

## Creating a Git Repository

### Lab overview

GitHub runs an instance of git, which is version-control software that runs in the cloud. GitHub is popular among open source projects and businesses.

In this lab, you will:

- Download solutions for the earlier labs
- Create a GitHub account
- Read the GitHub *Hello World* guide
- Create a private repository for your labs
- Download your repository

### Estimated completion time

45 minutes

## Accessing the AWS Cloud9 IDE

1. Start your lab environment by going to the top of these instructions and choosing **Start Lab**.

A **Start Lab** panel opens, displaying the lab status.

2. Wait until you see the message *Lab status: ready*, and then close the **Start Lab** panel by choosing the X.

Activate Windows

Go to Settings to activate Windows.

3. At the top of these instructions, choose **AWS**.

The AWS Management Console opens in a new browser tab. The system automatically logs you in.

**Note:** If a new browser tab does not open, a banner or icon at the top of your browser typically indicates that your browser is preventing the site from opening pop-up windows. Choose the banner or icon, and choose **Allow pop ups**.

4. In the AWS Management Console, choose **Services > Cloud9**. In the **Your environments** panel, locate the **reStart-python-cloud9** card, and choose **Open IDE**.

The AWS Cloud9 environment opens.

**Note:** If a pop-up window opens with the message *.c9/project.settings have been changed on disk*, choose **Discard** to ignore it. Likewise, if a dialog window prompts you to *Show third-party content*, choose **No** to decline.

## Exercise 1: Downloading your Python files from the previous labs

5. From the menu bar, choose **File > Download Project**.

This action creates an untitled file.

6. This action downloads a compressed file onto your local machine. Extract the contents of this file.

## Exercise 2: Creating a GitHub account

At the time of this writing, GitHub offered a free account for individuals.

Activate Windows  
Go to Settings to activate Windows.

7. Visit **GitHub** at <https://www.github.com> and create an account.

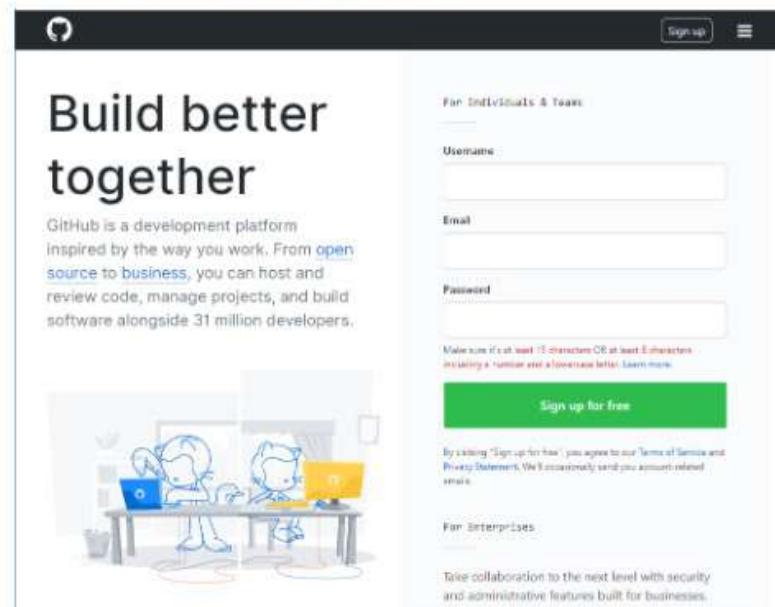
This action creates an untitled file.

6. This action downloads a compressed file onto your local machine. Extract the contents of this file.

## Exercise 2: Creating a GitHub account

At the time of this writing, GitHub offered a free account for individuals.

7. Visit [GitHub](https://www.github.com) at <https://www.github.com> and create an account.



Activate Windows  
Go to Settings to activate Windows.



Take collaboration to the next level with security and administrative features built for businesses.

## Exercise 3: Reading the GitHub Hello World Guide

After you log in to GitHub, you can access to the *Hello World* guide for creating a repository.

The screenshot shows a light blue background with a central white box. At the top of the box is a green header bar with the text "Learn Git and GitHub without any code!" in white. Below the header, there is a brief description: "Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request." At the bottom of the box are two buttons: a green "Read the guide" button and a white "Start a project" button with black text.

At the time of this writing, the GitHub Guides page hosted the *Hello World* guide at <https://guides.github.com/activities/hello-world/>.

8. Read the GitHub *Hello World* guide.

Activate Windows  
Go to Settings to activate Windows.

## Exercise 4: Creating a private repository

9. After you read the GitHub *Hello World* guide, make sure that you are logged in to GitHub and choose the **New** button.

The **Create a new repository form** (see the screen capture) should open.

10. Give your repository a name, such as *aws\_restart*.

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere?  
[Import a repository.](#)

Owner  Repository name \*  

Great repository names are short and memorable. Need inspiration? How about [potential-engine](#)?

#### Description (optional)

#### Public

Anyone can see this repository. You choose who can commit.

#### Private

You choose who can see and commit to this repository.

#### Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

[Add .gitignore: None](#)

[Add a license: None](#)



**Create repository**

**Note:** You can make your repository public or private. Choose to create a **private** repository. Also, select the **Initialize this repository with a README** option.

Activate Windows  
Go to Settings to activate Windows.

**Note:** You can make your repository public or private. Choose to create a **private** repository. Also, select the **Initialize this repository with a README** option.

If your repository is created successfully, you should see a default repository with a README file that is similar to this example.

A screenshot of a GitHub repository page. The repository name is 'dguevar1/aws\_restart'. It is a private repository with 1 commit, 1 branch, and 0 releases. There are no issues, pull requests, or projects. The README file contains the text 'aws\_restart'. The page includes standard GitHub navigation and repository details.

Choose the **Upload files** button to get to the **Upload files** page.

A screenshot of the 'Upload files' page on GitHub. It shows a large text input field with the placeholder 'Drag additional files here to add them to your repository' and a smaller 'Or choose your files' button below it. Two files are listed in the file list: 'while\_loops\_solution.py' and 'list\_tuples\_dictionary\_solution.py'. Below the file list is a navigation bar with numbers 1 through 17, where number 9 is highlighted.

Activate Windows  
Go to Settings to activate Windows.

The screenshot shows a GitHub commit dialog. At the top, there's a placeholder text "Drag additional files here to add them to your repository" and a link "Or choose your files". Below this is a list of files with delete icons:

- while\_loop\_solution.py
- list\_tuple\_dictionary\_solution.py
- for\_loop\_solution.py
- string\_data\_type\_solution.py
- composite\_data\_types\_solution.py
- unzipper\_values\_solution.py
- conditional\_solution.py
- hello\_world\_solution.py
- numeric\_data\_types\_solution.py

The main dialog area has two input fields: "Add files via upload" and "Add an optional extended description...". Below these is a radio button group:  
 Commit directly to the `main` branch.  
 Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

At the bottom are two buttons: "Commit changes" and "Cancel".

11. Upload all the files that you previously extracted in Exercise 1.

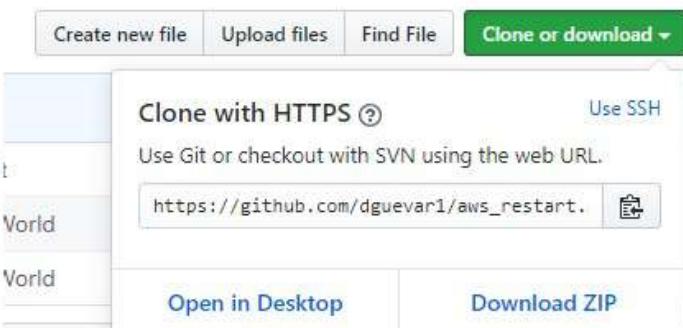
## Exercise 5: Downloading a repository

To download your repository, complete the following steps.

12. Choose the **Clone or download** button.
13. Select the **Download Zip** option.

Activate Windows  
Go to Settings to activate Windows.

13. Select the **Download Zip** option.



14. On your local machine, create an **aws\_restart** folder and save your .zip file to it.

15. To verify that the files were downloaded, extract the .zip file.

Congratulations! You have used some basic features of GitHub.

## End Lab

☒ Congratulations! You have completed the lab.

16. Choose ■ **End Lab** at the top of this page, and then select Yes to confirm that you want to end the lab.

A panel indicates that *DELETE has been initiated... You may close this message box now.*

17. A message *Ended AWS Lab Successfully* is briefly displayed, indicating that the lab has ended.

Activate Windows  
Go to Settings to activate Windows.