

Hands-on Lab: Getting Started with the Linux Terminal



Estimated time needed: **15** minutes

Learning Objectives

After completing this lab, you will be able to:

- Interact with the Linux terminal
- Browse directories on a Linux filesystem using the `ls` command
- Navigate directories using the `cd` command
- Save time and effort by using tab completion and your command history

About Skills Network Cloud IDE

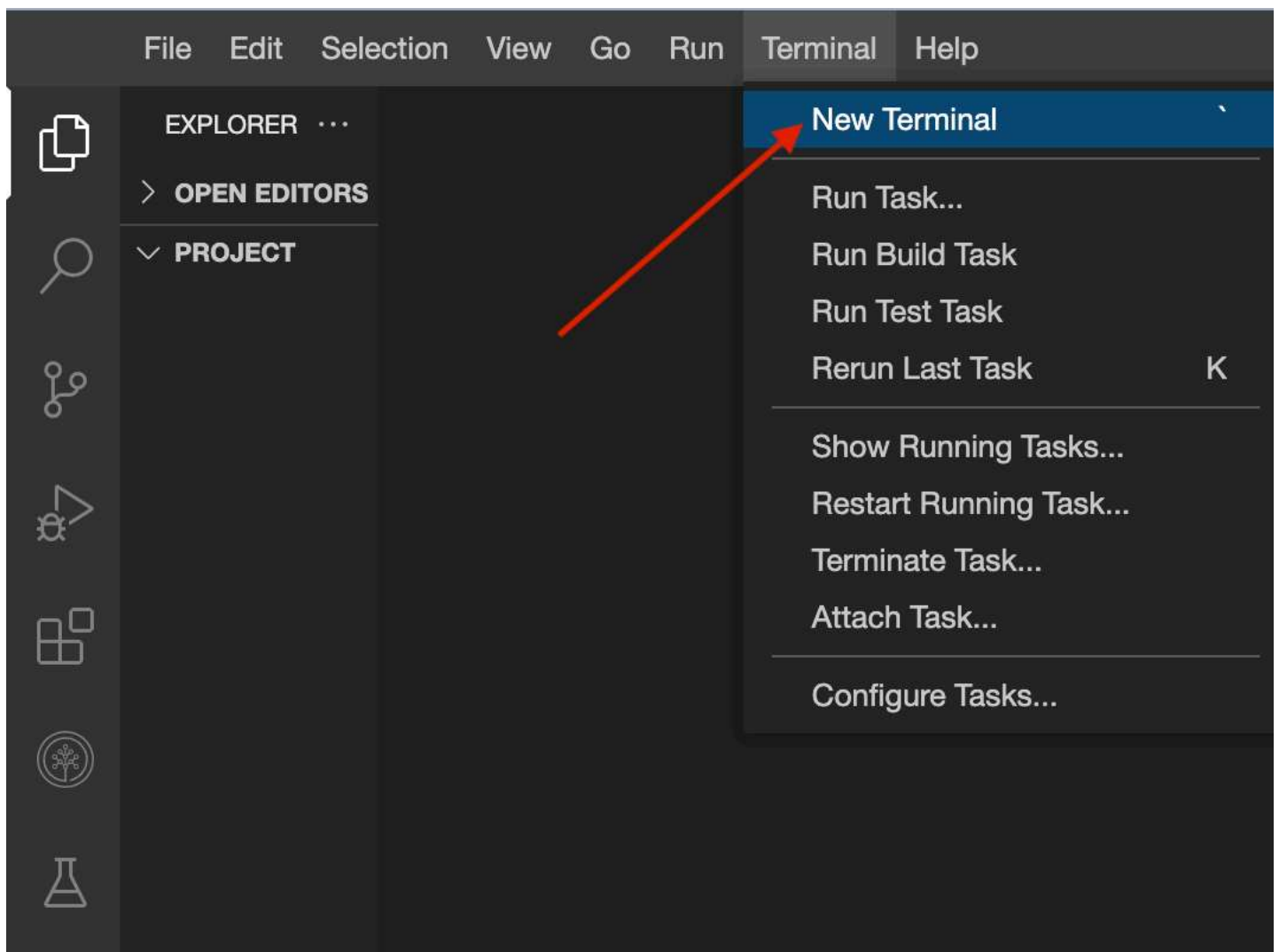
Skills Network Cloud IDE (based on Theia and Docker) provides an environment for hands-on labs for course and project-related labs. Theia is an open source Integrated Development Environment (IDE) that can be run on the desktop or on the cloud. To complete this lab, you will be using the Cloud IDE based on Theia.

Important Notice About This Lab Environment

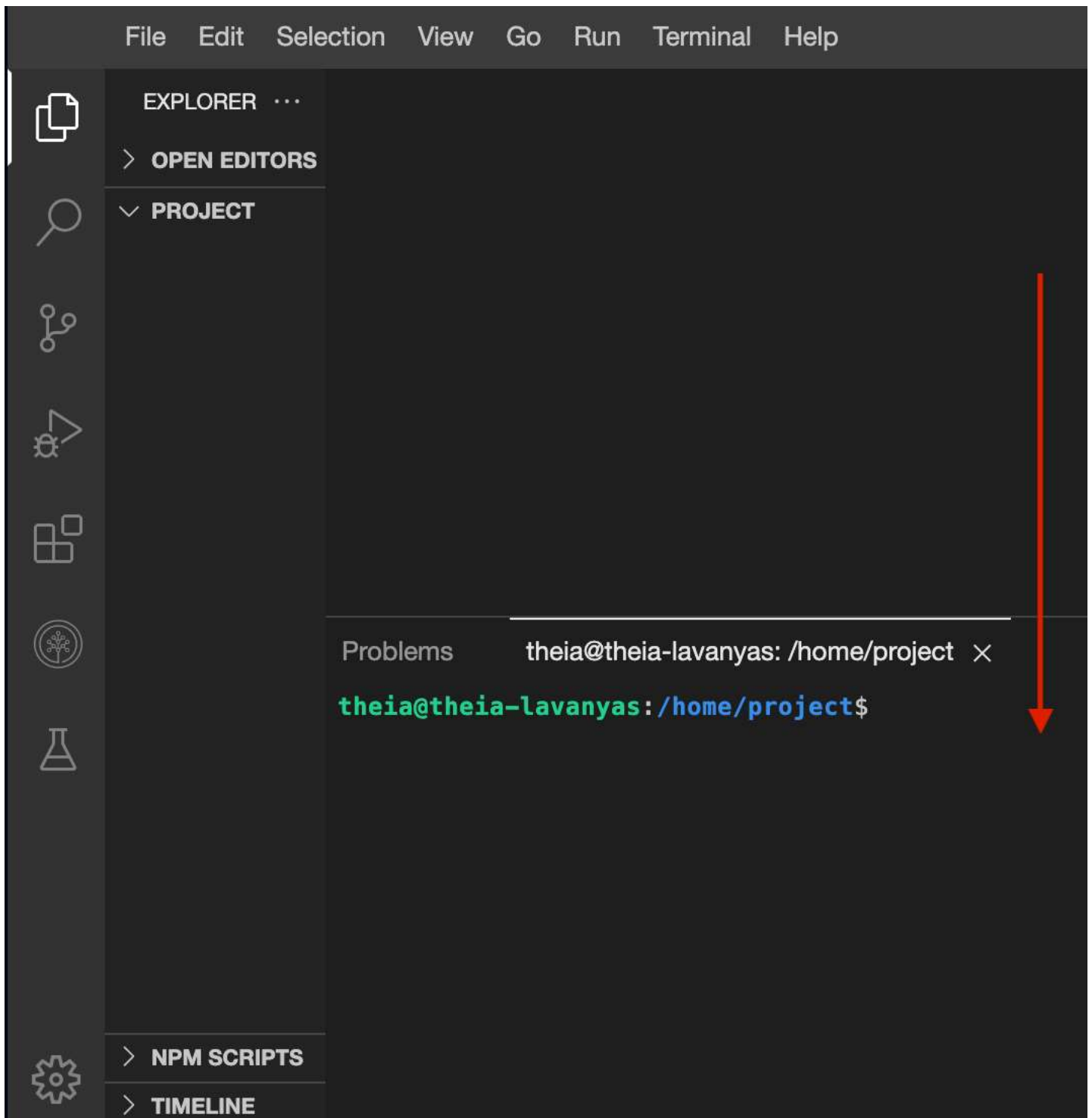
Please be aware that sessions for this lab environment are not persistent. Thus, every time you connect to this lab, a new environment is created for you, and any data or files you may have saved in a previous session will be lost. To avoid losing your data, plan to complete these labs in a single session.

Opening a terminal

When you first connect to the lab environment, there may already be an open terminal for you to use. If not, open a new terminal by clicking the menu bar and selecting **Terminal > New Terminal**.



This will open a new terminal at the bottom of the screen.



In your newly opened terminal, you will run the commands provided in the following exercises. You can copy the code to your clipboard by clicking the copy button on the bottom right of each codeblock, then pasting it to the command line. However, it will be more beneficial to type the code yourself, so you can start building some "command muscle memory".

Exercise 1 - Browsing Directories

1s

In this exercise, you will learn how to browse the content of directories using the `1s` command.

`1s` is a special command that the shell interprets and runs as a program.

1.1. Viewing files in the current working directory

Typing and entering `1s` will display all files and directories contained within your current working directory:

```
1s
```

Your default current working directory in this lab environment, the `/home/project` directory, is empty. Thus entering `ls` into the command line will not return anything yet.

1.2. Viewing files and directories within any directory

If you know the path to a directory, you can view its contents by passing the path name as a *command line argument* to the `ls` command as follows:

```
ls [PATH TO DIRECTORY]
```

For example:

```
ls /
```

will show the contents of 'slash', your Linux system's root directory.

Recall some of the standard subdirectories of 'slash' that you've learned about previously:

Directory	Contains
/bin	System commands, also called <i>binaries</i>
/sbin	System administration binaries
/usr	User programs and data
/home	Home directory
/media	Removable media device directories

For example, entering the following command:

```
ls /bin
```

will display the contents of the `/bin` directory.

Notice that one of the files in the `/bin` directory is called "`ls`". Indeed, when you enter the `ls` command, Linux searches for and runs the `ls` command by executing the binary file `/bin/ls`.

Exercise 2 - Navigating Directories

```
cd
```

In this exercise, you will explore directories using the `cd` command.

Recall the symbols used to navigate to special paths:

Symbol	Stands for path to
~	Home directory
/	Root directory
.	Current directory
..	Parent directory

2.1. Changing your present working directory to your home directory

To change your current working directory to your default home directory, use the `~` symbol as follows:

```
cd ~
```

Note that your lab environment's default home directory, `~`, is `/home/theia`. After changing directories to `~`, you can check this yourself by entering the `pwd` command, which displays the path to your current working directory.

2.2. Changing your present working directory to its parent directory

To change your current working directory to the *parent* directory of your present working directory, enter the following command using `..`:

```
cd ..
```

For example, if your working directory was `/home/theia`, then the new working directory will be `/home`.

2.3. Changing working directory to root directory

To change your current working directory to the root directory, use the following command with the `/` key:

```
cd /
```

2.4. Changing your present working directory to a child directory

Assuming you are still in your root directory, use the following command to change your present working directory to the `/bin` directory:

```
cd bin
```

The `/bin` directory is called a *child* of the root directory, `/`, because it's a subdirectory within it.

The `cd` command understands implicitly that `bin` is the name of a subdirectory of your current working directory. As a result, it deduces that its path is a combination of the current working directory path (`/`) and the subdirectory name (`bin`).

Another method to change directories is to enter the command `cd ./bin`. In this case, `.` represents the path to your current working directory.

Give it a try by entering `cd ./bin` in your command line! What directory did you end up in?

Tip: If `cd ./bin` doesn't take you to the `/bin` directory, double-check what directory you are currently in. Remember, `.` represents the directory you are in. If you're already in `/bin`, the terminal will attempt to direct you to `/bin/bin`.

2.5. Changing from your working directory back to your home directory

There are multiple ways to change your current working directory back to your default home directory. One way is to return to your parent directory, `..`, and type the address of your home directory `/home/theia`:

```
cd ../home/theia
```

A simpler way to do this would be to use the `~` symbol to quickly and directly navigate to your home directory:

```
cd ~
```

Both methods will change your working directory back to your home directory.

2.6. Changing from your working directory to your project directory

Directories that are contained within the same parent directory are called **siblings**.

In this environment, we have provided a special empty project directory for your work. This project directory is located at `/home/project` and is a sibling directory to `/home/theia`.

Change your current working directory to its sibling directory, your project directory, by entering the following command:

```
cd ../project
```

Exercise 3 - Using tab completion and the command history

In this exercise, you will gain familiarity with tab completion and the command history.

3.1. Scrolling through your command history

At the command prompt, simply hit the Up Arrow and Down Arrow keys, \uparrow and \downarrow , to scroll through your command history.

Assuming you are still in the same session as you were when you completed the previous exercises in this lab, you should see exactly what commands you have entered so far.

For example, suppose you want to use the `cd` command to navigate to `/bin` again. Rather than retyping the command, simply press the Up Arrow key until you see the `cd bin` displayed at the command prompt and press Enter.

Of course, this command will only work as expected if your current working directory is the correct one. If the command fails, you can keep scrolling through your history until you find the `cd` command to return to your root directory, (`cd /`). Enter that command first, then use the Up Arrow key to find and enter the `cd bin` command from your history.

Your command history is very useful if you are doing repetitive tasks and want to avoid retyping things repeatedly.

3.2. Using tab completion

Tab completion can save you a lot of typing, thinking, and time. Let's take a look at some examples.

Type `cd /bi` and press the Tab key. As you might expect, the command autocompletes to `cd /bin`. But what happens if you just type `cd /b` and press Tab?

In this case, nothing happens. But if you hit Tab a second time, you'll see a list of all the valid candidates for autocompletion, namely `bin/` and `boot/`.

Notice that `cd /b` reappears in the command prompt. You cannot use tab completion while there are multiple valid candidates for autocompletion. To narrow down your choices, add an `i` at the end of your line, then press Tab again. This time the prompt will fill with `cd /bin`, the only valid option remaining.

Tip: Tab completion only works when there is **one** valid candidate for autocompletion. When there are multiple possible options, the command prompt will prompt you to enter more letters until there is only one possible option.

You can even use tab completion repeatedly to "dig into" a directory tree.

Let's say you have used tab completion to get to the point where `ls /home` is displayed at the command prompt. You can press Tab twice to get a listing of all the directories under `/home`. As you did with `cd /bin`, use tab completion to complete the address of a subdirectory such as `ls /home/theia`. You can repeat this process as many times as you like until you reach the end of a chosen path.

As an exercise, practice using tab completion repeatedly to build up the command `ls /home/theia/dsdriver/bin`.

Future labs will cover command options more in-depth. In this lab, you became familiar with the terminal environment and some basic commands.

Practice Exercises

1. List the contents of the root directory.

- [Click here for Hint](#)
- [Click here for Solution](#)

2. Change directories to your default home directory.

- [Click here for Hint](#)
- [Click here for Solution](#)

3. Verify your current working directory is `/home/theia`.

- [Click here for Hint](#)

► [Click here for Solution](#)

4. Use tab completion to change directories to `/bin`.

- [Click here for Hint 1](#)
- [Click here for Hint 2](#)
- [Click here for Solution](#)

5. Use your terminal's command history to change directories back to your home directory.

- [Click here for Hint](#)
- [Click here for Solution](#)

Summary

Congratulations! You've just completed your first hands-on experience with the Linux terminal!

By now, you are getting a taste of just how powerful and fun the command line is. Now that you are up and running with the command line, we have a lot more exciting things for you to explore. We hope that you are as enthusiastic about learning Linux as we are!

Some of the key skills you practiced in this lab include:

- Interacting with the Linux terminal in a modern, cloud-based dev environment
- Using the `ls` and `cd` commands to browse and navigate and explore directories on your Linux system
- Making use of tab completion and your command history to save yourself time and effort

If you ever feel like you might have missed anything important or just want some extra practice, you can always come back and go through this lab again. As always, we encourage you to explore to your heart's content.

Authors

Jeff Grossman
Sam Prokopchuk

Other Contributors

Rav Ahuja

© IBM Corporation. All rights reserved.