

Getting Started with Javascript

05/23/2024 5:40 am EDT

Using the Autoaddress Javascript integration is easy, simply follow the steps below.

Setup

Add the following two files to the <head> section of your page. These files reference the JavaScript and CSS necessary for the Autoaddress plugin to work.

You can download the latest versions from our CDN.

```
<!-- Latest compiled and minified CSS -->
<link href="https://integrations.autoaddress.com/jscontrol/autoaddress-style.css" rel="stylesheet"/>
<!-- Latest compiled and minified JavaScript -->
<script src="https://integrations.autoaddress.com/jscontrol/autoaddress.min.js"></script>
```

or

The files are also available from npm

```
npm install @autoaddress.com/aa3-control
```

You can then reference the files from the node modules folder

```
<link href="" rel="stylesheet"></link>
<script src = "node_modules/@autoaddress.com/aa3-control/autoaddress/autoaddress.min.js"/></script>
```

Initialize

Step 1: Decide where you would like to place the plugin, then add an empty <div> in the <body> section of your HTML.

```
<div id="aa-control"></div>
```

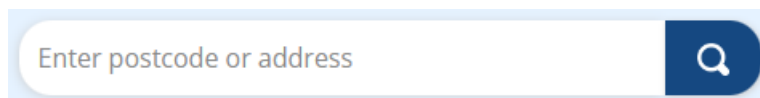
Step 2: Once that's done you initialise the plugin using the snippet below. The snippet below shows a minimal default implementation, which can be configured further according to your needs.

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <!-- Latest compiled and minified CSS -->
    <link href="https://integrations.autoaddress.com/jscontrol/autoaddress-style.css" rel="stylesheet"/>
    <!-- Latest compiled and minified JavaScript -->
    <script src="https://integrations.autoaddress.com/jscontrol/autoaddress.min.js"></script>
  </head>
  <body>
    <div id="aa-control"></div>
    <script>
      var aa = Autoaddress({
        apiKey: 'YOUR_API_KEY',
        elementId: "aa-control"
      });
    </script>
  </body>
</html>

```

You should now see the Autoaddress plugin on your page ready to use.



Notes:

- The code snippet uses an API Key for authentication - you should replace "YOUR_API_KEY" with one of your own API keys. Information on how to create an API Key can be found [here](#).
- The code snippet is initialised using an API Key, for illustration purposes. For greater security, Autoaddress recommends that you initialise the plugin using a Token, created on a back-end server, using the [CreateToken endpoint](#).
- The code snippet uses the default integration type of "Autoaddress form", which replaces your address from with our one line search bar and localized address form. The integration types "Combo" and "Inline" are also available which may be more suitable for your page design. More information around our integration types can be found [here](#).

Customize

Once you have a basic Autoaddress plugin working, you will then want to configure it further so that it is seamlessly integrated with your page design. The Autoaddress plugin is fully configurable and there are a number of ways that developers can use to modify it's behaviour.

1. [Settings](#) – Control the initial display (including integration type) and settings on page load
2. [Callbacks](#) – Hook into lifecycle events of the Autoaddress plugin as your end user quickly enters their address. The most useful callback is "onAddressResult" which is invoked when a user has fully entered an address, but there are other callbacks which give advanced users ultimate control over the end user experience.
3. [Functions](#) – Interact programmatically with the Autoaddress plugin, for example on Form submit.
4. [Styling](#) - Customize the appearance of the Autoaddress plugin to seamlessly blend with the overall design and user experience of your webpage.

