oduct or feature is in a pre-release state and might change or have limited support. For more information, see the pr
stages (/products/#product-launch-stages).

You can enable Stackdriver Trace for Node.js applications by using the Stackdriver Trace library for Node.js.

1. Before installing the Stackdriver Trace library for Node.js, make sure you've prepared your environment for Node.js development (/nodejs/docs/setup).
2. To install the Stackdriver Trace library for Node.js, use npm (https://www.npmjs.com/):

3. Import the Stackdriver Trace library for Node.js at the top of your application's main script or entry point **before any other code**:

View on GitHub (https://github.com/googleapis/cloud-trace-nodejs/blob/master/samples/app.js)

For more information or to report issues with the Stackdriver Trace library for Node.js, see the agent's `cloud-trace-nodejs` GitHub repository (https://github.com/googleapis/cloud-trace-nodejs).

You can customize the behavior of the Stackdriver Trace library for Node.js. See the library's configuration (https://github.com/googleapis/cloud-trace-nodejs/blob/master/src/config.ts) on GitHub for a

list of configuration options that you can pass to the library's `start` method by using an `options` object.

The following example demonstrates specifying the Google Cloud project ID and setting the path to your credential file. These two statements are optional when you're running on Google Cloud:

View on GitHub (https://github.com/googleapis/cloud-trace-nodejs/blob/master/samples/snippets.js)

If you're running on Google Cloud infrastructure, then you don't need to set `projectId` to your Google Cloud project ID. If you don't set this field, the client library for Node.js automatically gathers this data from a Google Cloud metadata server.

If you aren't running on Google Cloud infrastructure, then you must supply your Google Cloud project ID to your application.

Regardless of your infrastructure, for Node.js, when you don't explicitly set the Google Cloud project ID, the `cloud-trace-nodejs` library automatically determines if the environment variable `GCLOUD_PROJECT` is set, and if so, the library uses the value of `GCLOUD_PROJECT` as your Google Cloud project ID. For more information on the discovery file, go to `cloud-trace-nodejs/src/index` (https://github.com/googleapis/cloud-trace-nodejs/tree/master/src/index.ts). To set the environment variable, do the following:

You can use Trace on Google Cloud and when your application runs outside of Google Cloud.

When your application is running on Google Cloud, your application is automatically authenticated and you don't need to provide authentication credentials. However, you do need to ensure that your Google Cloud platform has the Stackdriver Trace API access scope (https://developers.google.com/identity/protocols/googlescopes#cloudtracev2) enabled.

For the following configurations, the default settings for the access scopes have the Stackdriver Trace API enabled:

- App Engine flexible environment
- App Engine standard environment
- Google Kubernetes Engine
- Compute Engine

If you use custom access scopes, then you must ensure that Stackdriver Trace API access scope (https://developers.google.com/identity/protocols/googlescopes#cloudtracev2) enabled. For `gcloud` users, specify access scopes using the `--scopes` flag and include the `trace.append` Stackdriver Trace API access scope. For example, to create a GKE cluster with only the Stackdriver Trace API enabled, do the following:

When your application is running outside of Google Cloud, you must provide authentication credentials in the form of a service account to the client library. The service account must contain the Cloud Trace agent role (/trace/docs/iam#roles). For instructions, see Creating a service account (/iam/docs/creating-managing-service-accounts).

Google Cloud client libraries use Application default credentials (ADC) (/docs/authentication/production) to find your application's credentials. You provide these credentials by setting the `GOOGLE_APPLICATION_CREDENTIALS` environment variable:

View on GitHub (https://github.com/googleapis/cloud-trace-nodejs/blob/master/samples/app.js)

After deployment, you can view the traces in the Cloud Console Trace Viewer.

Go to the Trace Viewer page (https://console.cloud.google.com/traces/overview)

- Node.js client (https://googleapis.dev/nodejs/trace/latest/)
- Source code (https://github.com/googleapis/cloud-trace-nodejs)
- GitHub issue tracker (https://github.com/googleapis/cloud-trace-nodejs/issues)
- Stack Overflow (https://stackoverflow.com/questions/tagged/google-cloud-trace)