#)

## Scale

Scale type of the map.///

```
public MapScaleTypes Scale { get; init; }
```

Property Value

[MapScaleTypes](#)

## Zoom

A zoom value for which the map should be displayed. Accepts values: 0 to 21, where zero (default) is considered 'omitted'. If omitted from a StaticMapsApi request, the Google API determines a zoom that best fits the map.

```
public int Zoom { get; init; }
```

Property Value

[int](#) ↗

# Class Marker

Namespace: [GoogleMapsWrapper.Elements](#)

Assembly: GoogleMapsWrapper.dll

Defines a marker (placemark) object and it's characteristics. It is primarily used as parameter to methods within the StaticMapsApi.

```
public record Marker : GoogleMapElement, IEquatable<GoogleMapElement>, IEquatable<Marker>
```

## Inheritance

[object](#) ← [GoogleMapElement](#) ← Marker

## Implements

[IEquatable](#)<[GoogleMapElement](#)>, [IEquatable](#)<[Marker](#)>

## Inherited Members

[GoogleMapElement.Name](#), [GoogleMapElement.Id](#), [GoogleMapElement.AssociatedData](#),  
[GoogleMapElement.Color](#), [GoogleMapElement.SecondaryColor](#), [GoogleMapElement.Visible](#),  
[GoogleMapElement.Height](#), [GoogleMapElement.Width](#), [object.Equals\(object\)](#),  
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),  
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

# Constructors

## Marker(GpsCoordinate)

Constructs a new marker instance.

```
public Marker(GpsCoordinate Coordinate)
```

## Parameters

Coordinate [GpsCoordinate](#)

# Properties

## Coordinate

The location of the marker.

```
public GpsCoordinate Coordinate { get; init; }
```

### Property Value

[GpsCoordinate](#)

## CustomIcon

A custom icon to represent the marker.

```
public StaticMapCustomIcon? CustomIcon { get; init; }
```

### Property Value

[StaticMapCustomIcon](#)

## Label

A label applied to the marker.

```
public char Label { get; init; }
```

### Property Value

[char](#)

### Remarks

Accepts values that are alpha-numerical, uppercase only. Single character.

## Scale

Affects the scale of the marker graphic rendered on a static map.

```
public MarkerScaleTypes Scale { get; init; }
```

Property Value

[MarkerScaleTypes](#)

## Size

Affects the scale of the marker graphic rendered on a static map.

```
public MarkerSizes Size { get; init; }
```

Property Value

[MarkerSizes](#)

# Class Polyline

Namespace: [GoogleMapsWrapper.Elements](#)

Assembly: GoogleMapsWrapper.dll

```
public record Polyline : GoogleMapElement, IEquatable<GoogleMapElement>,  
IEquatable<Polyline>
```

## Inheritance

[object](#) ← [GoogleMapElement](#) ← Polyline

## Implements

[IEquatable](#)<[GoogleMapElement](#)>, [IEquatable](#)<[Polyline](#)>

## Inherited Members

[GoogleMapElement.Name](#), [GoogleMapElement.Id](#), [GoogleMapElement.AssociatedData](#),  
[GoogleMapElement.Color](#), [GoogleMapElement.SecondaryColor](#), [GoogleMapElement.Visible](#),  
[GoogleMapElement.Height](#), [GoogleMapElement.Width](#), [object.Equals\(object\)](#),  
[object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),  
[object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#)

# Constructors

## Polyline(IEnumerable<GpsCoordinate>)

Initializes a new Polyline from a set of coordinate objects.

```
public Polyline(IEnumerable<GpsCoordinate> Coordinates)
```

## Parameters

Coordinates [IEnumerable](#)<[GpsCoordinate](#)>

# Properties

## Coordinates

Gets the coordinates contained in the Polyline.

```
public IReadOnlyCollection<GpsCoordinate> Coordinates { get; }
```

Property Value

[IReadOnlyCollection](#)<[GpsCoordinate](#)>

## Geodesic

Allows the line to follow the earth's contour.

```
public bool Geodesic { get; init; }
```

Property Value

[bool](#)

## Label

Label displayed with the marker.

```
public char? Label { get; init; }
```

Property Value

[char](#)?

## Weight

The weight of the line when rendered. In pixels.

```
public int Weight { get; init; }
```

Property Value

## Methods

### IsEmpty()

Returns true if the Polyline contains zero coordinates.

```
public bool IsEmpty()
```

Returns

[bool ↗](#)

### IsValid()

Returns true if the Polyline contains more than 1 coordinate.

```
public bool IsValid()
```

Returns

[bool ↗](#)

### ToString()

Returns a semicolon separated string of coordinates.

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

Returns

[string ↗](#)

# Namespace GoogleMapsWrapper.Engine

## Classes

### [ApiEngine](#)

An implementation of [IApiEngine](#) to send [IRequest](#)s and return data in form of [IResponse<TResponse>](#)s.

### [FlurlExtensions](#)

## Interfaces

### [IApiEngine](#)

# Class ApiEngine

Namespace: [GoogleMapsWrapper.Engine](#)

Assembly: GoogleMapsWrapper.dll

An implementation of [IApiEngine](#) to send [IRequests](#) and return data in form of [IResponse<TResponse>](#)s.

```
public class ApiEngine : IApiEngine
```

Inheritance

[object](#) ← ApiEngine

Implements

[IApiEngine](#)

Inherited Members

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

## Remarks

This layer of the application operates closest to the endpoint by using [HttpClient](#) to send HTTPGet requests.

## Constructors

### ApiEngine(HttpClient, IConfiguration)

```
public ApiEngine(HttpClient httpClient, IConfiguration config)
```

Parameters

httpClient [HttpClient](#)

config  [IConfiguration](#)

## Properties

## BaseUrl

```
public string BaseUrl { get; }
```

### Property Value

[string](#)

## Methods

### GetBytesAsync(IRequest)

Make a GET request to the API endpoint and return a byte array response.

```
public Task<IResponse<byte[]>> GetBytesAsync(IRequest request)
```

#### Parameters

request  [IRequest](#)

#### Returns

[Task](#)<[IResponse](#)<[byte](#)>[]>

### GetJsonAsync(IRequest)

Make a GET request to the API endpoint and return a JSON response.

```
public Task<IResponse<JsonDocument>> GetJsonAsync(IRequest request)
```

#### Parameters

request  [IRequest](#)

#### Returns

[Task](#)<[IResponse](#)<[JsonDocument](#)>>



# Class FlurlExtensions

Namespace: [GoogleMapsWrapper.Engine](#)

Assembly: GoogleMapsWrapper.dll

```
public static class FlurlExtensions
```

## Inheritance

[object](#) ← FlurlExtensions

## Inherited Members

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

## Methods

### SetQueryParamWithDefault(Url, string, object?, object?)

An extension method to flurl.url to SetQueryParam with a default value. If the value passed is null or an empty string, the default value is used. If the default value is null, Flurl.NullValueHandling.Remove is enacted.

```
public static Url SetQueryParamWithDefault(this Url url, string name, object? value,  
object? defaultValue)
```

## Parameters

url Url

name [string](#)

value [object](#)

defaultValue [object](#)

## Returns

Url

# Interface IApiEngine

Namespace: [GoogleMapsWrapper.Engine](#)

Assembly: GoogleMapsWrapper.dll

```
public interface IApiEngine
```

## Properties

### BaseUrl

```
string BaseUrl { get; }
```

### Property Value

[string](#)

## Methods

### GetBytesAsync(IRequest)

```
Task<IResponse<byte[]>> GetBytesAsync(IRequest request)
```

### Parameters

**request** [IRequest](#)

### Returns

[Task](#)<[IResponse](#)<[byte](#)[]>>

### GetJsonAsync(IRequest)

```
Task<IResponse<JsonDocument>> GetJsonAsync(IRequest request)
```

## Parameters

request  [IRequest](#)

## Returns

[Task](#) < [IResponse](#)< [JsonDocument](#)>>

# Namespace GoogleMapsWrapper.Exceptions

## Classes

[GoogleMapsApiException](#)

[GoogleMapsJavascriptException](#)

# Class GoogleMapsApiException

Namespace: [GoogleMapsWrapper.Exceptions](#)

Assembly: GoogleMapsWrapper.dll

```
public class GoogleMapsApiException : Exception, ISerializable
```

## Inheritance

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

## 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

### GoogleMapsApiException()

```
public GoogleMapsApiException()
```

### GoogleMapsApiException(string)

```
public GoogleMapsApiException(string message)
```

## Parameters

message [string](#)

## GoogleMapsApiException(string, Exception)

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

### Parameters

message [string](#)

innerException [Exception](#)

# Class GoogleMapsJavascriptException

Namespace: [GoogleMapsWrapper.Exceptions](#)

Assembly: GoogleMapsWrapper.dll

```
public class GoogleMapsJavascriptException : Exception, ISerializable
```

## Inheritance

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

## 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

### GoogleMapsJavascriptException()

```
public GoogleMapsJavascriptException()
```

### GoogleMapsJavascriptException(string)

```
public GoogleMapsJavascriptException(string message)
```

## Parameters

message [string](#)

## GoogleMapsJavascriptException(string, Exception)

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

### Parameters

message [string](#)

innerException [Exception](#)

# Namespace GoogleMapsWrapper.JavascriptApi

## Classes

[BrowserErrorEventArgs](#)

[MarkerMouseEventArgs](#)

## Interfaces

[IMapBoundElement](#)

# Class BrowserErrorEventArgs

Namespace: [GoogleMapsWrapper.JavascriptApi](#)

Assembly: GoogleMapsWrapper.dll

```
public class BrowserErrorEventArgs : EventArgs
```

## Inheritance

[object](#) ← [EventArgs](#) ← BrowserErrorEventArgs

## Inherited Members

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

## Constructors

### BrowserErrorEventArgs(string)

```
public BrowserErrorEventArgs(string message)
```

## Parameters

message [string](#)

## Properties

### Message

```
public string Message { get; }
```

## Property Value

[string](#)

# Interface IMapBoundElement

Namespace: [GoogleMapsWrapper.JavascriptApi](#)

Assembly: GoogleMapsWrapper.dll

```
public interface IMapBoundElement
```

## Properties

### Id

```
string? Id { get; }
```

Property Value

[string](#) ↗

## Methods

### Serialize()

```
string Serialize()
```

Returns

[string](#) ↗

# Class MarkerMouseEventArgs

Namespace: [GoogleMapsWrapper.JavascriptApi](#)

Assembly: GoogleMapsWrapper.dll

```
public class MarkerMouseEventArgs : EventArgs
```

## Inheritance

[object](#) ← [EventArgs](#) ← MarkerMouseEventArgs

## Inherited Members

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

## Constructors

### MarkerMouseEventArgs(string)

```
public MarkerMouseEventArgs(string id)
```

## Parameters

**id** [string](#)

## Properties

### Id

```
public string Id { get; }
```

## Property Value

[string](#)

# Namespace GoogleMapsWrapper.Javascript

## Api.Browser

### Classes

[GoogleMapsBrowser](#)

[GoogleMaps\\_READONLYRepository](#)

[GoogleMapsRepository](#)

[ScriptResult](#)

### Interfaces

[IBrowser](#)

Defines the minimum requirements needed for a browser-like object to interact with an IGoogleMapsBrowser.

[IGoogleMapsBrowser](#)

[IGoogleMapsJs\\_READONLYRepository](#)

A repository for storing map state. Data is loaded into this object by IGoogleMapsJsRepository

# Class GoogleMapsBrowser

Namespace: [GoogleMapsWrapper.JavascriptApi.Browser](#)

Assembly: GoogleMapsWrapper.dll

```
[ClassInterface(ClassInterfaceType.AutoDual)]
[ComVisible(true)]
public class GoogleMapsBrowser : IGoogleMapsBrowser, IGoogleMapsJsSendable,
IGoogleMapsJsListenable
```

## Inheritance

[object](#) ← GoogleMapsBrowser

## Implements

[IGoogleMapsBrowser](#), [IGoogleMapsJsSendable](#), [IGoogleMapsJsListenable](#)

## Inherited Members

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

## Constructors

### GoogleMapsBrowser(IBrowser)

```
public GoogleMapsBrowser(IBrowser browser)
```

## Parameters

browser [IBrowser](#)

## Properties

### Browser

```
public IBrowser Browser { get; }
```

Property Value

[IBrowser](#)

## Repository

```
public IGoogleMapsJs_READONLYRepository Repository { get; }
```

Property Value

[IGoogleMapsJs\\_READONLYRepository](#)

## Methods

### AddMarkerAsync(BoundMarker)

```
public Task AddMarkerAsync(BoundMarker boundMarker)
```

Parameters

boundMarker [BoundMarker](#)

Returns

[Task](#)

### Close()

```
public void Close()
```

### LoadAsync()

```
public Task LoadAsync()
```

Returns

[Task ↗](#)

## Navigate(GoogleMapsHtmlTemplate)

```
public void Navigate(GoogleMapsHtmlTemplate template)
```

Parameters

template [GoogleMapsHtmlTemplate](#)

## RemoveMarkerAsync(BoundMarker)

```
public Task RemoveMarkerAsync(BoundMarker boundMarker)
```

Parameters

boundMarker [BoundMarker](#)

Returns

[Task ↗](#)

## UpdateMarkerAsync(BoundMarker)

```
public Task UpdateMarkerAsync(BoundMarker boundMarker)
```

Parameters

boundMarker [BoundMarker](#)

Returns

[Task ↗](#)

## \_OnError(string)

```
public void _OnError(string message)
```

Parameters

message [string](#)

## \_OnMapClick(string)

```
public void _OnMapClick(string coordinate)
```

Parameters

coordinate [string](#)

## \_OnMapDblClick(string)

```
public void _OnMapDblClick(string coordinate)
```

Parameters

coordinate [string](#)

## \_OnMapRightClick(string)

```
public void _OnMapRightClick(string coordinate)
```

Parameters

coordinate [string](#)

## \_OnMarkerClick(string)

```
public void _OnMarkerClick(string id)
```

Parameters

**id** [string](#)

## \_OnMarkerDbClick(string)

```
public void _OnMarkerDbClick(string id)
```

Parameters

**id** [string](#)

## \_OnMarkerDrag(string, string)

```
public void _OnMarkerDrag(string id, string coordinate)
```

Parameters

**id** [string](#)

**coordinate** [string](#)

## \_OnMarkerMouseOut(string)

```
public void _OnMarkerMouseOut(string id)
```

Parameters

**id** [string](#)

## \_OnMarkerMouseOver(string)

```
public void _OnMarkerMouseOver(string id)
```

Parameters

[id string](#)

## \_OnMarkerRightClick(string)

```
public void _OnMarkerRightClick(string id)
```

Parameters

[id string](#)

# Events

## OnMapClick

```
public event EventHandler<MapClickEventArgs>? OnMapClick
```

Event Type

[EventHandler](#) <[MapClickEventArgs](#)>

## OnMapDblClick

```
public event EventHandler<MapClickEventArgs>? OnMapDblClick
```

Event Type

[EventHandler](#) <[MapClickEventArgs](#)>

## OnMapRightClick

```
public event EventHandler<MapClickEventArgs>? OnMapRightClick
```

Event Type

[EventHandler](#) <[MapClickEventArgs](#)>

## OnMarkerClick

```
public event EventHandler<MarkerClickEventArgs>? OnMarkerClick
```

Event Type

[EventHandler](#) <[MarkerClickEventArgs](#)>

## OnMarkerDblClick

```
public event EventHandler<MarkerClickEventArgs>? OnMarkerDblClick
```

Event Type

[EventHandler](#) <[MarkerClickEventArgs](#)>

## OnMarkerDrag

```
public event EventHandler<MarkerDragEventArgs>? OnMarkerDrag
```

Event Type

[EventHandler](#) <[MarkerDragEventArgs](#)>

## OnMarkerMouseOut

```
public event EventHandler<MarkerMouseEventArgs>? OnMarkerMouseOut
```

Event Type

[EventHandler](#) <[MarkerMouseEventArgs](#)>

## OnMarkerMouseOver

```
public event EventHandler<MarkerMouseEventArgs>? OnMarkerMouseOver
```

Event Type

[EventHandler](#) <[MarkerMouseEventArgs](#)>

## OnMarkerRightClick

```
public event EventHandler<MarkerClickEventArgs>? OnMarkerRightClick
```

Event Type

[EventHandler](#) <[MarkerClickEventArgs](#)>

# Class GoogleMaps\_READONLYRepository

Namespace: [GoogleMapsWrapper.JavascriptApi.Browser](#)

Assembly: GoogleMapsWrapper.dll

```
public class GoogleMaps_READONLYRepository : IGoogleMapsJs_READONLYRepository
```

## Inheritance

[object](#) ← GoogleMaps\_READONLYRepository

## Implements

[IGoogleMapsJs\\_READONLYRepository](#)

## Inherited Members

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

## Constructors

GoogleMaps\_READONLYRepository(IReadOnlyList<BoundMarker>  
)

```
public GoogleMaps_READONLYRepository(IReadOnlyList<BoundMarker> markers)
```

## Parameters

markers [IReadOnlyList](#)<BoundMarker>

## Properties

### Markers

```
public IReadOnlyList<BoundMarker> Markers { get; }
```

### Property Value

[IReadOnlyList](#) <[BoundMarker](#)>

# Class GoogleMapsRepository

Namespace: [GoogleMapsWrapper.JavascriptApi.Browser](#)

Assembly: GoogleMapsWrapper.dll

```
public class GoogleMapsRepository
```

## Inheritance

[object](#) ← GoogleMapsRepository

## Inherited Members

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

## Constructors

### GoogleMapsRepository()

```
public GoogleMapsRepository()
```

## Properties

### Markers

```
public IReadOnlyList<BoundMarker> Markers { get; }
```

### Property Value

[IReadOnlyList](#)<[BoundMarker](#)>

## Source

```
public IGoogleMapsBrowser? Source { get; }
```

Property Value

[IGoogleMapsBrowser](#)

## Methods

### AddMarker(BoundMarker)

```
public void AddMarker(BoundMarker marker)
```

Parameters

marker [BoundMarker](#)

### AsReadonly()

```
public IGoogleMapsJs_READONLYRepository AsReadonly()
```

Returns

[IGoogleMapsJs\\_READONLYRepository](#)

### ContainsMarker(BoundMarker)

```
public bool ContainsMarker(BoundMarker marker)
```

Parameters

marker [BoundMarker](#)

Returns

[bool](#)

## RemoveMarker(BoundMarker)

```
public void RemoveMarker(BoundMarker marker)
```

### Parameters

marker [BoundMarker](#)

## UpdateMarker(BoundMarker)

```
public void UpdateMarker(BoundMarker marker)
```

### Parameters

marker [BoundMarker](#)

# Interface IBrowser

Namespace: [GoogleMapsWrapper.JavascriptApi.Browser](#)

Assembly: GoogleMapsWrapper.dll

Defines the minimum requirements needed for a browser-like object to interact with an IGoogleMapsBrowser.

```
public interface IBrowser
```

## Properties

### Html

```
string Html { get; }
```

Property Value

[string](#)

## Methods

### BindObject(string, object)

```
void BindObject(string name, object sourceObject)
```

Parameters

name [string](#)

sourceObject [object](#)

### Close()

```
void Close()
```

## ExecuteScriptAsync(string)

```
Task<ScriptResult> ExecuteScriptAsync(string script)
```

### Parameters

script [string](#)

### Returns

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

## ExecuteScriptAsyncInvoke(string)

```
Task<ScriptResult> ExecuteScriptAsyncInvoke(string script)
```

### Parameters

script [string](#)

### Returns

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

## Navigate(string)

```
void Navigate(string html)
```

### Parameters

html [string](#)

## Navigate(Uri)

`void Navigate(Uri source)`

### Parameters

`source Uri`

## SourceObject()

`object SourceObject()`

### Returns

`object`

# Interface IGoogleMapsBrowser

Namespace: [GoogleMapsWrapper.JavascriptApi.Browser](#)

Assembly: GoogleMapsWrapper.dll

```
public interface IGoogleMapsBrowser : IGoogleMapsJsSendable, IGoogleMapsJsListenable
```

## Inherited Members

[IGoogleMapsJsSendable.AddMarkerAsync\(BoundMarker\)](#),  
[IGoogleMapsJsSendable.RemoveMarkerAsync\(BoundMarker\)](#),  
[IGoogleMapsJsSendable.UpdateMarkerAsync\(BoundMarker\)](#), [IGoogleMapsJsListenable.OnMapClick](#),  
[IGoogleMapsJsListenable.OnMapDbClick](#), [IGoogleMapsJsListenable.OnMapRightClick](#),  
[IGoogleMapsJsListenable.OnMarkerClick](#), [IGoogleMapsJsListenable.OnMarkerDbClick](#),  
[IGoogleMapsJsListenable.OnMarkerRightClick](#), [IGoogleMapsJsListenable.OnMarkerDrag](#),  
[IGoogleMapsJsListenable.OnMarkerMouseOver](#), [IGoogleMapsJsListenable.OnMarkerMouseOut](#),  
[IGoogleMapsJsListenable.OnError\(string\)](#), [IGoogleMapsJsListenable.OnMapClick\(string\)](#),  
[IGoogleMapsJsListenable.OnMapDbClick\(string\)](#), [IGoogleMapsJsListenable.OnMapRightClick\(string\)](#),  
[IGoogleMapsJsListenable.OnMarkerClick\(string\)](#), [IGoogleMapsJsListenable.OnMarkerDbClick\(string\)](#),  
[IGoogleMapsJsListenable.OnMarkerRightClick\(string\)](#),  
[IGoogleMapsJsListenable.OnMarkerDrag\(string, string\)](#),  
[IGoogleMapsJsListenable.OnMarkerMouseOver\(string\)](#),  
[IGoogleMapsJsListenable.OnMarkerMouseOut\(string\)](#).

## Properties

### Browser

```
IBrowser Browser { get; }
```

### Property Value

[IBrowser](#)

### Repository

```
IGoogleMapsJs_READONLYRepository Repository { get; }
```

Property Value

[IGoogleMapsJs\\_READONLYRepository](#)

## Methods

Close()

```
void Close()
```

Navigate(GoogleMapsHtmlTemplate)

```
void Navigate(GoogleMapsHtmlTemplate template)
```

Parameters

template [GoogleMapsHtmlTemplate](#)

# Interface IGoogleMapsJsReadonlyRepository

Namespace: [GoogleMapsWrapper.JavascriptApi.Browser](#)

Assembly: GoogleMapsWrapper.dll

A repository for storing map state. Data is loaded into this object by IGoogleMapsJsRepository

```
public interface IGoogleMapsJsReadonlyRepository
```

## Properties

### Markers

IReadOnlyList<BoundMarker> Markers { [get](#); }

### Property Value

[IReadOnlyList](#) <[BoundMarker](#)>

# Class ScriptResult

Namespace: [GoogleMapsWrapper.JavascriptApi.Browser](#)

Assembly: GoogleMapsWrapper.dll

```
public record ScriptResult : IEquatable<ScriptResult>
```

## Inheritance

[object](#) ← ScriptResult

## Implements

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

## Inherited Members

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

## Constructors

### ScriptResult(string?, bool, string)

```
public ScriptResult(string? Result, bool IsSuccess, string ExceptionMessage)
```

## Parameters

Result [string](#)

IsSuccess [bool](#)

ExceptionMessage [string](#)

## Properties

### ExceptionMessage

```
public string ExceptionMessage { get; init; }
```

Property Value

[string](#) ↗

IsSuccess

```
public bool IsSuccess { get; init; }
```

Property Value

[bool](#) ↗

Result

```
public string? Result { get; init; }
```

Property Value

[string](#) ↗

Methods

TryDeserialize<T>(out T?)

```
public bool TryDeserialize<T>(out T? obj)
```

Parameters

**obj** T

Returns

[bool](#) ↗

Type Parameters



# Namespace GoogleMapsWrapper.Javascript

## Api.Elements

### Classes

[BoundMarker](#)

[BoundMarkerJsonConverter](#)

[LockSafeHashSet<T>](#)

A HashSet that is modified within a thread lock mechanism.

[MapBoundElementEqualityComparer](#)

An equality comparer for IMapBoundElement type, to check if two IMapBoundElements have equal ID values.

[MapLabel](#)

[MapSvgIcon](#)

# Class BoundMarker

Namespace: [GoogleMapsWrapper.JavascriptApi.Elements](#)

Assembly: GoogleMapsWrapper.dll

```
public record BoundMarker : IMapBoundElement, IEquatable<BoundMarker>
```

## Inheritance

[object](#) ← BoundMarker

## Implements

[IMapBoundElement](#), [IEquatable](#)<BoundMarker>

## Inherited Members

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

## Constructors

BoundMarker(GpsCoordinate?, MapLabel, string?, bool)

```
public BoundMarker(GpsCoordinate? coordinates, MapLabel label, string? info = null, bool  
draggable = false)
```

## Parameters

coordinates [GpsCoordinate](#)?

label [MapLabel](#)

info [string](#)

draggable [bool](#)

BoundMarker(GpsCoordinate?, MapSvgIcon, MapLabel, string?, bool)

Overload constructor, Id property is produced internally as a GUID.

```
public BoundMarker(GpsCoordinate? coordinates, MapSvgIcon icon, MapLabel label, string? info  
= null, bool draggable = false)
```

## Parameters

coordinates [GpsCoordinate?](#)

icon [MapSvgIcon](#)

label [MapLabel](#)

info [string](#)

draggable [bool](#)

BoundMarker(string?, GpsCoordinate?, MapSvgIcon, MapLabel,  
string?, bool)

```
public BoundMarker(string? Id, GpsCoordinate? Coordinates, MapSvgIcon Icon, MapLabel Label,  
string? Info = null, bool Draggable = false)
```

## Parameters

Id [string](#)

Coordinates [GpsCoordinate?](#)

Icon [MapSvgIcon](#)

Label [MapLabel](#)

Info [string](#)

Draggable [bool](#)

# Properties

## Coordinates

```
public GpsCoordinate? Coordinates { get; init; }
```

### Property Value

[GpsCoordinate?](#)

## Draggable

```
public bool Draggable { get; init; }
```

### Property Value

[bool](#) ↗

## Icon

```
public MapSvgIcon Icon { get; init; }
```

### Property Value

[MapSvgIcon](#)

## Id

```
public string? Id { get; init; }
```

### Property Value

[string](#) ↗

## Info

```
public string? Info { get; init; }
```

Property Value

[string](#)

Label

```
public MapLabel Label { get; init; }
```

Property Value

[MapLabel](#)

## Methods

CopyAssignNewId(BoundMarker)

```
public static BoundMarker CopyAssignNewId(BoundMarker source)
```

Parameters

source [BoundMarker](#)

Returns

[BoundMarker](#)

CopyAssignNewId(string?, BoundMarker)

```
public static BoundMarker CopyAssignNewId(string? id, BoundMarker source)
```

Parameters

`id` [string](#)

`source` [BoundMarker](#)

Returns

[BoundMarker](#)

## FromJson(string)

```
public static BoundMarker FromJson(string json)
```

Parameters

`json` [string](#)

Returns

[BoundMarker](#)

## Serialize()

```
public string Serialize()
```

Returns

[string](#)

# Class BoundMarkerJsonConverter

Namespace: [GoogleMapsWrapper.JavascriptApi.Elements](#)

Assembly: GoogleMapsWrapper.dll

```
public class BoundMarkerJsonConverter : JsonConverter<BoundMarker>
```

## Inheritance

[object](#) ← [JsonConverter](#) ← [JsonConverter](#)<[BoundMarker](#)> ← BoundMarkerJsonConverter

## Inherited Members

[JsonConverter](#)<[BoundMarker](#)>.CanConvert([Type](#))[,](#)  
[JsonConverter](#)<[BoundMarker](#)>.ReadAsPropertyName([ref](#) [Utf8JsonReader](#), [Type](#), [JsonSerializerOptions](#))[,](#)  
[JsonConverter](#)<[BoundMarker](#)>.WriteAsPropertyName([Utf8JsonWriter](#), [BoundMarker](#),  
[JsonSerializerOptions](#))[,](#)  
[JsonConverter](#)<[BoundMarker](#)>.HandleNull[,](#) [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 to type [BoundMarker](#).

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

## Parameters

[reader](#) [Utf8JsonReader](#)[,](#)

The reader.

[typeToConvert](#) [Type](#)[,](#)

The type to convert.

[options](#) [JsonSerializerOptions](#)[,](#)

An object that specifies serialization options to use.

Returns

[BoundMarker](#)

The converted value.

## Write(Utf8JsonWriter, BoundMarker, JsonSerializerOptions)

Writes a specified value as JSON.

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

Parameters

[writer](#) [Utf8JsonWriter](#)

The writer to write to.

[value](#) [BoundMarker](#)

The value to convert to JSON.

[options](#) [JsonSerializerOptions](#)

An object that specifies serialization options to use.

# Class LockSafeHashSet<T>

Namespace: [GoogleMapsWrapper.JavascriptApi.Elements](#)

Assembly: GoogleMapsWrapper.dll

A HashSet that is modified within a thread lock mechanism.

```
public class LockSafeHashSet<T>
```

## Type Parameters

T

The type of object to store in the hash set. Null ref types are rejected during Add method. Note: the type should have an IEqualityComparerT for this hashset to determine equality. The equality comparer used for a given type can be checked with: EqualityComparerT.Default

## Inheritance

[object](#) ← LockSafeHashSet<T>

## Inherited Members

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

## Constructors

### LockSafeHashSet(IEqualityComparer<T>, bool)

```
public LockSafeHashSet(IEqualityComparer<T> comparer, bool ignoreExistsException = false)
```

## Parameters

comparer [IEqualityComparer](#)<T>

ignoreExistsException [bool](#)

## Properties

## Items

```
public IReadOnlyList<T> Items { get; }
```

### Property Value

[IReadOnlyList](#)<T>

## Methods

### Add(T, bool)

Add item.

```
public void Add(T item, bool throwIfExists = true)
```

### Parameters

item T

throwIfExists [bool](#)

Throw exception if item exists. Else ignore quietly.

### Exceptions

[ArgumentNullException](#)

[InvalidOperationException](#)

### AddOrReplace(T)

Add or replace item.

```
public void AddOrReplace(T item)
```

### Parameters

`item` T

## Clear()

```
public void Clear()
```

## Contains(T)

```
public bool Contains(T item)
```

### Parameters

`item` T

### Returns

[bool](#)

## Remove(T)

Removes item from set.

```
public void Remove(T item)
```

### Parameters

`item` T

Ignored if doesn't exist.

### Exceptions

[ArgumentNullException](#)

## Replace(T)

Replace an item in the set with a new item. Existing item is determined by the comparer.

```
public void Replace(T item)
```

### Parameters

**item** T

# Class MapBoundElementEqualityComparer

Namespace: [GoogleMapsWrapper.JavascriptApi.Elements](#)

Assembly: GoogleMapsWrapper.dll

An equality comparer for IMapBoundElement type, to check if two IMapBoundElements are have equal ID values.

```
public class MapBoundElementEqualityComparer : IEqualityComparer<IMapBoundElement>
```

## Inheritance

[object](#) ← MapBoundElementEqualityComparer

## Implements

[IEqualityComparer](#) < [IMapBoundElement](#) >

## Inherited Members

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

## Methods

### Equals(IMapBoundElement?, IMapBoundElement?)

Determines whether the specified objects are equal.

```
public bool Equals(IMapBoundElement? x, IMapBoundElement? y)
```

#### Parameters

x [IMapBoundElement](#)

The first object of type T to compare.

y [IMapBoundElement](#)

The second object of type T to compare.

#### Returns

[bool](#)

[true](#) if the specified objects are equal; otherwise, [false](#).

## GetHashCode(IMapBoundElement)

Returns a hash code for the specified object.

```
public int GetHashCode(IMapBoundElement obj)
```

### Parameters

[obj](#) [IMapBoundElement](#)

The [object](#) for which a hash code is to be returned.

### Returns

[int](#)

A hash code for the specified object.

### Exceptions

[ArgumentNullException](#)

The type of [obj](#) is a reference type and [obj](#) is [null](#).

# Class MapLabel

Namespace: [GoogleMapsWrapper.JavascriptApi.Elements](#)

Assembly: GoogleMapsWrapper.dll

```
public record MapLabel : IEquatable<MapLabel>
```

## Inheritance

[object](#) ← MapLabel

## Implements

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

## Inherited Members

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

## Constructors

### MapLabel(string)

```
public MapLabel(string Text)
```

#### Parameters

Text [string](#)

### MapLabel(string, Color, double, int, Color)

```
public MapLabel(string Text, Color Color, double Scale = 1, int FontWeight = 400, Color  
BackgroundColor = default)
```

#### Parameters

Text [string](#)

Color [Color](#)

Scale [double](#)

FontWeight [int](#)

BackgroundColor [Color](#)

## Properties

### BackgroundColor

```
public Color BackgroundColor { get; init; }
```

Property Value

[Color](#)

### Color

```
public Color Color { get; init; }
```

Property Value

[Color](#)

### FontWeight

```
public int FontWeight { get; init; }
```

Property Value

[int](#)

### Scale

```
public double Scale { get; init; }
```

Property Value

double ↗

Text

```
public string Text { get; init; }
```

Property Value

string ↗

## Methods

Serialize()

```
public string Serialize()
```

Returns

string ↗

# Class MapSvgIcon

Namespace: [GoogleMapsWrapper.JavascriptApi.Elements](#)

Assembly: GoogleMapsWrapper.dll

```
public record MapSvgIcon : IEquatable<MapSvgIcon>
```

## Inheritance

[object](#) ← MapSvgIcon

## Implements

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

## Inherited Members

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

## Constructors

### MapSvgIcon(string, Color, double, Color, double, double)

```
public MapSvgIcon(string SvgPath, Color Color, double Scale = 1, Color OutlineColor =  
default, double OutlineWeight = 1, double Opacity = 1)
```

## Parameters

SvgPath [string](#)

Color [Color](#)

Scale [double](#)

OutlineColor [Color](#)

OutlineWeight [double](#)

Opacity [double](#)

# Properties

## Color

```
public Color Color { get; init; }
```

### Property Value

[Color](#) ↗

## Opacity

```
public double Opacity { get; init; }
```

### Property Value

[double](#) ↗

## OutlineColor

```
public Color OutlineColor { get; init; }
```

### Property Value

[Color](#) ↗

## OutlineWeight

```
public double OutlineWeight { get; init; }
```

### Property Value

[double](#) ↗

## Scale

```
public double Scale { get; init; }
```

Property Value

[double](#) ↗

## SvgPath

```
public string SvgPath { get; init; }
```

Property Value

[string](#) ↗

## Methods

### CircleHollowIcon(Color, double)

```
public static MapSvgIcon CircleHollowIcon(Color color, double scale = 1)
```

Parameters

color [Color](#) ↗

scale [double](#) ↗

Returns

[MapSvgIcon](#)

### CircleIcon(Color, double)

```
public static MapSvgIcon CircleIcon(Color color, double scale = 1)
```

Parameters

**color** [Color](#)

**scale** [double](#)

Returns

[MapSvgIcon](#)

## PinIcon(Color, double)

```
public static MapSvgIcon PinIcon(Color color, double scale = 1)
```

Parameters

**color** [Color](#)

**scale** [double](#)

Returns

[MapSvgIcon](#)

## Serialize()

```
public string Serialize()
```

Returns

[string](#)

# Namespace GoogleMapsWrapper.Javascript Api.Html

## Classes

[GoogleMapsHtmlTemplate](#)

# Class GoogleMapsHtmlTemplate

Namespace: [GoogleMapsWrapper.JavascriptApi.Html](#)

Assembly: GoogleMapsWrapper.dll

```
public class GoogleMapsHtmlTemplate
```

## Inheritance

[object](#) ← GoogleMapsHtmlTemplate

## Inherited Members

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

## Constructors

GoogleMapsHtmlTemplate(string, string, string, Map?)

```
public GoogleMapsHtmlTemplate(string templateHtml, string apiKey, string  
objectBindingScript, Map? loadMap = null)
```

## Parameters

templateHtml [string](#)

apiKey [string](#)

objectBindingScript [string](#)

loadMap [Map](#)

## Properties

Html

```
public string Html { get; }
```

Property Value

[string](#) ↗

## TemplateHtml

```
public string TemplateHtml { get; }
```

Property Value

[string](#) ↗

# Namespace GoogleMapsWrapper.Javascript Api.Listener

## Classes

[MapClickEventArgs](#)

[MarkerClickEventArgs](#)

[MarkerDragEventArgs](#)

## Interfaces

[IGoogleMapsJsListenable](#)

# Interface IGoogleMapsJsListenable

Namespace: [GoogleMapsWrapper.JavascriptApi.Listener](#)

Assembly: GoogleMapsWrapper.dll

```
public interface IGoogleMapsJsListenable
```

## Methods

### \_OnError(string)

```
void _OnError(string message)
```

Parameters

message [string](#)

### \_OnMapClick(string)

```
void _OnMapClick(string coordinate)
```

Parameters

coordinate [string](#)

### \_OnMapDblClick(string)

```
void _OnMapDblClick(string coordinate)
```

Parameters

coordinate [string](#)

## \_OnMapRightClick(string)

```
void _OnMapRightClick(string coordinate)
```

Parameters

coordinate [string ↗](#)

## \_OnMarkerClick(string)

```
void _OnMarkerClick(string id)
```

Parameters

id [string ↗](#)

## \_OnMarkerDblClick(string)

```
void _OnMarkerDblClick(string id)
```

Parameters

id [string ↗](#)

## \_OnMarkerDrag(string, string)

```
void _OnMarkerDrag(string id, string coordinate)
```

Parameters

id [string ↗](#)

coordinate [string ↗](#)

## \_OnMarkerMouseOut(string)

```
void _OnMarkerMouseOut(string id)
```

Parameters

**id** [string](#)

## \_OnMarkerMouseOver(string)

```
void _OnMarkerMouseOver(string id)
```

Parameters

**id** [string](#)

## \_OnMarkerRightClick(string)

```
void _OnMarkerRightClick(string id)
```

Parameters

**id** [string](#)

# Events

## OnMapClick

**event** EventHandler<MapClickEventArgs>? OnMapClick

Event Type

[EventHandler](#)<[MapClickEventArgs](#)>

## OnMapDbClick

**event** EventHandler<MapClickEventArgs>? OnMapDbClick

Event Type

[EventHandler](#) <MapClickEventArgs>

## OnMapRightClick

**event** EventHandler<MapClickEventArgs>? OnMapRightClick

Event Type

[EventHandler](#) <MapClickEventArgs>

## OnMarkerClick

**event** EventHandler<MarkerClickEventArgs>? OnMarkerClick

Event Type

[EventHandler](#) <MarkerClickEventArgs>

## OnMarkerDbClick

**event** EventHandler<MarkerClickEventArgs>? OnMarkerDbClick

Event Type

[EventHandler](#) <MarkerClickEventArgs>

## OnMarkerDrag

```
event EventHandler<MarkerDragEventArgs>? OnMarkerDrag
```

Event Type

[EventHandler](#)<[MarkerDragEventArgs](#)>

## OnMarkerMouseOut

```
event EventHandler<MarkerMouseEventArgs>? OnMarkerMouseOut
```

Event Type

[EventHandler](#)<[MarkerMouseEventArgs](#)>

## OnMarkerMouseOver

```
event EventHandler<MarkerMouseEventArgs>? OnMarkerMouseOver
```

Event Type

[EventHandler](#)<[MarkerMouseEventArgs](#)>

## OnMarkerRightClick

```
event EventHandler<MarkerClickEventArgs>? OnMarkerRightClick
```

Event Type

[EventHandler](#)<[MarkerClickEventArgs](#)>

# Class MapClickEventArgs

Namespace: [GoogleMapsWrapper.JavascriptApi.Listener](#)

Assembly: GoogleMapsWrapper.dll

```
public class MapClickEventArgs : EventArgs
```

## Inheritance

[object](#) ← [EventArgs](#) ← MapClickEventArgs

## Inherited Members

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

## Constructors

### MapClickEventArgs(GpsCoordinate)

```
public MapClickEventArgs(GpsCoordinate coordinates)
```

## Parameters

coordinates [GpsCoordinate](#)

# Class MarkerClickEventArgs

Namespace: [GoogleMapsWrapper.JavascriptApi.Listener](#)

Assembly: GoogleMapsWrapper.dll

```
public class MarkerClickEventArgs : EventArgs
```

## Inheritance

[object](#) ← [EventArgs](#) ← MarkerClickEventArgs

## Inherited Members

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

## Constructors

### MarkerClickEventArgs(string)

```
public MarkerClickEventArgs(string id)
```

## Parameters

**id** [string](#)

## Properties

### Id

```
public string Id { get; }
```

## Property Value

[string](#)

# Class MarkerDragEventArgs

Namespace: [GoogleMapsWrapper.JavascriptApi.Listener](#)

Assembly: GoogleMapsWrapper.dll

```
public class MarkerDragEventArgs : EventArgs
```

## Inheritance

[object](#) ← [EventArgs](#) ← MarkerDragEventArgs

## Inherited Members

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

## Constructors

### MarkerDragEventArgs(string, GpsCoordinate)

```
public MarkerDragEventArgs(string id, GpsCoordinate coordinates)
```

## Parameters

**id** [string](#)

**coordinates** [GpsCoordinate](#)

## Properties

### Coordinates

```
public GpsCoordinate Coordinates { get; }
```

## Property Value

[GpsCoordinate](#)

Id

```
public string Id { get; }
```

Property Value

[string ↗](#)

# Namespace GoogleMapsWrapper.Javascript Api.Sender

## Interfaces

[IGoogleMapsJsSendable](#)

# Interface IGoogleMapsJsSendable

Namespace: [GoogleMapsWrapper.JavascriptApi.Sender](#)

Assembly: GoogleMapsWrapper.dll

```
public interface IGoogleMapsJsSendable
```

## Methods

### AddMarkerAsync(BoundMarker)

Task **AddMarkerAsync**(BoundMarker boundMarker)

Parameters

boundMarker [BoundMarker](#)

Returns

[Task](#)

### RemoveMarkerAsync(BoundMarker)

Task **RemoveMarkerAsync**(BoundMarker boundMarker)

Parameters

boundMarker [BoundMarker](#)

Returns

[Task](#)

## UpdateMarkerAsync(BoundMarker)

Task **UpdateMarkerAsync**(BoundMarker boundMarker)

Parameters

boundMarker [BoundMarker](#)

Returns

[Task](#)

# Namespace GoogleMapsWrapper.Parsers

## Classes

[ElevationParser](#)

[GeocodeParser](#)

## Interfaces

[IParser<TOutput, TInput>](#)

# Class ElevationParser

Namespace: [GoogleMapsWrapper.Parsers](#)

Assembly: GoogleMapsWrapper.dll

```
public class ElevationParser : IParser<ElevationContainer, JsonDocument>
```

## Inheritance

[object](#) ← ElevationParser

## Implements

[IParser<ElevationContainer, JsonDocument>](#)

## Inherited Members

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

## Methods

### Parse(JsonDocument)

```
public ElevationContainer Parse(JsonDocument input)
```

#### Parameters

[input](#) [JsonDocument](#)

#### Returns

[ElevationContainer](#)

### TryParse(JsonDocument, out ElevationContainer?)

```
public bool TryParse(JsonDocument input, out ElevationContainer? output)
```

## Parameters

input [JsonDocument](#) ↗

output [ElevationContainer](#)

## Returns

[bool](#) ↗

# Class GeocodeParser

Namespace: [GoogleMapsWrapper.Parsers](#)

Assembly: GoogleMapsWrapper.dll

```
public class GeocodeParser : IParser<GeocodeContainer, JsonDocument>
```

## Inheritance

[object](#) ← GeocodeParser

## Implements

[IParser<GeocodeContainer, JsonDocument>](#)

## Inherited Members

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

## Methods

### Parse(JsonDocument)

```
public GeocodeContainer Parse(JsonDocument input)
```

#### Parameters

**input** [JsonDocument](#)

#### Returns

[GeocodeContainer](#)

### TryParse(JsonDocument, out GeocodeContainer?)

```
public bool TryParse(JsonDocument input, out GeocodeContainer? output)
```

## Parameters

input [JsonDocument](#)

output [GeocodeContainer](#)

## Returns

[bool](#)

# Interface IParser<TOutput, TInput>

Namespace: [GoogleMapsWrapper.Parsers](#)

Assembly: GoogleMapsWrapper.dll

```
public interface IParser<TOutput, TInput>
```

## Type Parameters

TOutput

TInput

## Methods

### Parse(TInput)

```
TOutput Parse(TInput input)
```

#### Parameters

input TInput

#### Returns

TOutput

### TryParse(TInput, out TOutput?)

```
bool TryParse(TInput input, out TOutput? output)
```

#### Parameters

input TInput

**output** TOutput

Returns

[bool](#) ↗

# Namespace GoogleMapsWrapper.Requests

## Interfaces

[IRequest](#)

# Interface IRequest

Namespace: [GoogleMapsWrapper.Requests](#)

Assembly: GoogleMapsWrapper.dll

```
public interface IRequest
```

## Properties

### Api

```
ApiTypes Api { get; }
```

Property Value

[ApiTypes](#)

### Category

```
RequestTypes Category { get; }
```

Property Value

[RequestTypes](#)

### Id

```
string? Id { get; }
```

Property Value

[string](#) ↗

## Query

```
string Query { get; }
```

Property Value

[string](#) ↗

## Url

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

Property Value

[Uri](#) ↗

# Namespace GoogleMapsWrapper.Responses

## Classes

[ByteResponse](#)

[JsonResponse](#)

## Interfaces

[IResponse<TResponse>](#)

# Class ByteResponse

Namespace: [GoogleMapsWrapper.Responses](#)

Assembly: GoogleMapsWrapper.dll

```
public class ByteResponse : IResponse<byte[]>
```

## Inheritance

[object](#) ← ByteResponse

## Implements

[IResponse<byte\[\]>](#)

## Inherited Members

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

## Constructors

ByteResponse(IRequest, byte[], HttpResponseMessage)

```
public ByteResponse(IRequest sentRequest, byte[] content, HttpResponseMessage  
responseMessage)
```

## Parameters

sentRequest  [IRequest](#)

content  [byte\[\]](#)

responseMessage  [HttpResponseMessage](#)

## Properties

### Content

```
public byte[]? Content { get; }
```

Property Value

[byte](#)[]

## ResponseMessage

```
public HttpResponseMessage ResponseMessage { get; }
```

Property Value

[HttpResponseMessage](#)

## SentRequest

```
public IRequest SentRequest { get; }
```

Property Value

[IRequest](#)

## Methods

### Parse<T>(IParser<T, byte[]>)

```
public T Parse<T>(IParser<T, byte[]> parser)
```

Parameters

parser [IParser](#)<T, byte[]>

Returns

T

## Type Parameters

T

# Interface IResponse<TResponse>

Namespace: [GoogleMapsWrapper.Responses](#)

Assembly: GoogleMapsWrapper.dll

```
public interface IResponse<TResponse>
```

Type Parameters

TResponse

## Properties

Content

```
TResponse? Content { get; }
```

Property Value

TResponse

## ResponseMessage

```
HttpResponseMessage ResponseMessage { get; }
```

Property Value

[HttpResponseMessage](#) ↗

## SentRequest

```
IRequest SentRequest { get; }
```

Property Value

[IRequest](#)

## Methods

Parse<T>(IParser<T, TResponse>)

T Parse<T>(IParser<T, TResponse> parser)

Parameters

parser [IParser](#)<T, TResponse>

Returns

T

Type Parameters

T

# Class JsonResponse

Namespace: [GoogleMapsWrapper.Responses](#)

Assembly: GoogleMapsWrapper.dll

```
public class JsonResponse : IResponse<JsonDocument>
```

## Inheritance

[object](#) ← JsonResponse

## Implements

[IResponse<JsonDocument>](#)

## Inherited Members

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

## Constructors

JsonResponse(IRequest, JsonDocument, HttpResponseMessage)

```
public JsonResponse(IRequest sentRequest, JsonDocument content,  
HttpResponseMessage responseMessage)
```

## Parameters

sentRequest  [IRequest](#)

content  [JsonDocument](#)

responseMessage  [HttpResponseMessage](#)

## Properties

### Content

```
public JsonDocument Content { get; }
```

Property Value

[JsonDocument](#)

## ResponseMessage

```
public HttpResponseMessage ResponseMessage { get; }
```

Property Value

[HttpResponseMessage](#)

## SentRequest

```
public IRequest SentRequest { get; }
```

Property Value

[IRequest](#)

## Methods

### Parse<T>(IParser<T, JsonDocument>)

```
public T Parse<T>(IParser<T, JsonDocument> parser)
```

Parameters

parser [IParser](#)<T, [JsonDocument](#)>

Returns

T

## Type Parameters

T

# Namespace GoogleMapsWrapper.Types

## Structs

[GpsCoordinate](#)

## Enums

[ApiTypes](#)

Type of API per google. For categorization purposes.

[MapImageFormats](#)

Static map image formats.

[MapScaleTypes](#)

Type of scale that a map is displayed at. 1 = normal, 2 = high resolution.

[MapTypes](#)

[MarkerIconAnchorTypes](#)

Anchor type when using a custom static map icon.

[MarkerScaleTypes](#)

Scale type of a marker.

[MarkerSizes](#)

Type of scale that a map is displayed at. 1 = normal, 2 = high resolution.

[RequestTypes](#)

Type of request. For categorization purposes.

# Enum ApiTypes

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

Type of API per google. For categorization purposes.

```
public enum ApiTypes
```

## Fields

Environment = 3

Maps = 0

Places = 2

Routes = 1

# Struct GpsCoordinate

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

```
public readonly struct GpsCoordinate : IEquatable<GpsCoordinate>, IFormattable
```

Implements

[IEquatable](#)<[GpsCoordinate](#)>, [IFormattable](#)

Inherited Members

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

## Constructors

### GpsCoordinate(decimal, decimal)

```
public GpsCoordinate(decimal latitude, decimal longitude)
```

Parameters

latitude [decimal](#)

longitude [decimal](#)

## Properties

### Latitude

```
public decimal Latitude { get; }
```

Property Value

[decimal](#)

# Longitude

```
public decimal Longitude { get; }
```

## Property Value

[decimal](#)

# Methods

## Equals(GpsCoordinate)

Indicates whether the current object is equal to another object of the same type.

```
public bool Equals(GpsCoordinate other)
```

### Parameters

**other** [GpsCoordinate](#)

An object to compare with this object.

### Returns

[bool](#)

[true](#) if the current object is equal to the **other** parameter; otherwise, [false](#).

## Equals(object?)

Indicates whether this instance and a specified object are equal.

```
public override bool Equals(object? obj)
```

### Parameters

**obj** [object](#)

The object to compare with the current instance.

Returns

[bool](#)

[true](#) if [obj](#) and this instance are the same type and represent the same value; otherwise, [false](#).

## GetHashCode()

Returns the hash code for this instance.

```
public override int GetHashCode()
```

Returns

[int](#)

A 32-bit signed integer that is the hash code for this instance.

## Parse(string)

```
public static GpsCoordinate Parse(string s)
```

Parameters

[s](#) [string](#)

Returns

[GpsCoordinate](#)

## ToString()

Returns the fully qualified type name of this instance.

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

Returns

[string](#)

The fully qualified type name.

## ToString(string?, IFormatProvider?)

Formats the value of the current instance using the specified format.

```
public string ToString(string? format, IFormatProvider? formatProvider)
```

Parameters

**format** [string](#)

The format to use.

-or-

A null reference ([Nothing](#) in Visual Basic) to use the default format defined for the type of the [IFormattable](#) implementation.

**formatProvider** [IFormatProvider](#)

The provider to use to format the value.

-or-

A null reference ([Nothing](#) in Visual Basic) to obtain the numeric format information from the current locale setting of the operating system.

Returns

[string](#)

The value of the current instance in the specified format.

## TryParse(ReadOnlySpan<char>, out GpsCoordinate)

```
public static bool TryParse(ReadOnlySpan<char> s, out GpsCoordinate result)
```

Parameters

s [ReadOnlySpan](#)<char>

result [GpsCoordinate](#)

Returns

[bool](#)

## TryParse(string?, out GpsCoordinate)

```
public static bool TryParse(string? s, out GpsCoordinate result)
```

Parameters

s [string](#)

result [GpsCoordinate](#)

Returns

[bool](#)

## Operators

### operator ==(GpsCoordinate, GpsCoordinate)

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

Parameters

left [GpsCoordinate](#)

right [GpsCoordinate](#)

Returns

[bool](#)

## operator !=(GpsCoordinate, GpsCoordinate)

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

Parameters

[left](#) [GpsCoordinate](#)

[right](#) [GpsCoordinate](#)

Returns

[bool](#)

# Enum MapImageFormats

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

Static map image formats.

```
public enum MapImageFormats
```

## Fields

Gif = 2

Jpg = 3

Png = 0

Png32 = 1

# Enum MapScaleTypes

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

Type of scale that a map is displayed at. 1 = normal, 2 = high resolution.

```
public enum MapScaleTypes
```

## Fields

HighRes = 2

Normal = 1

# Enum MapTypes

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

```
public enum MapTypes
```

## Fields

Hybrid = 3

RoadMap = 0

Satellite = 1

Terrain = 2

# Enum MarkerIconAnchorTypes

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

Anchor type when using a custom static map icon.

```
public enum MarkerIconAnchorTypes
```

## Fields

Bottom = 1

BottomLeft = 7

BottomRight = 8

Center = 4

Left = 2

Right = 3

Top = 0

TopLeft = 5

TopRight = 6

# Enum MarkerScaleTypes

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

Scale type of a marker.

```
public enum MarkerScaleTypes
```

## Fields

Large = 4

Medium = 2

Normal = 1

# Enum MarkerSizes

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

Type of scale that a map is displayed at. 1 = normal, 2 = high resolution.

```
public enum MarkerSizes
```

## Fields

Mid = 1

Small = 2

Tiny = 0

# Enum RequestTypes

Namespace: [GoogleMapsWrapper.Types](#)

Assembly: GoogleMapsWrapper.dll

Type of request. For categorization purposes.

```
public enum RequestTypes
```

## Fields

Elevation = 1

Geocoding = 2

StaticMaps = 0

# Namespace GoogleMapsWrapper.Utilities

## Classes

[PolylineEncoder](#)

[Utilities](#)

A static class for generic utility functions that may be used in this application.

# Class PolylineEncoder

Namespace: [GoogleMapsWrapper.Utilities](#)

Assembly: GoogleMapsWrapper.dll

```
public static class PolylineEncoder
```

## Inheritance

[object](#) ← PolylineEncoder

## Inherited Members

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

## Methods

### Encode(IList<GpsCoordinate>)

```
public static string Encode(IList<GpsCoordinate> coordinates)
```

#### Parameters

coordinates [IList](#)<[GpsCoordinate](#)>

#### Returns

[string](#)

# Class Utilities

Namespace: [GoogleMapsWrapper.Utilities](#)

Assembly: GoogleMapsWrapper.dll

A static class for generic utility functions that may be used in this application.

```
public static class Utilities
```

## Inheritance

[object](#) ← Utilities

## Inherited Members

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

# Methods

## ColorToHex(Color)

Convert a .NET Color object to a hexadecimal notation. If transparency (alpha) is provided in the color parameter, outputs 32 bit hex, else 24.

```
public static string ColorToHex(Color color)
```

### Parameters

color [Color](#)

### Returns

[string](#)

## UrlEncodeChar(char)

Convert a char to a URL encoded string representation.

```
public static string UrlEncodeChar(char c)
```

Parameters

c [char](#)

Returns

[string](#)