

Git CLI

All docs on Git CLI KBs

- [\[How-To\] How to Commit Files to GitHub from Terminal](#)

[How-To] How to Commit Files to GitHub from Terminal

Purpose

To commit your Ansible files to your GitHub repo from your SSH session, follow these steps to initialize the repository (if not already done), commit your files, and push them to your GitHub repository.

Prerequisites

List of prerequisites:

- Root user or sudo user
- Linux LXC or VM

Instructions

1. Set Up Your GitHub Repo (If You Haven't Already)

Make sure you've created a repository on GitHub (e.g., `my_dev`). Copy the repository URL (either HTTPS or SSH, depending on your preferred method).

2. Initialize Git in Your Local Directory (If Not Done)

If your Ansible project folder isn't already a Git repository, initialize it:

```
cd /path/to/your/ansible/files # Navigate to your Ansible files directory git init # Initialize the Git repository
```

3. Add Your GitHub Repository as a Remote

Add the GitHub repo as the remote for your local repository:

```
# Replace the URL with your GitHub repo's URL (HTTPS or SSH)
git remote add origin https://github.com/your-username/my_dev.git
```

If you prefer SSH (recommended for regular usage), use this URL format:

```
git remote add origin git@github.com:your-username/my_dev.git
```

4. Check Out the `develop` Branch

To start working in `develop`, create and switch to the branch if it doesn't exist locally:

```
git checkout -b develop
```

If `develop` already exists in your GitHub repo, pull it down and switch to it:

```
git fetch origin develop
git checkout develop
```

5. Stage and Commit Your Changes

Add all files in the current directory and commit them:

```
git add .
# Stages all changes
git commit -m "Initial commit of Ansible files"
# Commits the changes with a message
```

6. Push to the `develop` Branch on GitHub

Push your commit to the `develop` branch on GitHub:

```
git push -u origin develop
# Pushes and sets up tracking for the 'develop' branch
```