

## Documentation

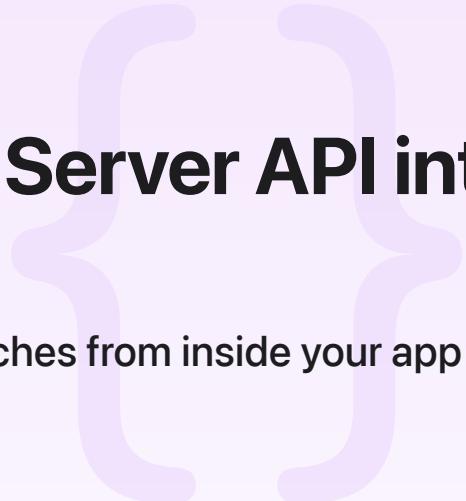
[Apple Maps Server API](#) / Integrating the Apple Maps Server API into Java server applications

Sample Code

# Integrating the Apple Maps Server API into Java server applications

Streamline your app's API by moving georelated searches from inside your app to your server.

[Download](#)



## Overview

This sample demonstrates how to integrate the Apple Maps Server API into Java-based apps.

The `MapsApiClientDemo.java` file demonstrates how you use the Apple Maps Server APIs and the following API features:

- Getting an Access Token — Authenticate with the service and retrieve an Apple Maps Server API token.
- Geocoding — Retrieve the latitude and longitude from a text address.
- Reverse Geocoding — Retrieve a list of addresses that are present at the specified latitude and longitude.
- Searching — Search for locations by criteria you provide.
- SearchAutoComplete - Get a list of autocomplete results for the specified search query.
- ETAs — Calculate estimated times of arrival (ETAs) between a specified starting location and one or more destinations.
- Directions - Get directions between origin and destination points.

## Note

This sample code project is associated with WWDC22 session: 10006 [Meet Apple Maps Server APIs](#)

# Configure the sample code project

To build this sample, you need the following tools and other information:

- [Java 17](#) — This sample code can run on older versions of Java with some minor modifications, depending upon your Java installation.
- [Gradle](#) — The project includes a Gradle command wrapper that uses Gradle version 7.5.1; you may a different version if you need to use a different Java installation.
- Your Apple Developer team ID — This is a 10-character team ID you obtain from the membership tab in your Apple Developer portal account.
- A Maps key ID and private key — This is a 10-character key identifier that provides the ID of the private key and the private key that you obtain from your Apple Developer portal account. To create a Maps ID and private key, follow the steps in [Creating a Maps identifier and a private key](#).

In the `MapsApiClientDemo.java` file, edit the `createJwt()` method to set the `teamId`, `keyId`, and `key` variables to the values you obtained from your Apple Developer portal account.

# Run the sample

To run the sample, enter the following commands in Terminal while in the `server-api-examples` directory:

```
% gradle wrapper  
% ./gradlew clean run
```

# See Also

## Essentials

- 📄 [Creating and using tokens with Maps Server API](#)  
Sign JSON Web Tokens to use Maps Server API and debug common signing errors.
- 📄 [Creating a Maps identifier and a private key](#)

Create a Maps identifier and a private key before generating tokens for MapKit JS.

## Generate a Maps token

Returns a JWT maps access token that you use to call the service API.

### Debugging an Invalid token

Inspect the JavaScript console logs, the token, and events to determine why a token is invalid.

### Common objects

Understand the common JSON objects that API responses contain.