Quickstart for Go 1.12+ in the App Engine Standard Environment

Python 2.7/3 (/appengine/docs/standard/python3/ quickstart) | Java 8/11 (/appengine/docs/standard/java11/ quickstart) | PHP 5/7 (/appengine/docs/standard/php7/ quickstart) | Ruby (/appengine/docs/standard/ruby/ quickstart) | Go 1.11 (/appengine/docs/standard/go111/ quickstart) /**1.12+** | Node.js (/appengine/docs/standard/nodejs/ quickstart)

This page shows you how to deploy a basic App Engine application that displays a short message. For a more in-depth introduction to Go on App Engine, see Building an App (/appengine/docs/standard/go/building-app).

Is App Engine the right computing option for you? See other options for hosting your application on Google Cloud (/docs/choosing-a-compute-option).

Costs

There are no costs associated with running this guide. Running this sample app alone does not exceed your free quota (/appengine/quotas).

Before you begin

Before running and deploying this quickstart, install the Cloud SDK and then set up a Cloud project for App Engine:

1. Download and install Cloud SDK:

   Download the SDK (/sdk/docs)

   Note: If you already have the Cloud SDK installed, update it by running the following command:

```
gcloud components update
```

2. Create a new project:

   ```
   gcloud projects create [YOUR_PROJECT_ID] --set-as-default
   ```

   Replace `[YOUR_PROJECT_ID]` with a string of characters (/sdk/gcloud/reference/projects/create#PROJECT_ID) that uniquely identifies your project. For example, `my-project-24`.

   Verify the project was created:

   ```
   gcloud projects describe [YOUR_PROJECT_ID]
   ```

   You see project details that might look like the following:

   ```
   createTime: year-month-hour
   lifecycleState: ACTIVE
   name: project-name
   parent:
   id: '433637338589'
   type: organization
   projectId: project-name-id
   projectNumber: 499227785679
   ```

3. Initialize your App Engine app with your project and choose its region:

   ```
   gcloud app create --project=[YOUR_PROJECT_ID]
   ```

   When prompted, select the region (#before-you-begin) where you want your App Engine application located.

   Caution: You cannot change an app's region once it has been set.

4. Make sure billing is enabled for your project. A billing account needs to be linked to your project in order for the application to be deployed to App Engine.

   Enable billing (https://console.cloud.google.com/projectselector/billing?lang=go&st=true)

Note: Running this sample app does not exceed your free quota. (/appengine/quotas)

5. Install the following prerequisites:

   - Download and install Git (https://git-scm.com/).
   - Run the following command to install the gcloud component (/sdk/docs/managing-components) that includes the App Engine extension for Go 1.12+:

     ```
     gcloud components install app-engine-go
     ```

This quickstart assumes that you are familiar with the Go programming language and that you have installed Go (https://golang.org/doc/install).

Download the Hello World app

We've created a simple Hello World app for Go so you can quickly get a feel for deploying an app to Google Cloud. Follow these steps to download the Hello World app to your local machine.

Download the sample app and navigate into the app directory:

1. Clone the Hello World sample app repository (https://github.com/GoogleCloudPlatform/golang-samples.git):

   ```
   git clone https://github.com/GoogleCloudPlatform/golang-samples.git
   ```

Alternatively, download (https://github.com/GoogleCloudPlatform/golang-samples/archive/master.zip) the sample as a zip file and extract it.

2. Change to the directory that contains the sample code:

```
cd golang-samples/appengine/go11x/helloworld
```

Deploying Hello World to App Engine

1. Deploy the Hello World app on App Engine by running the following command from the `golang-samples/appengine/go11x/helloworld` directory:

```
gcloud app deploy
```

2. View the live app at `https://`***`PROJECT_ID`***`.`***`REGION_ID`*** (#appengine-urls)`.r.appspot.com`:

```
gcloud app browse
```

The message **Hello, World!** is now delivered by a web server running on an App Engine instance.

Congratulations! You've successfully deployed a Go web service on App Engine!

Clean up

If you want to continue learning about App Engine, you can keep your Cloud project and follow along with Building an App (/appengine/docs/standard/go/building-app).

Alternatively, you can delete your Cloud project to stop billing for all the resources used within that project.

> **Caution**: Deleting a project has the following effects:
>
> - **Everything in the project is deleted.** If you used an existing project for this tutorial, when you delete it, you also delete any other work you've done in the project.
> - **Custom project IDs are lost.** When you created this project, you might have created a custom project ID that you want to use in the future. To preserve the URLs that use the project ID, such as an `appspot.com` URL, delete selected resources inside the project instead of deleting the whole project.

1. In the Cloud Console, go to the **Manage resources** page.

   Go to the Manage resources page (https://console.cloud.google.com/iam-admin/projects)

2. In the project list, select the project that you want to delete and then click **Delete** 🗑 .
3. In the dialog, type the project ID and then click **Shut down** to delete the project.

What's next

Now that you have experience deploying a web service on App Engine, get a more in-depth introduction to Go on App Engine by reading Building an App (/appengine/docs/standard/go/building-app).

Last updated 2020-08-14 UTC.