

[MapKit](#) / MapKit for AppKit and UIKit

API Collection

# MapKit for AppKit and UIKit

## Topics

### Essentials

 Enabling Maps capability in Xcode

Configure your routing app to support providing directions.

 Identifying unique locations with Place IDs

Obtain information about a point of interest that persists over its lifetime.

`class MKMapView`

An embeddable map interface, similar to the one that the Maps app provides.

`class MKMapItem`

A point of interest on the map.

### Map coordinates

`struct MKCoordinateRegion`

A rectangular geographic region that centers around a specific latitude and longitude.

`struct MKCoordinateSpan`

The width and height of a map region.

`struct MKMapRect`

A rectangular area on a two-dimensional map projection.

```
struct MKMapPoint
```

A point on a two-dimensional map projection.

```
struct MKMapSize
```

Width and height information on a two-dimensional map projection.

```
class MKDistanceFormatter
```

A utility object that converts between a geographic distance and a string-based expression of that distance.

## Map customization

```
class MKMapCamera
```

A virtual camera for defining the appearance of the map.

```
class MKCompassButton
```

A specialized view that displays the compass heading for its associated map.

```
class MKScaleView
```

A specialized view that displays the scale information for its associated map.

```
class MKZoomControl
```

A specialized view that displays and controls the zoom level of the map view.

```
class MKPitchControl
```

A specialized view that displays and controls the pitch angle of the map view.

```
class MKUserTrackingButton
```

A specialized button that allows the user to toggle whether the map tracks to the heading the user is facing.

```
class MKUserTrackingBarButtonItem
```

A specialized bar button item that allows the user to toggle whether the map tracks to the heading the user is facing.

## Annotations and overlays

☰ MapKit annotations

Create annotations to add indicators and additional details for specific locations on a map.

☰ MapKit overlays

Create overlays to highlight geographic regions or paths.

## Directions

`class MKDirections`

A utility object that computes directions and travel-time information based on the route information you provide.

`class Request`

The start and end points of a route, along with the planned mode of transportation.

`class Response`

The route information that Apple servers return in response to your request for directions.

`class ETAResponse`

The travel-time information that Apple servers return.

`class MKRoute`

A single route between a requested start and end point.

`class Step`

One portion of an overall route.

## Geographical features

{ } Displaying an Indoor Map

Use the Indoor Mapping Data Format (IMDF) to show an indoor map with custom overlays and points of interest.

`class MKGeoJSONDecoder`

An object that decodes GeoJSON objects into MapKit types.

`class MKGeoJSONFeature`

The decoded representation of a GeoJSON feature.

`protocol MKGeoJSONObject`

Objects that the GeoJSON decoder can return.

## Local search

{ } Interacting with nearby points of interest

Provide automatic search completions for a partial search query, search the map for relevant locations nearby, and retrieve details for selected points of interest.

#### enum MKLocalSearchRegionPriority

A value that indicates the importance of the configured region.

#### struct ResultType

Options that indicate types of search results.

#### class MKLocalSearch

A utility object for initiating map-based searches and processing the results.

#### struct Options

A structure that contains options for filtering results in a search.

#### class MKAddressFilter

An object that filters which address options to include or exclude in search results.

#### struct ResultType

Options that indicate types of search completions.

#### class MKLocalSearchCompleter

A utility object for generating a list of completion strings based on a partial search string that you provide.

#### class MKLocalSearchCompletion

A fully formed string that completes a partial string.

#### class MKLocalPointsOfInterestRequest

A structured request to use when searching for points of interest.

## Exploring at street level

#### class MKLookAroundScene

A utility class that encapsulates information the framework requires to retrieve and display a specific Look Around location's imagery.

#### class MKLookAroundSceneRequest

A class you use to request a LookAround scene at the location you specify.

#### class MKLookAroundViewController

A class that manages the presentation and display of a LookAround view.

```
class MKLookAroundSnapshotter
```

A utility class that you use to create a static image from a LookAround scene.

## Place information

```
protocol MKMapItemDetailViewControllerDelegate
```

The methods that you use to receive events from an associated map view controller.

```
class MKMapItemDetailViewController
```

An object that displays detailed information about a map item.

```
class MapItemDetailPresentationStyle
```

The type of map item detail accessory presentation to use.

```
class MKSelectionAccessory
```

The type of accessory to display for a selected annotation.

```
enum CalloutStyle
```

The style to use for a map item detail callout presentation.

## Points of interest

 Identifying unique locations with Place IDs

Obtain information about a point of interest that persists over its lifetime.

```
class MKMapFeatureAnnotation
```

A class that describes an annotation element on the map's display such as a point of interest, territorial boundary, or physical feature.

```
struct MKMapFeatureOptions
```

A structure you use to tell the map which kinds of features users can interact with.

```
class MKMapItemRequest
```

A utility class you use to request additional information about a map feature.

```
class MKIconStyle
```

A class you use to customize the annotation view icon of a point of interest (POI) on the map.

```
class MKPointOfInterestFilter
```

A filter that includes or excludes point of interest categories from a map view, local search, or local search completer.

```
struct MKPointOfInterestCategory
```

A point of interest category.

## Static map snapshots

```
class MKMapSnapshotter
```

A utility class for capturing a map and its content into an image.

```
class Snapshot
```

An image that a snapshotter object generates.

## Reference

### ☰ MapKit Functions

The functions of the MapKit framework provide convenient ways to package map-related data structures.

## Errors

```
let MKErrorDomain: String
```

The error domain for MapKit.

```
struct MKError
```

Error constants for the MapKit framework.

```
enum Code
```

Error constants for the MapKit framework.

## Deprecated

### ☰ Deprecated Symbols

Map protocols and view modifiers that are no longer supported.

## See Also

[The MapKit APIs](#)

## ☰ MapKit for SwiftUI

MapKit for SwiftUI allows you to build map-centric views and apps across Apple platforms. You can design expressive and highly interactive Maps with minimal code by composing views, using ViewBuilders and view modifiers.

### 📄 Adopting unified Maps URLs

Access Maps URLs and options for displaying Maps information across Apple platforms.