# JS

Start integration with your preferred language.

Get started for free

- Ruby
- Python
- JS
- Golang
- PHP

## Code to integrate

- Java
- Rust
- .NET .Net
- Swift
- C++

```
const { getJson } = require("serpapi");

getJson({
  q: "Coffee",
  location: "Austin, Texas, United States",
  hl: "en",
  gl: "us",
  google_domain: "google.com",
  api_key: "secret_api_key"
}, (json) => {
  console.log(json);
});
```

# SerpApi for JavaScript/TypeScript

npm v2.2.1 Deno version Build no status license MIT SerpApi Libraries

Scrape and parse search engine results using SerpApi. Get search results from Google, Bing, Baidu, Yandex, Yahoo, Home Depot, eBay and more.

🪧 Coming from `google-search-results-nodejs` ? Check out the migration document to find out how to upgrade.

## Quick start

### Node.js

- Supports Node.js 7.10.1 and newer.
- Refer to this example for help.

```
npm install serpapi
# or if you prefer yarn
yarn add serpapi
```

```
const { getJson } = require("serpapi");
getJson({
  engine: "google",
  api_key: API_KEY, // Get your API_KEY from https://serpapi.com/manage-api-key
  q: "coffee",
  location: "Austin, Texas",
}, (json) => {
  console.log(json["organic_results"]);
});
```

## Node.js with ES Modules (ESM) and top-level await

- If you prefer using the `import` syntax and top-level `await` , you need to use at least Node.js 14.8.0.
- Refer to this example for help.

You will need to add `"type": "module"` to your `package.json` :

```
{
  "type": "module",
  // rest of package.json
}
```

```
import { getJson } from "serpapi";
const response = await getJson({
  engine: "google",
  api_key: API_KEY, // Get your API_KEY from https://serpapi.com/manage-api-key
  q: "coffee",
  location: "Austin, Texas",
});
console.log(response);
```

### Deno

- Import directly from deno.land.
- Usage is otherwise the same as above.
- Refer to this example for help.

```
import { getJson } from "https://deno.land/x/serpapi/mod.ts";
const response = await getJson({
  engine: "google",
  api_key: API_KEY, // Get your API_KEY from https://serpapi.com/manage-api-key
  q: "coffee",
  location: "Austin, Texas",
});
console.log(response);
```

## Features

- TypeScript support.
- Works out-of-the-box with Node.js and Deno.
- Promises and async/await support.

- Callbacks support.
- Examples in JavaScript/TypeScript on Node.js/Deno using ESM/CommonJS, and more.

## Configuration

You can declare a global `api_key` and `timeout` value by modifying the `config` object. `timeout` is defined in milliseconds and defaults to 60 seconds.

All functions, other than `getLocations` , accepts an optional `api_key` and `timeout` that will take precedence over the values defined in `config` .

`getLocations` doesn't require an API key.

```
import { config, getJson } from "serpapi";

config.api_key = API_KEY;
config.timeout = 60000;

await getJson({ engine: "google", q: "coffee" }); // uses the API key defined in the conf
await getJson({ engine: "google", api_key: API_KEY_2, q: "coffee" }); // API_KEY_2 will b
```

### Using a Proxy

You can use a proxy by passing `requestOptions` with an `HttpsProxyAgent` instance. This can be done either globally through the config object or per-request in the parameters.

First, install the required package:

```
npm install https-proxy-agent
# or if you prefer yarn
yarn add https-proxy-agent
```

Then use it in your code:

```
import { config, getJson } from "serpapi";
import { HttpsProxyAgent } from "https-proxy-agent";

// Global configuration
config.requestOptions = {
  agent: new HttpsProxyAgent("http://proxy-server:port"),
};

// Or per-request configuration
await getJson({
  engine: "google",
  q: "coffee",
  requestOptions: {
    agent: new HttpsProxyAgent("http://proxy-server:port"),
  },
});
```

## Pagination

Built-in pagination is not supported. Please refer to our pagination examples for a manual approach:

- Pagination example (Node.js >= 7)
- Pagination example (Node.js >= 14)
- Pagination example (Deno)

# Functions

### Table of Contents

## getJson

Get a JSON response based on search parameters.

### Parameters

- `parameters` **object** search query parameters for the engine
- `callback` **fn?** optional callback

### Examples

```
// single call (async/await)
const json = await getJson({ engine: "google", api_key: API_KEY, q: "coffee" });

// single call (callback)
getJson({ engine: "google", api_key: API_KEY, q: "coffee" }, console.log);
```

## getHtml

Get a HTML response based on search parameters.

- Accepts an optional callback.
- Responds with a JSON string if the search request hasn't completed.

### Parameters

- `parameters` **object** search query parameters for the engine
- `callback` **fn?** optional callback

### Examples

```
// async/await
const html = await getHtml({ engine: "google", api_key: API_KEY, q: "coffee" });

// callback
getHtml({ engine: "google", api_key: API_KEY, q: "coffee" }, console.log);
```

## getJsonBySearchId

Get a JSON response given a search ID.

- This search ID can be obtained from the `search_metadata.id` key in the response.
- Typically used together with the `async` parameter.
- Accepts an optional callback.

### Parameters

- `searchId` **string** search ID
- `parameters` **object** (optional, default `{}` )
  - `parameters.api_key` **string?** API key
  - `parameters.timeout` **number?** timeout in milliseconds
- `callback` **fn?** optional callback

### Examples

```
const response = await getJson({
  engine: "google",
  api_key: API_KEY,
  async: true,
  q: "coffee",
});
const { id } = response.search_metadata;
await delay(1000); // wait for the request to be processed.

// async/await
const json = await getJsonBySearchId(id, { api_key: API_KEY });

// callback
getJsonBySearchId(id, { api_key: API_KEY }, console.log);
```

## getHtmlBySearchId

Get a HTML response given a search ID.

- This search ID can be obtained from the `search_metadata.id` key in the response.
- Typically used together with the `async` parameter.
- Accepts an optional callback.
- Responds with a JSON if the search request hasn't completed.

### Parameters

- `searchId` **string** search ID

- `parameters` **object** (optional, default `{}` )

  - `parameters.api_key` **string?** API key
  - `parameters.timeout` **number?** timeout in milliseconds

- `callback` **fn?** optional callback

### Examples

```
const response = await getJson({
  engine: "google",
  api_key: API_KEY,
  async: true,
  q: "coffee",
});
const { id } = response.search_metadata;
await delay(1000); // wait for the request to be processed.

// async/await
const html = await getHtmlBySearchId(id, { api_key: API_KEY });

// callback
getHtmlBySearchId(id, { api_key: API_KEY }, console.log);
```

## getAccount

Get account information of an API key. https://serpapi.com/account-api

### Parameters

- `parameters` **object** (optional, default `{}` )
    - `parameters.api_key` **string?** API key
    - `parameters.timeout` **number?** timeout in milliseconds
- `callback` **fn?** optional callback

### Examples

```
// async/await
const info = await getAccount({ api_key: API_KEY });

// callback
getAccount({ api_key: API_KEY }, console.log);
```

## getLocations

Get supported locations. Does not require an API key. https://serpapi.com/locations-api

### Parameters

- `parameters` **object** (optional, default `{}` )
  - `parameters.q` **string?** query for a location
  - `parameters.limit` **number?** limit on number of locations returned
  - `parameters.timeout` **number?** timeout in milliseconds
- `callback` **fn?** optional callback

### Examples

```
// async/await
const locations = await getLocations({ limit: 3 });

// callback
getLocations({ limit: 3 }, console.log);
```

## Free Plan · 250 searches / month

Get started 🚀

## They trust us

You are in good company. Join them.

## Documentation

- Google Search API
- Google Light Search API
- Google Ads Transparency API
- Google AI Mode API
- Google AI Overview API
- Google Autocomplete API
- Google Events API
- Google Forums API
- Google Finance API
- Google Flights API

## About

- Home
- Search Engine APIs
- Integrations
- Features
- Pricing

## Contact

SerpApi, LLC
5540 N Lamar Blvd #12
Austin, TX 78751

+1 (512) 666-8245

- Google Hotels API
- Google Images API
- Google Images Light API
- Google Immersive Product API
- Google Jobs API
- Google Lens API
- Google Local API
- Google Local Services API
- Google Maps API
- Google Maps Reviews API
- Google News API
- Google News Light API
- Google Patents API
- Google Play Store API
- Google Related Questions API
- Google Reverse Image API
- Google Scholar API
- Google Shopping API
- Google Shopping Light API
- Google Light Fast API
- Google Short Videos API
- Google Travel Explore API

- Google Trends API
- Google Videos API
- Google Videos Light API
- Amazon Search API
- Apple App Store API
- Baidu Search API
- Bing Images API
- Bing Copilot API
- Bing Search API
- DuckDuckGo Search API
- DuckDuckGo Light API
- eBay Search API
- Facebook Profile API
- Naver Search API
- The Home Depot Search API
- Tripadvisor Search API
- Walmart Search API
- Yahoo! Search API
- Yandex Search API
- Yelp Search API
- YouTube Search API
- Extra APIs

- Use cases
- Team
- Careers
- FAQ
- Legal
- Security
- Contact us
- Blog

contact@serpapi.com

Status and Error Codes

## Our Values

We value full transparency and painful honesty both in our internal and external communications. We believe a world with complete and open transparency is a better world.