This page provides guidance on using Cloud Shell to download and run samples from the GitHub golang-samples/profiler (https://github.com/GoogleCloudPlatform/golang-samples/tree/master/profiler) repository. You can also run these samples outside of Google Cloud. For the additional configuration steps required in this case, see Profiling applications running outside of Google Cloud (/profiler/docs/profiling-external).

If you choose to run these programs, you can quickly create profiling data that you can use to explore the capabilities of the Stackdriver Profiler interface. You can also modify your copy of these samples, then redeploy and analyze the impact of your changes using the Stackdriver Profiler interface.

1. Sign in (https://accounts.google.com/Login) to your Google Account.

   If you don't already have one, sign up for a new account (https://accounts.google.com/SignUp).

2. In the Cloud Console, on the project selector page, select or create a Cloud project.

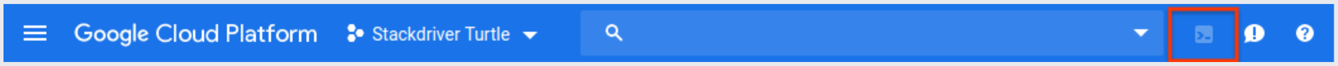
   ★ **Note**: If you don't plan to keep the resources that you create in this procedure, create a project instead of selecting an existing project. After you finish these steps, you can delete the project, removing all resources associated with the project.

   Go to the project selector page (https://console.cloud.google.com/projectselector2/home/dashboard)

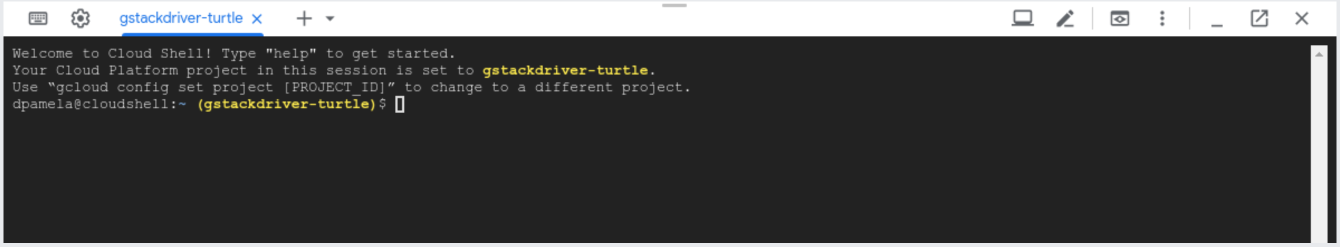
3. Enable the Stackdriver Profiler API.

   Enable the API (https://console.cloud.google.com/flows/enableapi?apiid=cloudprofiler.googleapis.com&redirect=https://console.cloud.google.com)

At the top of the Google Cloud Console page for your project, click **Activate Cloud Shell**:



A Cloud Shell session opens inside a new frame at the bottom of the console and displays two messages and a command-line prompt. The first message lists the Google Cloud project for your Cloud Shell session. The second message tells you how to change the session project. It can take a few seconds for the shell session to be initialized:



```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to gstackdriver-turtle.
Use “gcloud config set project [PROJECT_ID]” to change to a different project.
dpamela@cloudshell:~ (gstackdriver-turtle)$ 
```

From Cloud Shell, retrieve the package of Go samples:

The package retrieval takes a few moments to complete.

The sample **profiler_quickstart** is configured to run the **hello-profiler** service. The Profiler quickstart (/profiler/docs/quickstart) uses this sample.

To start **hello-profiler**, do the following:

1. Change to the **profiler_quickstart** directory:

2. Start the service by running the following command:

The message **profiler has started** is displayed in your Cloud Shell session a few seconds after you start the service. New messages are displayed each time a profile is uploaded to your Google Cloud project.

To stop the service, enter Ctrl-C.

The sample **hotapp** uses an infinite loop that calls two functions, and then the Go scheduler. The sample supports setting the flag **-local_work**. When this flag is **true**, work, in the form of for loops, is added to individual functions. By default, this flag is set to **false**.

To run the **hotapp** sample, do the following:

1. Change to the **hotapp** directory:

2. Start the service by running the following command:

   If you wish to generate profile data that is consistent with data shown in the UI Overview (/profiler/docs/using-profiler) and filtering (/profiler/docs/filtering-profiles) pages, then use the following invocation:

The message `profiler has started` is displayed in your Cloud Shell session a few seconds after you start the service. New messages are displayed each time a profile is uploaded to your Google Cloud project.

To stop the service, enter Ctrl-C.

The section on comparing profiles (/profiler/docs/comparing-profiles) used two different deployments of the `docdemo-service`. In one deployment, the `skew` and `version` were set as previously described. In the second deployment, the `skew` was set to `25` and the `version` was set to `1.25.0`.

If you receive a permission denied error message after starting the service, see Why am I getting a permission denied error? (/profiler/docs/troubleshooting#permission-denied) for possible causes.