You can enable Stackdriver Trace for Java applications by using OpenCensus (https://opencensus.io/). OpenCensus is a set of instrumentation libraries for collecting trace and metric data that work with multiple backends. For the latest details about OpenCensus for Java, along with additional documentation and examples, go to `census-instrumentation/opencensus-java` (https://github.com/census-instrumentation/opencensus-java).

To collect traces, add OpenCensus tracing and the Stackdriver exporter to your application's Maven or Gradle file:

In order to export the collected Trace data, use a `StackdriverTraceExporter` object:

ogleCloudPlatform/java-docs-samples/blob/master/trace/src/main/java/com/example/trace/TraceSample.java)

If you are running on Google Cloud infrastructure, then you don't need to call `setProjectID` and supply your Google Cloud project ID. If you don't set this field, the client library for Java 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, when you don't explicitly set the Google Cloud project ID, the `google-cloud` Java library, which is invoked by OpenCensus, automatically determines if the environment variable `GOOGLE_CLOUD_PROJECT` is set, and if so, the library uses the value of `GOOGLE_CLOUD_PROJECT` as your Google Cloud project ID. For more information, go to `google-cloud-library` specifying a project id (https://github.com/googleapis/google-cloud-java#specifying-a-project-id). To set the environment variable, do the following:

While the OpenCensus library contains automatic integrations for several popular web and RPC frameworks, you can also create custom traces:

ogleCloudPlatform/java-docs-samples/blob/master/trace/src/main/java/com/example/trace/TraceSample.java)

Do not use this option in a production environment. Doing so can generate a large volume of trace data and increase bill.

By default, only 1 in 10,000 traces is sampled (https://github.com/census-instrumentation/opencensus-specs/blob/master/trace/Sampling.md).

In a developer environment, this sampling rate might be too slow to show you trace data. You can use the `alwaysSample` option to sample all traces.

To enable full sampling, use the `setSampler` method and specify the `alwaysSample` option:

ogleCloudPlatform/java-docs-samples/blob/master/trace/src/main/java/com/example/trace/TraceSample.java)

With Java, you can override the automatic authentication and project selection. For example, the following sample illustrates how to create an exporter whose credentials expire 60 seconds from creation time:

ogleCloudPlatform/java-docs-samples/blob/master/trace/src/main/java/com/example/trace/TraceSample.java)

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:

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)

- OpenCensus (https://opencensus.io/)
- GitHub: `census-instrumentation/opencensus-java` (https://github.com/census-instrumentation/opencensus-java)
- API Reference Documentation (https://googleapis.dev/java/google-cloud-clients/latest/)
- Source code (https://github.com/googleapis/google-cloud-java)
- GitHub issue tracker (https://github.com/GoogleCloudPlatform/google-cloud-java/issues)
- Stack Overflow (https://stackoverflow.com/questions/tagged/google-cloud-trace)