



Setup: Visual Studio Code Server

i This lab is offered as an **AWS-hosted lab** meaning AWS personnel will supply a temporary AWS account for running the IDE and the other required components. You supply the computer with a web browser, and the ability to connect to the internet. You also have the option of running the lab using your own AWS account. If running the lab on your own system, it is recommended that you install and run the [Visual Studio Code](#) IDE locally to complete the workflows.


To get started with the lab as quickly as possible, we will be using [Visual Studio Code Server](#) (VS Code Server) running on an Amazon EC2 Instance inside the AWS account. This means you do not need to install anything on your local computer, such as VS Code, NodeJS, or other software utilities.

Accessing Visual Studio Code Server

1. Open the [Workshop Studio event dashboard](#).
2. Navigate to the **Event Outputs** pane at the bottom of the page.
3. Copy the **Password** string. You will need this password in the next step.
4. Select the **URL** to open Visual Studio Code Server.

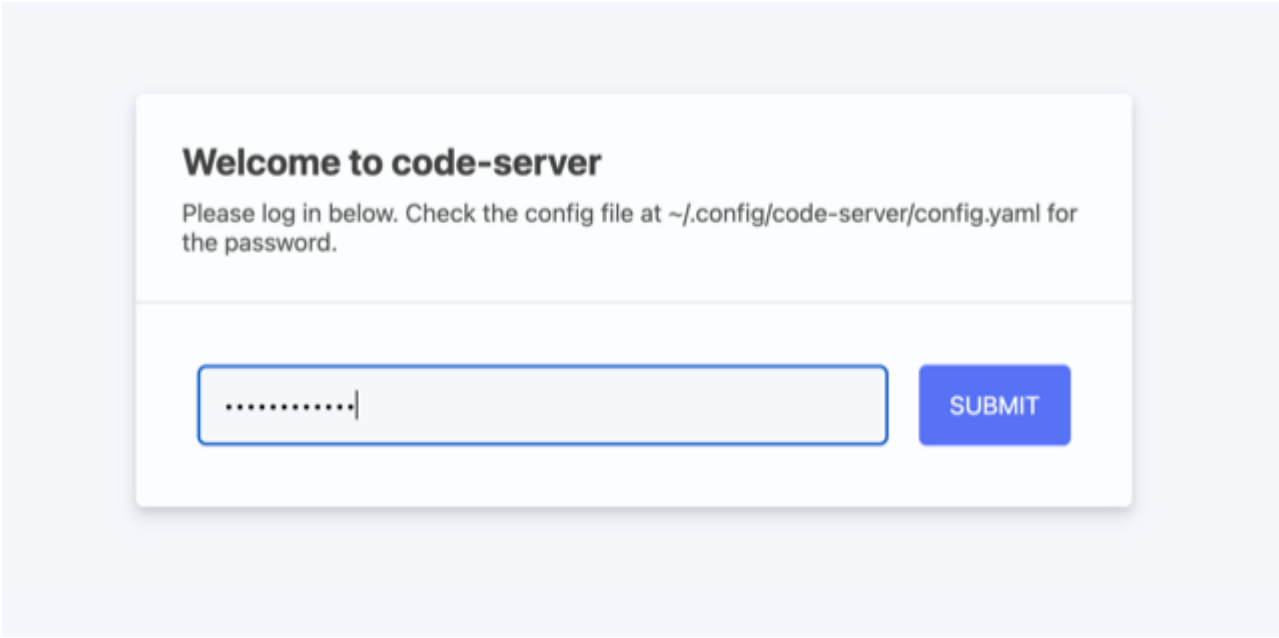
Event Outputs (2)

Q Search

< 1 > 

Key	Value	Stack name	Description	Type
vscode-server-Password	958137408447	vscode-server	VSCode-Server Password	export
vscode-server-URL	https://drelvckgb6vza.cloudfront.net/?folder=/workshop	vscode-server	VSCode-Server URL	export

5. In the **Welcome to code-server** dialog, paste the password that you copied earlier, and click **Submit**.




You should now see the Visual Studio Code IDE, as shown below.

EXPLORE

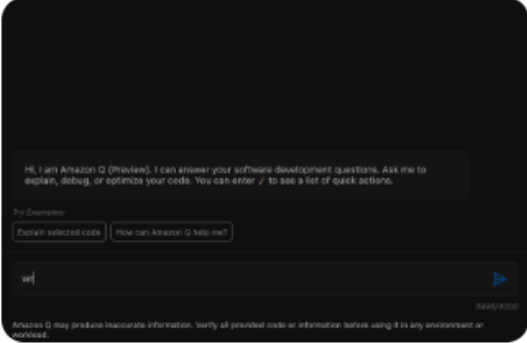
WORKSHOP

AWS Toolkit: Add Connection to AWS

 AWS Toolkit for VS Code
Documentation | Join us on Github | Share Feedback

Select a sign-in option to continue:

Amazon Q + CodeWhisperer

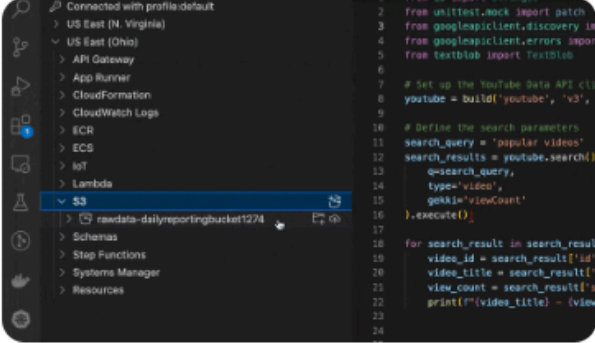


Build, maintain, and transform applications using generative AI.
Learn more about [Amazon Q](#) and [CodeWhisperer](#).

Use for free, no AWS Account required

Use Professional License

Resource Explorer



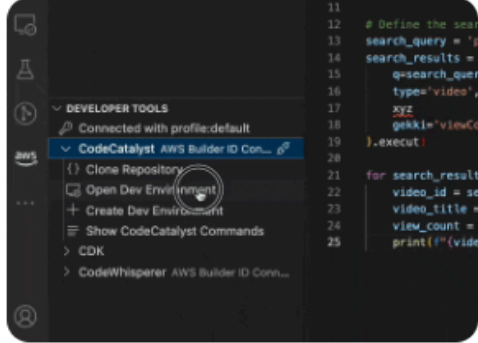
Work with S3, CloudWatch, and more. [Learn more.](#)

Use Single Sign-on (SSO)

Or, provide IAM Roles Credentials

Don't have an AWS account? [Sign up for free.](#)

Amazon CodeCatalyst



Spend more time coding and less time managing development environments. [Learn more.](#)

Use for free with AWS Builder ID

Use Single Sign-on (SSO)

Don't have a CodeCatalyst account? [Get started with creating a Space.](#)

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

PORTS

CODE REFERENCE LOG

ubuntu@dev: /Workshop\$

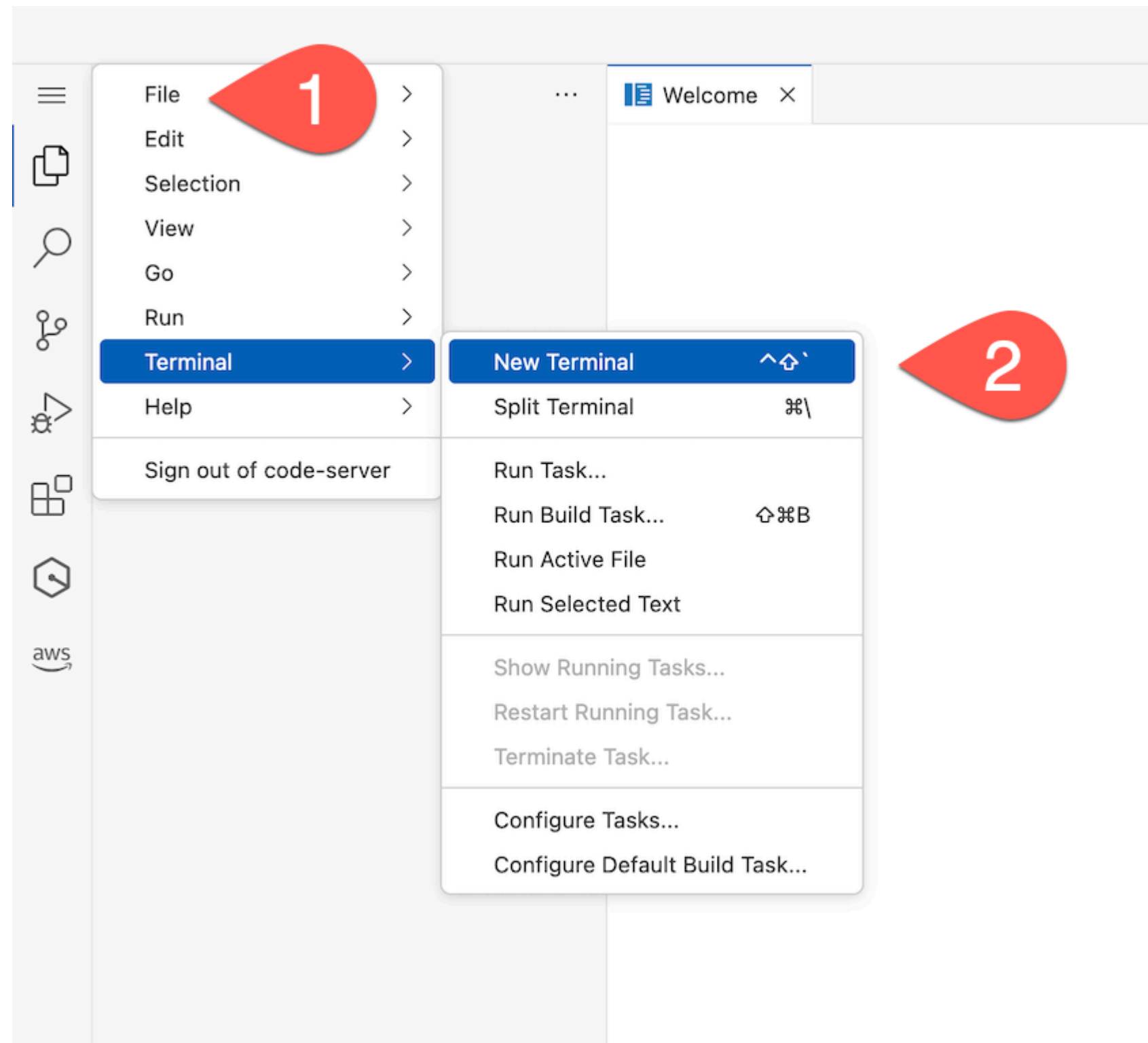
Visual Studio Code Server installed components

The following items have already been installed on your VS Code Server instance, so you don't need to install them locally or on the EC2 instance.

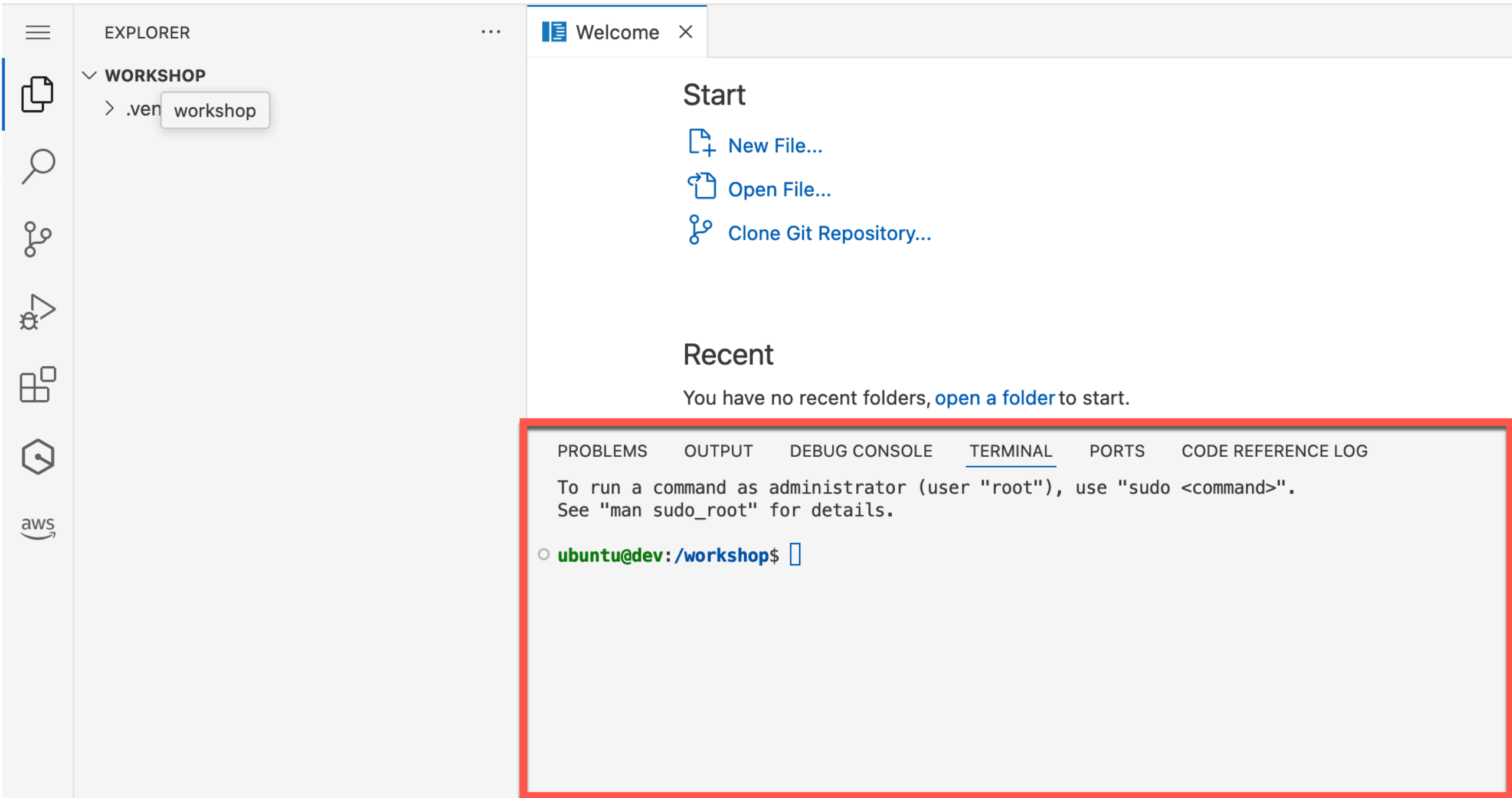
- [Node](#).
- [Python](#).

VSCode also includes a terminal utility that will be used throughout the lab workflows. To enable the terminal, from the upper-left corner of the VSCode interface:

1. Click "File" > "Terminal" > "New Terminal"



This will open up a terminal session at the bottom of VSCode. Whenever the lab instructs you to enter commands from the terminal, you can enter them at the available prompt.



Conclusion of workflow

What did you do in this workflow?

In this introductory workstream, we successfully connected to the VSCode server created on an EC2 instance. This will enable us to leverage an integrated development environment (or IDE) for creating files, and using a terminal interface for executing commands.

[Previous](#)

[Next](#)