

# Mapbox

# Web Mapping Libraries

Google Maps JavaScript API

ArcGIS Maps SDK for JavaScript

Leaflet

OpenLayers

Mapbox gl js

CesiumJS

...

# What is Mapbox?

Mapbox is a platform that provides mapping services, allowing developers to create and customize maps for web and mobile applications. It offers a variety of APIs and SDKs for integrating maps, search, navigation, and location data into applications. Mapbox also provides tools like [Mapbox Studio](#) for designing custom map styles.

3dgisking ▾

🏠 Home

Tools

🔗 Style editor

🗂 Data manager

Admin

</> Tokens

📄 Invoices

📊 Statistics

⚙ Settings

Help & resources

Docs >

Ask AI >

Tutorials >

Playgrounds >

Troubleshooting >

Blog >

# Styles

New folder New style ▾

🔍 Search

☐ ▾ 0 selected 📁 🔒 🗑

Filter by: All ▾ Sort by: Date edited ▾ | 1-2 of 2 files | Trash



Untitled  
● Published Oct 28, 2024



Untitled  
● Published Oct 28, 2024 • Public



## Getting started

- 📖 Read the Studio Manual
- 📱 Preview styles on iOS or Android
- 🔍 Find inspiration in the style gallery
- 📺 Watch how to videos



## Mapbox GL JS

## GUIDES ^

Getting Started  
Map Styles  
User interaction  
Migrate to Mapbox GL JS v3  
Globe and Atmosphere  
Projections  
Browsers and testing  
Pricing

## API REFERENCE ▼

PLUGINS AND  
FRAMEWORKS

## EXAMPLES

## STYLE SPECIFICATION ↗

## TUTORIALS ↗

## TROUBLESHOOTING ↗

## HOW-TO VIDEOS ↗

## Mapbox GL JS

Current version: [v3.11.0](#) [View changelog](#)

- ✓ Custom map styles
- ✓ Fast vector maps
- ✓ Compatible with other Mapbox tools

Install

Contribute on GitHub



**Mapbox GL JS** is a client-side JavaScript library for building web maps and web applications with Mapbox's modern mapping technology. You can use Mapbox GL JS to display Mapbox maps in a web browser or client, add user interactivity, and customize the map experience in your application.



&lt;!DOCTYPE html&gt;

&lt;html&gt;

&lt;head&gt;

## On this page

Use cases  
Key concepts  
Mapbox GL  
Client-side rendering  
The Map class  
Layers  
Camera  
Use Mapbox GL JS with other tools  
Use your own data  
Style your maps  
Interactivity  
Mapbox web services APIs  
JavaScript frameworks  
Attribution

## Additional Developer Resources

- Mapbox Developer Discord
- Developer Cheatsheet
- Mapbox Support
- Ask AI




# Getting Started

## Prerequisites

Before you get started with Mapbox GL JS, you need to have a Mapbox access token and add Mapbox GL JS to your project using either the CDN or the `mapbox-gl` npm package.

To use Mapbox GL JS in your project, you need to install the `mapbox-gl` npm package.

A CDN, or Content Delivery Network, is a network of distributed servers that aim to deliver web content to users faster and more efficiently. It works by caching and storing website files, like images and HTML, on servers geographically closer to the users, reducing the distance that data needs to travel. This results in quicker load times and a better user experience. 

```
<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Mapbox GL JS map</title>

<meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no">

<link href="https://api.mapbox.com/mapbox-gl-js/v3.11.0/mapbox-gl.css" rel="stylesheet">

<script src="https://api.mapbox.com/mapbox-gl-js/v3.11.0/mapbox-gl.js"></script>

<style>

body { margin: 0; padding: 0; }

#map { position: absolute; top: 0; bottom: 0; width: 100%; }

</style>

</head>

<body>

<div id="map"></div>

<script>

    mapboxgl.accessToken = 'pk.eyJ1Ijoim2RnaXNraW5nIiwiaSI6ImNtMnNvYmF5bDF3OGcyY3E4Mm41MjJhYWUifQ.eyJ17EtxA-
    HVY3fUTDXqZw';

    const map = new mapboxgl.Map({

        container: 'map', // container ID

        center: [-74.5, 40], // starting position [lng, lat]. Note that lat must be set between -90 and 90

        zoom: 9 // starting zoom

    });

</script>
```

```
import mapboxgl from 'mapbox-gl'; // or "const mapboxgl = require('mapbox-gl');"

import 'mapbox-gl/dist/mapbox-gl.css';

mapboxgl.accessToken =

'pk.eyJ1IjoiM2RnaXNraW5nIiwiaSI6ImNtMnNvYmF5bDF3OGcyY3E4Mm41MjJhYWUifQ.eyJl7EtxA

-HVY3fUTDXqZw';

const map = new mapboxgl.Map({

  container: 'map', // container ID

  style: 'mapbox://styles/mapbox/streets-v12', // style URL

  center: [-74.5, 40], // starting position [lng, lat]

  zoom: 9, // starting zoom

});
```



Interactive, thoroughly customizable maps in the browser, powered by vector tiles and WebGL

[docs.mapbox.com/mapbox-gl-js/](https://docs.mapbox.com/mapbox-gl-js/)

javascript webgl maps priority 3d

Readme

View license

Code of conduct

Security policy

Activity

Custom properties

11.6k stars

371 watching

2.3k forks

Report repository

Releases 241

v3.11.1 Latest 5 days ago

+ 240 releases

Used by 143k

mourner v3.12.0-beta.1

3082753 · last week

12,133 Commits

.circleci	Increase browser-tools 1.5.1 (internal-2163)	2 months ago
.github	Update CODEOWNERS (#13449)	3 weeks ago
3d-style	<a href="#">Initial building layer style spec (internal-2375)</a>	last week
build	Initial building layer style spec (internal-2375)	last week
debug	Add ImageProvider (internal-2363)	last week
dist	Switch dev environment to native ESM to support Node 12+...	4 years ago
rollup	Remove the old benchmarks in the bench folder (internal-867)	2 years ago
src	v3.12.0-beta.1	last week
test	[MAPS3D-1670] Port 3D intersections shadows (internal-2320)	last week
.browserslistrc	[GLJS-888] Add .browserslistrc (internal-2386)	last week
.editorconfig	EditorConfig updates (#3537) [ckip ci]	9 years ago
.gitignore	Add uSVG protobuf renderer (internal-1759)	6 months ago
.npmignore	Add diff-tarball script and more explicit files property in pac...	5 years ago
.nvmrc	Switch to Node v20 (internal-1654)	9 months ago
.stylelintrc	Bump stylelint-config-standard from 36.0.1 to 37.0.0 (interna...	3 months ago
tonicssuperc	Update high priority label	7 years ago

```

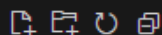
4494 .....}^
4495 ^
4496 .....if (this._loaded && !this._fullyLoaded && !somethingDirty) {^
4497 .....this._fullyLoaded = true;^
4498 .....LivePerformanceUtils.mark(LivePerformanceMarkers.fullLoad);^
4499 .....// Following lines are billing and metrics related code. Do not change. See LICENSE.txt^
4500 .....if (this._performanceMetricsCollection) {^
4501 .....postPerformanceEvent(this._requestManager._customAccessToken, {^
4502 .....width: this.painter.width,^
4503 .....height: this.painter.height,^
4504 .....interactionRange: this._interactionRange,^
4505 .....visibilityHidden: this._visibilityHidden,^
4506 .....terrainEnabled: !!this.painter.style.getTerrain(),^
4507 .....fogEnabled: !!this.painter.style.getFog(),^
4508 .....projection: this.getProjection().name,^
4509 .....zoom: this.transform.zoom,^
4510 .....renderer: this.painter.context.renderer,^
4511 .....vendor: this.painter.context.vendor^
4512 .....});^
4513 .....}^
4514 .....this.authenticate();^
4515 .....}^
4516 .....}^
4517 ^
4518 ....._forceMarkerAndPopupUpdate(shouldWrap?: boolean) {^
4519 .....for (const marker of this._markers) {^
4520 .....// Wrap marker location when toggling to a projection without world copies^
4521 .....if (shouldWrap && !this.getRenderWorldCopies()) {^
4522 .....marker._lngLat = marker._lngLat.wrap();^
4523 .....}^
4524 .....marker._update();^
4525 .....}^
4526 .....for (const popup of this._popups) {^
4527 .....// Wrap popup location when toggling to a projection without world copies and track pointer set
4528 .....if (shouldWrap && !this.getRenderWorldCopies() && !popup._trackPointer) {^
4529 .....popup._lngLat = popup._lngLat.wrap();^
4530 .....}^
4531 .....popup._update();^
4532 .....}^
4533 .....}^
4534 .....}^
4535 .....// **^

```

```

4494 .....}^
4495 ^
4496 .....if (this._loaded && !this._fullyLoaded && !somethingDirty) {^
4497 .....this._fullyLoaded = true;^
4498 .....LivePerformanceUtils.mark(LivePerformanceMarkers.fullLoad);^
4499 .....// Following lines are billing and metrics related code. Do not change. See LICENSE.txt^
4500 .....if (this._performanceMetricsCollection) {^
4501 .....postPerformanceEvent(this._requestManager._customAccessToken, {^
4502 .....width: this.painter.width,^
4503 .....height: this.painter.height,^
4504 .....interactionRange: this._interactionRange,^
4505 .....visibilityHidden: this._visibilityHidden,^
4506 .....terrainEnabled: !!this.painter.style.getTerrain(),^
4507 .....fogEnabled: !!this.painter.style.getFog(),^
4508 .....projection: this.getProjection().name,^
4509 .....zoom: this.transform.zoom,^
4510 .....renderer: this.painter.context.renderer,^
4511 .....vendor: this.painter.context.vendor^
4512 .....});^
4513 .....}^
4514 .....// this.authenticate();^
4515 .....}^
4516 .....}^
4517 ^
4518 ....._forceMarkerAndPopupUpdate(shouldWrap?: boolean) {^
4519 .....for (const marker of this._markers) {^
4520 .....// Wrap marker location when toggling to a projection without world copies^
4521 .....if (shouldWrap && !this.getRenderWorldCopies()) {^
4522 .....marker._lngLat = marker._lngLat.wrap();^
4523 .....}^
4524 .....marker._update();^
4525 .....}^
4526 .....for (const popup of this._popups) {^
4527 .....// Wrap popup location when toggling to a projection without world copies and track pointer set
4528 .....if (shouldWrap && !this.getRenderWorldCopies() && !popup._trackPointer) {^
4529 .....popup._lngLat = popup._lngLat.wrap();^
4530 .....}^
4531 .....popup._update();^
4532 .....}^
4533 .....}^
4534 .....}^
4535 .....// **^

```



- > .circleci
- > .github
- > 3d-style
- > build
- > debug
- > dist
- > node\_modules
- > rollup
- > src
- > test
- .editorconfig
- .eslintrc
- .gitignore
- .npmignore
- .nvmrc
- .stylelintrc
- .topissuesrc
- ARCHITECTURE.md
- buildspec.yml
- CHANGELOG.md
- CODE-OF-CONDUCT.md
- CONTRIBUTING.md
- LICENSE.txt
- package-lock.json
- package.json
- postcss.config.cjs

package.json &gt; ...

```
10   "repository": {
11     "url": "git://github.com/mapbox/mapbox-gl-js.git"
12   },
13   "workspaces": [
14     "src/style-spec",
15     "test/build/typings"
16   ],
17   "dependencies": { ...
47 },
48   "devDependencies": { ...
120   },
121   "scripts": {
122     "build-dev": "rollup -c --environment BUILD:dev",
123     "watch-dev": "rollup -c --environment BUILD:dev --watch",
124     "build-bench": "rollup -c --environment BUILD:bench,MINIFY:true",
125     "build-prod": "rollup -c --environment BUILD:production",
126     "build-prod-min": "rollup -c --environment BUILD:production,MINIFY:true",
127     "build-csp": "rollup -c rollup.config.csp.js",
128     "build-test-suite": "rollup -c test/integration/rollup.config.test.js"
```

PROBLEMS 1

OUTPUT

TERMINAL

PORTS

GITLENS

TERMINAL

```
PS D:\MyResearch\mapbox-gl\mapbox-gl-js> yarn build-dev
yarn run v1.22.19
$ rollup -c --environment BUILD:dev
```

```
src/index.ts, src/source/worker.ts → rollup/build/mapboxgl...
```

```
(!) Circular dependencies
```

```
src/style-spec/ast/interpolate.ts > src/style-spec/ast/interpolate.ts > src/style-spec/ast/interpolate.ts
```

# <https://docs.mapbox.com/mapbox-gl-js/example/>

## Examples

### Search

### Topics

- 3D
- Atmosphere
- Browser support
- Camera
- Controls and overlays
- Expressions
- Geocoder
- Getting started



Click to load

### Use a clip layer to replace a landmark on the map

Use a clip layer to remove a 3D building from the Mapbox Standard style and add a custom 3D model.



Click to load

### Set initial map's style configuration property

Use different initial configuration properties with Mapbox Standard Style.



Click to load

### Initialize a map using a bounding box

Initialize a map using a bounding box instead of center and zoom.



Click to load

### Add a patterned line to a map

Add a line layer with a repeating pattern to the map.



Click to load



Click to load



Click to load



# https://docs.mapbox.com/help/tutorials/?product=Mapbox+GL+JS

mapbox | Docs

Maps Navigation Search Data Help

Q Search... Ctrl K

English

Help

GETTING STARTED  
DIVE DEEPER  
TUTORIALS  
HOW-TO VIDEOS  
TROUBLESHOOTING  
FAQS  
GLOSSARY

All docs > Help > Tutorials

?

## Tutorials

Find a step-by-step guide to help you get started or take your project to the next level.

Search

Products

Languages

Q Search title and description...

Mapbox GL JS x

Choose language...

23 RESULTS [Clear filters](#)

**Add points to a web map, Part 3: add interactivity**

Add popups when markers are clicked using Mapbox GL JS.

 JS JavaScript

Start

**Analyze data with Turf.js and Mapbox GL JS**

Using Turf.js, add spatial analysis to our map to solve problems. This guide walks through an example of Turf.js and Mapbox GL JS in a real-world context.

 JS JavaScript

Start

**Build a store locator using Mapbox GL JS**

Build a map application with Mapbox GL JS. This guide walks you through all the code that you need to build a store locator.

 JS JavaScript

Start

**Make a choropleth map, part 2: add interactivity**

Publish your style with Mapbox GL JS, create a legend, and add interactive elements.

 JS JavaScript

Start

**Create interactive hover effects with Mapbox GL JS**

Use feature state and expressions with Mapbox GL JS to dynamically style individual features in a map that shows earthquakes from the past week.

 JS JavaScript

Start

**Add custom markers in Mapbox GL JS**

Add custom HTML markers, style them, and add tooltips with Mapbox GL JS.

 JS JavaScript

Start

# Mapbox Style Specification

A Mapbox [style](#) is a document that defines the visual appearance of a map: what data to draw, the order to draw it in, and how to style the data when drawing it. A style document is a [JSON](#) object with specific root level and nested properties. This specification defines and describes these properties.

The intended audience of this specification includes:

- Advanced designers and cartographers who want to write styles by hand rather than use [Mapbox Studio](#).
- Developers using style-related features of [Mapbox GL JS](#), the [Mapbox Maps SDK for Android](#), or the [Mapbox Maps SDK for iOS](#).
- Authors of software that generates or processes Mapbox styles.

Explore the Style Spec Reference to find details about each property, including its type, default value, and a description of how the property works.

## Style document structure

A Mapbox style consists of a set of [root properties](#), some of which describe a single global property, and some of which contain nested properties. Some root properties, like [version](#), [name](#), and [metadata](#), don't have any influence over the appearance or behavior of your map, but provide important descriptive information related to your map. Others, like [layers](#) and [sources](#), are critical and determine which map features will appear on your map and what they will look like. Some properties, like [center](#), [zoom](#), [pitch](#), and [bearing](#), provide the map renderer with a set of defaults to be used when initially displaying the map.

The snippet below shows an example style JSON, including the basic structure and some of the most common properties. See the [Root](#) section of the spec reference for the full list of root properties.

---

# Sample style document

```
{

  "name": "some-style",

  "version": 8,

  // default camera position

  "center": [ -74, 40.73 ],

  "zoom": 11.3,

  "bearing": 20.8,

  "pitch": 17.5,

  // sprites and fonts

  "sprite": "mapbox://sprites/mapbox/light-v11",

  "glyphs": "mapbox://fonts/mapbox/{fontstack}/{range}.pbf",

  // spatial data to include, used one-to-many with layers

  "sources": {

    "some-vector-source": {

      "type": "vector",
```



# Vector Tile



Docs

Maps

Navigation

Search

Data

Help

## Help

GETTING STARTED

DIVE DEEPER

TUTORIALS

HOW-TO VIDEOS

TROUBLESHOOTING

FAQS

GLOSSARY

[All docs](#) > [Help](#) > [Glossary](#) > vector tiles

## vector tiles

A **vector tile** is a lightweight data format for storing geospatial vector data, such as points, lines, and polygons.

Vector tiles are used to create Mapbox [vector tilesets](#). Vector tiles can be used as sources for [styles](#) or queried directly to create interactive experiences with the [Mapbox Mobile SDKs](#) and [Mapbox GL JS](#).

Vector tiles encode geographic information according to the [Vector tile specification](#).

### Related resources:

- [Vector tiles documentation](#)
- [Vector Tiles API documentation](#)

QGIS "Untitled Project" — QGIS

Project Edit View Layer Settings Plugins Vector Raster Database Web Mesh Processing Help

Browser

- Favorites
- Spatial Bookmarks
- Home
- C:\(未命名)
- D:\(未命名)
- GeoPackage
- Spatialite
- PostgreSQL
- SAP HANA
- MS SQL Server
- Oracle
- WMS/WMTS
- test
- Scenes
- SensorThings
- Vector Tiles
- test

Layer Properties

Metadata Preview

General

Name test

URL <http://localhost:1217/api.mapbox.com/v4/mapbox.mapbox-bathymetry-v2.mapbox.mapbox-streets-v8/z10x10/vector.tif>

Source type=xyz&url=http://localhost:1217/api.mapbox.com/v4/mapbox.mapbox-bathymetry-v2.mapbox.mapbox-streets-v8/%7B%7D/%7B%7D/vector.pbf&zmax=14&zmin=0&http-header=referer=xyzvectortiles

Provider xyzvectortiles

Information from provider

Source type xyz

Zoom levels 0 - 14

Coordinate Reference System (CRS)

Name	EPSG:3857 - WGS 84 / Pseudo-Mercator
Units	meters
Type	Projected
Method	Mercator
Celestial Body	Earth
Accuracy	Based on World Geodetic System 1984 ensemble ITPSG

Close

Processing Toolbox

Search...

- Recently used
- 3D Tiles
- Cartography
- Database

Identify Results

Feature

- test [3]
- depth
- (Derived)
- min\_depth
- place\_label
- water

Mode Current Layer

View Tree

Coordinate -27745540, 8052110 Scale 178493664 Magnifier 100% Rotation 0.0° Render EPSG:3857

8:44 PM

# Add 3D terrain to a map



```
<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Add 3D terrain to a map</title>

<meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no">

<link href="https://api.mapbox.com/mapbox-gl-js/v3.11.0/mapbox-gl.css" rel="stylesheet">

<script src="https://api.mapbox.com/mapbox-gl-js/v3.11.0/mapbox-gl.js"></script>

<style>

body { margin: 0; padding: 0; }

#map { position: absolute; top: 0; bottom: 0; width: 100%; }

</style>

</head>

<body>

<div id="map"></div>

<script>

    mapboxgl.accessToken = 'pk.eyJ1Ijoim2RnaXNraW5nliwiYSI6ImNtMnNvYmF5bDF3OGcyY3E4Mm41MjJhYWUifQ.eyJl7EtxA-HVY3IUTDXqZw';

    const map = new mapboxgl.Map({

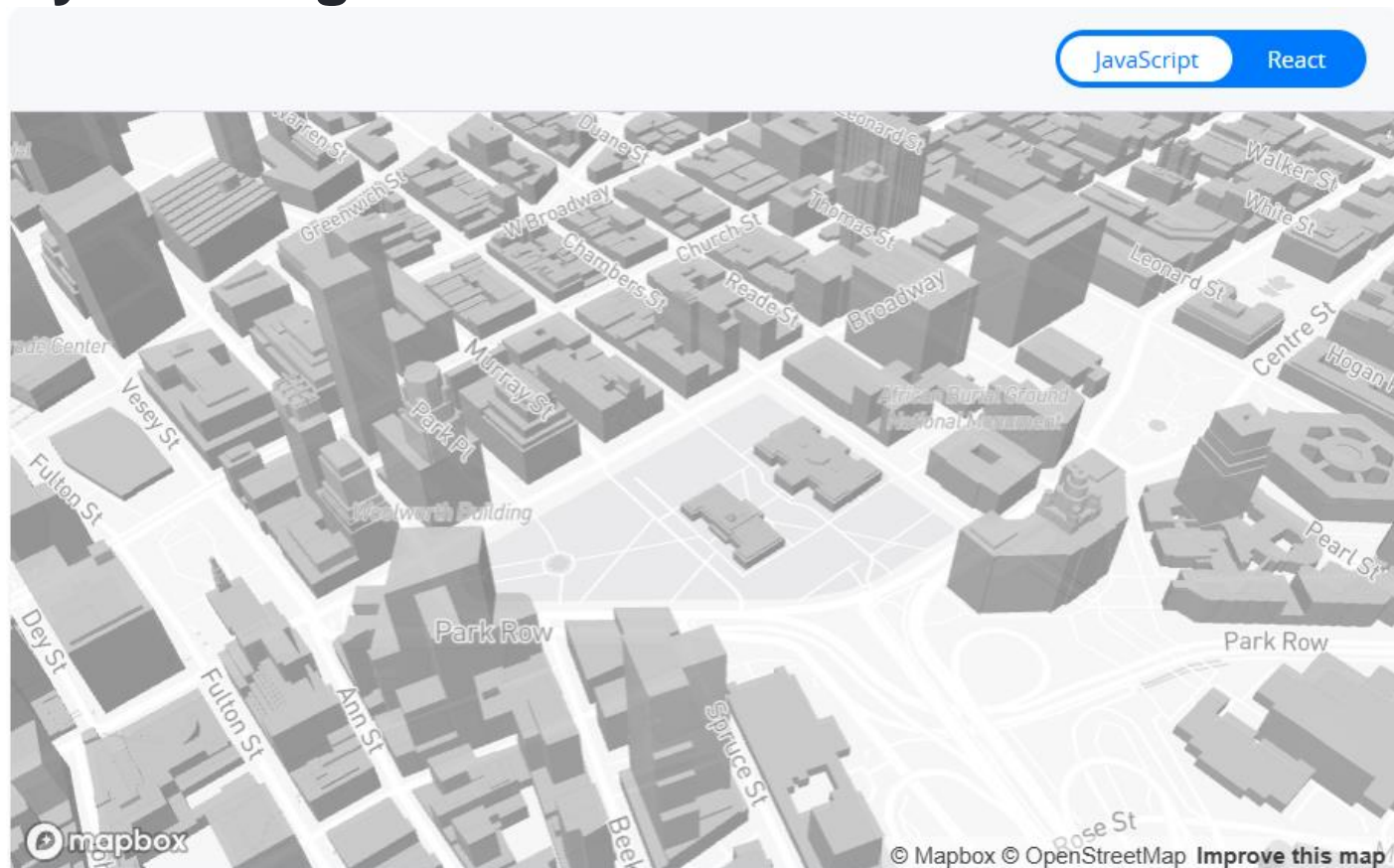
        container: 'map',

        zoom: 14,

        center: [-114.26608, 32.7213],

        style: 'mapbox://styles/mapbox/streets-v11'
```

# Display buildings in 3D



```
<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Display buildings in 3D</title>

<meta name="viewport" content="initial-scale=1,maximum-scale=1,user-scalable=no">

<link href="https://api.mapbox.com/mapbox-gl-js/v3.11.0/mapbox-gl.css" rel="stylesheet">

<script src="https://api.mapbox.com/mapbox-gl-js/v3.11.0/mapbox-gl.js"></script>

<style>

body { margin: 0; padding: 0; }

#map { position: absolute; top: 0; bottom: 0; width: 100%; }

</style>

</head>

<body>

<div id="map"></div>

<script>

    mapboxgl.accessToken = 'pk.eyJ1Ijo1M2RnaXNraW5nliwiYSI6ImNtMnNvYmF5bDF3OGcyY3E4Mm41MjJhYWUifQ.eyjI7EtXA-HVY3fUTDXqZw';

    const map = new mapboxgl.Map({

        // Choose from Mapbox's core styles, or make your own style with Mapbox Studio

        style: 'mapbox://styles/mapbox/light-v11',

        center: [-74.0066, 40.7135],

        zoom: 15.5,
```