Shell Script

Estimated Time Needed: 40 min

In this lab, you will practice creating and executing a simple bash shell scripts to satisfy the requirements given in the problem statement.

Objectives

After completing this lab, you will be able to use Linux CLI to:

- Run commands on CLI to perform some basic operations on files
- Create and run shell scripts for ETL
- Create cron jobs

About Skills Network Cloud IDE

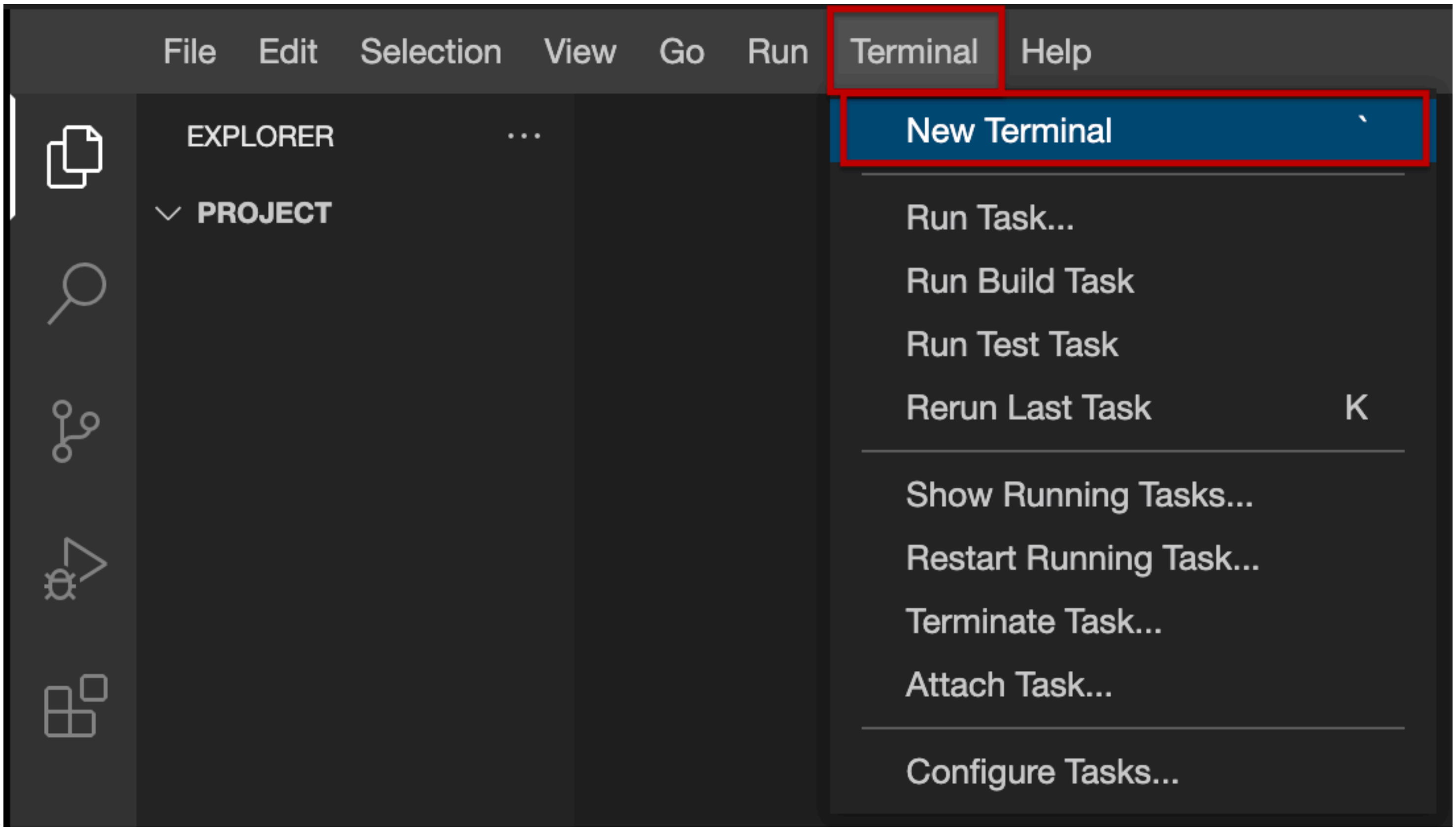
Skills Network Cloud IDE (based on Theia and Docker) provides an environment for hands-on labs for course and project-related labs. Theia is an open-source Integrated Development Environment (IDE) that can be run on the desktop or on in the cloud. To complete this lab, you will be using the Cloud IDE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

Important Notice About This Lab Environment

Please be aware that sessions for this lab environment are not persisted. 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.

Setup Lab environment

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. You can run the commands in your newly opened terminal. You can copy the code to your clipboard by clicking the copy button on the bottom right of each code block, and then pasting it on the command line.



Create the data file to be used

1. On the terminal copy, paste and run the following command to create an empty file named `SampleDataFile.txt`.

```
touch /home/project/SampleDataFile.txt
```

Copied! Executed!

2. Click on the button below to create file named SampleDataFile.txt.

Open **SampleDataFile.txt** in IDE

3. Paste the following content in the file and save it.

```
Username        ID              Orders  City
Emily           2017    3               Phoenix
Luna            3812    4               Louisville
Penelope        9013    1               Seattle
Adrian          1209    2               Portland
Joshua          5122    1               Boston
Everett         8917    2               Boston
Eva                     6390    3               Louisville
Tom                     7190    4               Phoenix
Sadie           2165    4               Seattle
Eva                     6390    3               Louisville
```

Copied!

Shell scripts and commands

1. Display the total number of rows, words and character count in the file

▼ Solution Syntax

```
wc SampleDataFile.txt
```

Copied! Executed!

2. Display the top 15 rows in the file

▼ Solution Syntax

```
head -15 SampleDataFile.txt
```

Copied! Executed!

3. Display the bottom 10 rows in the file

▼ Solution Syntax

```
tail -10 SampleDataFile.txt
```

Copied! Executed!

4. Search and display all rows that mention Phoenix in the input file.

▼ Solution Syntax

```
grep -i Phoenix SampleDataFile.txt
```

Copied! Executed!

5. Display the Username and City columns from the file.

▼ Solution Syntax

```
cat SampleDataFile.txt|cut –f 1,4
```

Copied! Executed!

6. Check for data quality, specifically duplicates in the given file. How many times is a line repeated in the file?

▼ Solution Syntax

```
cat SampleDataFile.txt|cut –f 1|sort|uniq -c
```

Copied! Executed!

7. Extract first three columns to another file and convert the tab delimited to a comma delimited file.

▼ Solution Syntax

- Extracting and redirecting data

```
cat SampleDataFile.txt|cut -f 1-3>NewFile.txt
```

Copied! Executed!

- Transforming data

```
cat NewFile.txt|tr '\t' ','
```

Copied! Executed!

Cron jobs

Cron is a system daemon used to execute desired tasks in the background at designated times. For example, you can schedule to run a cron job each day at 9pm and direct output to another file.

A crontab file is a simple text file containing a list of commands meant to be run at specified times. It is edited using the crontab command.

Script and schedule a crontab file that prints the current time and the current disk usage statistics.

▼ Solution Syntax

Create a new file diskusage.sh and enter following script and save the file

```
#! /bin/bash
# print the current date time

date

# print the disk free statistics

df -h

# Add a job to crontab and save it with the following script
# This job will run the diskusage script at 9.00PM and store the output to diskusage.log

crontab -e

0 21 * * * /home/project/disksusage.sh >>/home/project/diskusage.log
```

Copied! Executed!

Author

Roopa Raghavan

Changelog

Date	Version	Changed by	Change Description
13-Dec-2022	1.0	Lavanya	Initial version created