Management Tools (https://cloud.google.com/products/management/)
Cloud Shell (https://cloud.google.com/shell/) Documentation

# Configuring Cloud Shell

This page describes the variety of ways you can configure Cloud Shell to customize your experience.

## Configuration files in your Cloud Shell home directory

Cloud Shell provisions 5 GB of free persistent disk storage (https://cloud.google.com/compute/docs/disks/persistent-disks) mounted as your `$HOME` directory. All files you store in your home directory, including installed software, scripts and user configuration files like `.bashrc` and `.vimrc`, persist between sessions. Since your `.bashrc` persists across sessions, it's a great way to customize your Cloud Shell behavior. Similarly, you can install packages into your home directory to have your installations persist.

## Pre-configured environment variables

When Cloud Shell is started, the active project in Console is propagated to your `gcloud` configuration inside Cloud Shell for immediate use. `GOOGLE_CLOUD_PROJECT`, the environmental variable used by Application Default Credentials library support to define project ID, is also set to point to the active project in Console.

## gcloud command-line tool preferences

To set the project you'd like to work on, run:

```
gcloud config set project [PROJECT_NAME]
```

Your command prompt will be updated to reflect your currently active project and will respect this format: `[USERNAME]@cloudshell:~ ([PROJECT_NAME])$`

To select your preferred region, run:

```
gcloud config set compute/region us-east1
```

The commands above are examples of gcloud command-line tool configurations you can set; for a full list and greater detail about setting properties, refer to the Cloud SDK properties guide (https://cloud.google.com/sdk/docs/properties).

## gcloud command-line tool credentials

You are automatically authenticated with the gcloud command-line tool (and other GCP tools) in Cloud Shell with your current logged-in user. This can be verified by running `gcloud auth list`.

To authenticate as a different user, run `gcloud auth login`.

## Environment customization

**Note:** Personal custom Cloud Shell images are deprecated. Instead, use the environment customization script (https://cloud.google.com/shell/docs/configuring-cloud-shell#environment_customization_script) to add configuration to Cloud Shell that will persist across sessions. For working with public custom images that create temporary environments, use the Open in Cloud Shell (https://cloud.google.com/shell/docs/open-in-cloud-shell.md) feature.

**Note:** If you would like to see a specific package included with Cloud Shell by default, submit your suggestion as feedback (https://cloud.google.com/shell/docs/getting-support#file_bugs_or_feature_requests).

Environment customization allows you to install additional packages into your Cloud Shell environment when it starts.

### Environment customization script

Cloud Shell automatically runs the script, `$HOME/.customize_environment`, when your instance boots up. Unlike `.profile` or `.bashrc`, this script runs once when Cloud Shell boots (rather than once for each shell login).

This script runs as root and you can install any packages that you want to exist in each Cloud Shell session using Debian package management commands.

For example, if you'd like to have erlang installed on Cloud Shell, your `.customize_environment` file will look like this:

```
#!/bin/sh
apt-get update
apt-get -y install erlang
```

Execution logs of your `.customize_environment` script can be found at `/var/log/customize_environment`. The `.customize_environment` script runs as a background process and on successful execution, will touch `/google/devshell/customize_environment_done`. Because package installation runs in parallel with your logging in, the installed packages may become available a few moments after you reach the login prompt.

## tmux support

Cloud Shell uses `tmux` by default, which allows it to improve persistence across browser tab sessions. For example, if you refresh the Cloud Console in a tab or connect to your Cloud Shell from a different machine, the session state will not be lost.

Cloud Shell supports the default `tmux` key bindings (http://man.openbsd.org/OpenBSD-current/man1/tmux.1). For example, if you press **Ctrl+b** and then **%**, `tmux` splits the current session window into left and right panes, which can be useful for debugging.

To disable `tmux` in Cloud Shell, click the **Terminal Settings** button (wrench icon), select **Tmux Settings** and de-select the **Enable Tmux Integration** option.

**Note:** Cloud Shell does not support nested `tmux` sessions. If you try to start `tmux` in a Cloud Shell session with `tmux` already enabled, you will see the error `Sessions should be nested with care, unset $TMUX to force`. If you want to run `tmux` in a custom way, disable the default behavior as described above, restart your Cloud Shell session and then run the `tmux` command manually.

## Terminal preferences

You can customize your Cloud Shell terminal experience by using the **Terminal Settings** menu (wrench icon) and selecting your preferences for color theme, text size, font type, and copy, keyboard and scrollbar default settings.

### Tab title customization

Cloud Shell understands xterm escape sequences for setting tab titles (http://tldp.org/HOWTO/Xterm-Title-4.html#ss4.3). To rename a tab title, set the `PS1` environment variable in your .bashrc to your desired configuration and source your .bashrc for the change to take.

### Keyboard layout support

If you're using an international keyboard or would like to customize your key bindings, Cloud Shell allows you to specify which key behaves as Alt Gr (your modifier key). To specify key mapping, go to the **Terminal Settings** menu, select **Terminal Preferences** > **Keyboard** > **Alt Gr Key** and choose one of the following options:

- Auto - Autodetect based on browser language
- None - Disable any AltGr related munging
- Ctrl-Alt - Set Ctrl+Alt as AltGr
- Left-Alt - Set left Alt as AltGr
- Right-Alt - Set right Alt as AltGr

Additionally, you can set 'Alt is meta' to specify that your alt key should behave as a meta key.

## Usage statistics

Cloud Shell collects anonymized usage statistics on commands that are run inside the Cloud Shell terminal. Statistics are collected only on commands that come pre-installed in the Cloud Shell. Additionally, these metrics cannot be tied back to your account.

Metrics aren't collected on any personally identifiable information, such as arguments passed to these commands.

To change metrics collection preferences at any time, select **Terminal Usage Statistics** in the **Settings** menu  . Any change in your preferences on the Settings menu takes effect on the next session. You'll have to close your tab and open another one once you've made your change to enable/disable metrics collection.

**Note:** This is separate from gcloud usage statistics (https://cloud.google.com/sdk/usage-statistics), which is enabled by default in Cloud Shell.

To disable gcloud command-line tool metrics collection, run the following command in your Cloud Shell session:

```
gcloud config set disable_usage_reporting true
```



*Last updated November 18, 2019.*