Week 3-Assignment 2 – Practical work 1 -Task 1

Introducing Linux Terminal

Objectives

In this practical work, you will:

- Interact with the Linux Terminal
- Navigate directories on a Linux filesystem and explore their contents
- Install and update packages
- Create and edit files using `nano`
- Run shell commands and applications from the terminal

If you don't have a Linux terminal running, you can use the Skills Network Cloud IDE

About Skills Network Cloud IDE

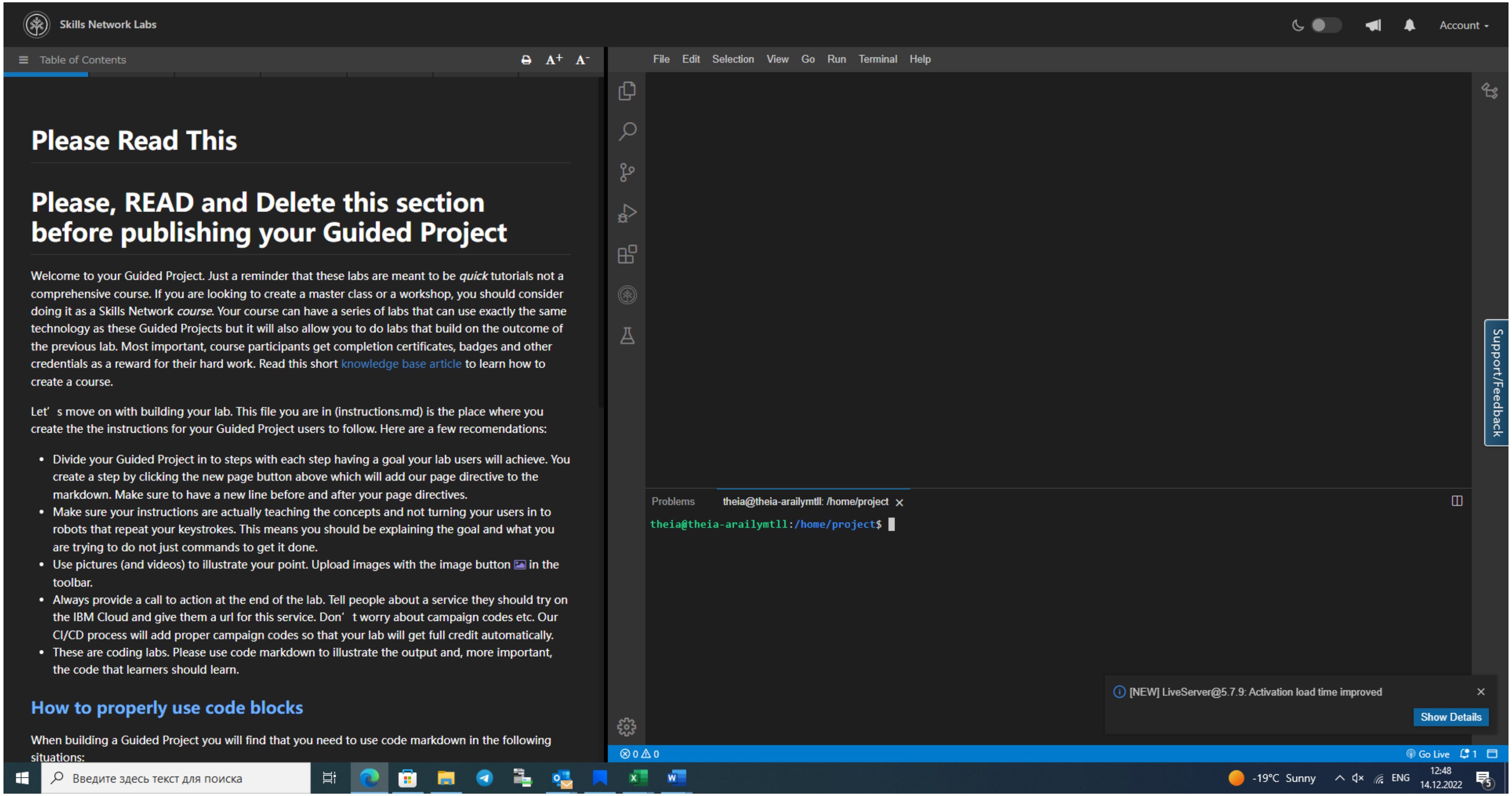
Skills Network Cloud IDE (based on Theia and Docker) provides an environment for hands-on labs for course and project-related labs. Theia is an open source Integrated Development Environment (IDE) that can be run on the desktop or on the cloud. To complete this lab, you will be using the Cloud IDE based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

Important Notice About This Lab Environment

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

Skills Network Authors   create an account

Create a Guided Project



Please Read This

Please, READ and Delete this section before publishing your Guided Project

Welcome to your Guided Project. Just a reminder that these labs are meant to be *quick* tutorials not a comprehensive course. If you are looking to create a master class or a workshop, you should consider doing it as a Skills Network *course*. Your course can have a series of labs that can use exactly the same technology as these Guided Projects but it will also allow you to do labs that build on the outcome of the previous lab. Most important, course participants get completion certificates, badges and other credentials as a reward for their hard work. Read this short knowledge base article to learn how to create a course.

Let' s move on with building your lab. This file you are in (instructions.md) is the place where you create the the instructions for your Guided Project users to follow. Here are a few recomendations:

- Divide your Guided Project in to steps with each step having a goal your lab users will achieve. You create a step by clicking the new page button above which will add our page directive to the markdown. Make sure to have a new line before and after your page directives.
- Make sure your instructions are actually teaching the concepts and not turning your users in to robots that repeat your keystrokes. This means you should be explaining the goal and what you are trying to do not just commands to get it done.
- Use pictures (and videos) to illustrate your point. Upload images with the image button in the toolbar.
- Always provide a call to action at the end of the lab. Tell people about a service they should try on the IBM Cloud and give them a url for this service. Don' t worry about campaign codes etc. Our CI/CD process will add proper campaign codes so that your lab will get full credit automatically.
- These are coding labs. Please use code markdown to illustrate the output and, more important, the code that learners should learn.

How to properly use code blocks

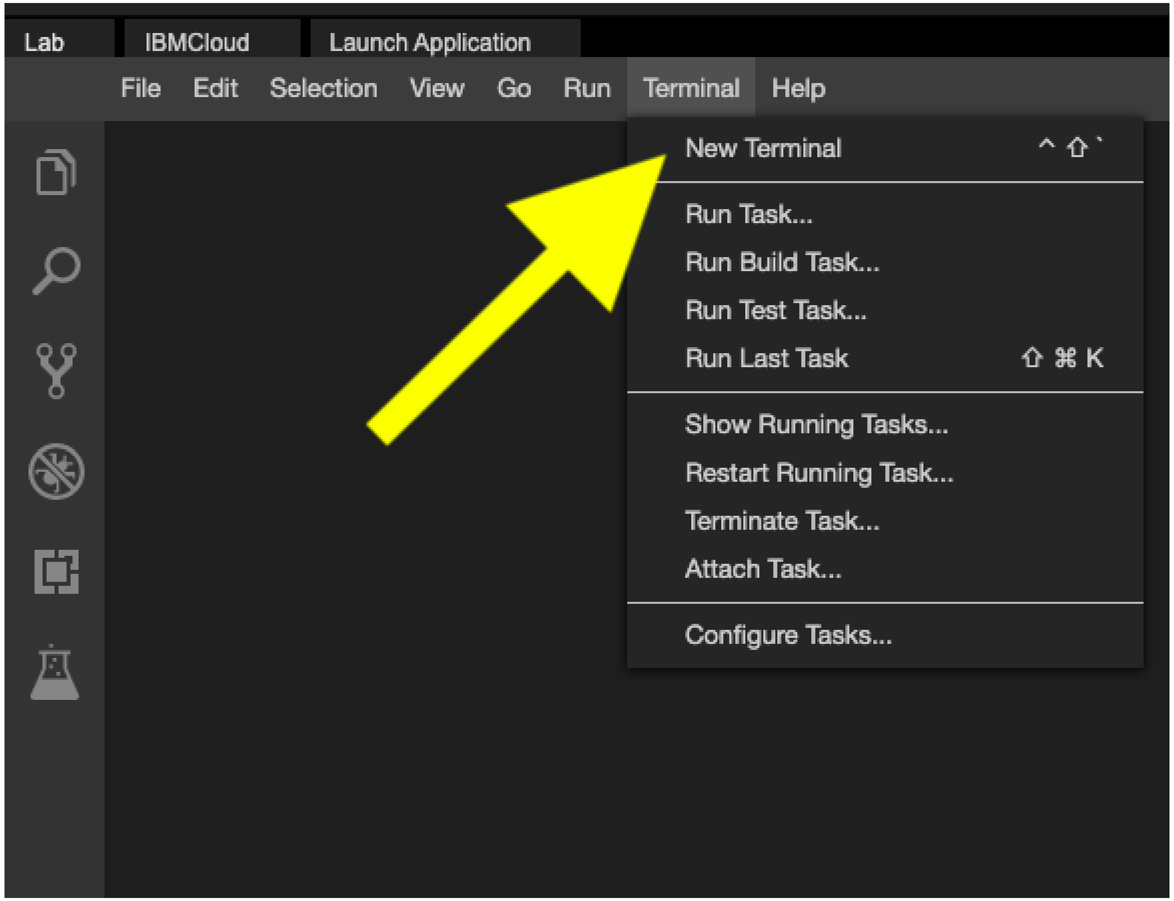
When building a Guided Project you will find that you need to use code markdown in the following situations:

```
theia@theia-arailymtll:/home/project$
```

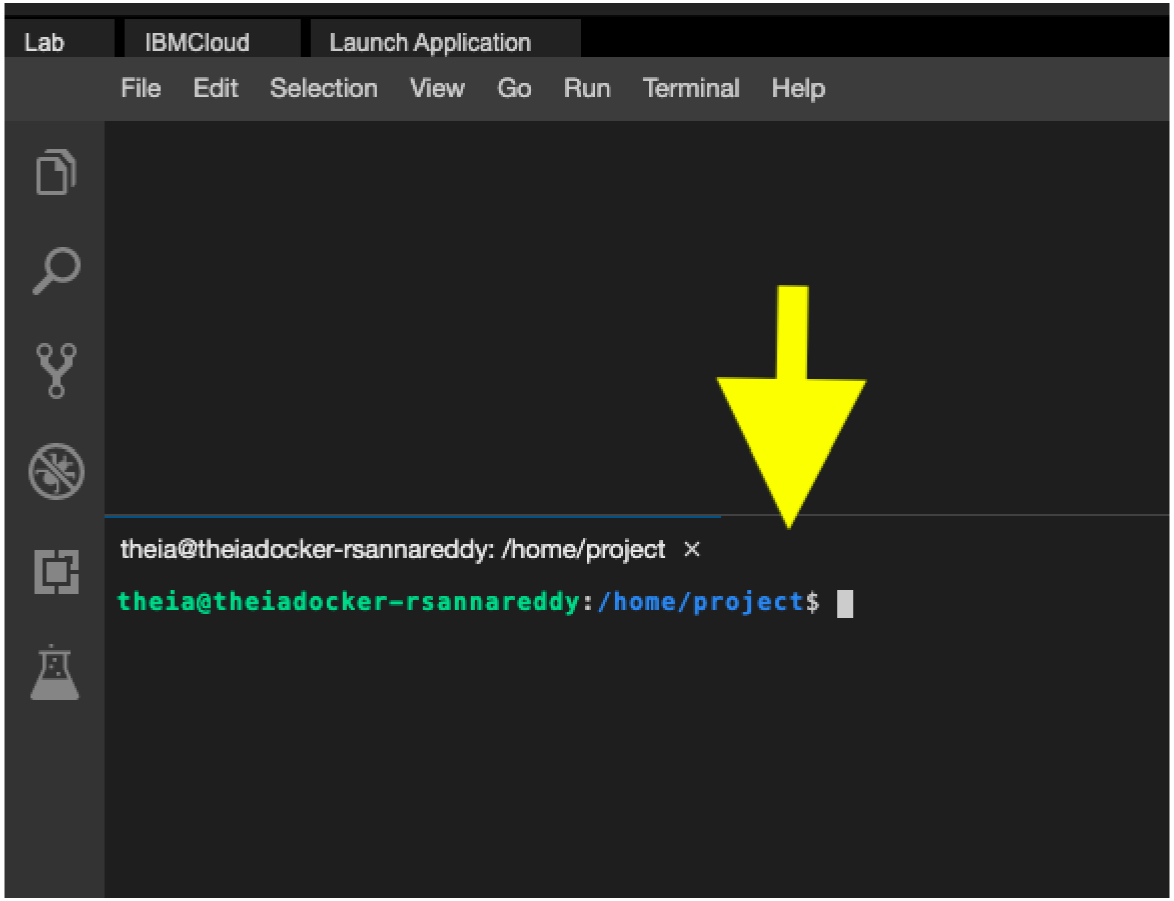
[NEW] LiveServer@5.7.9: Activation load time improved

Setup

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.



```
theia@theiadocker-rsannareddy:/home/project$
```

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

Exercise 1 - Navigating Directories

cd

In this exercise, you will explore directories on the cloud IDE Linux system using the command.`cd`

Recall the special paths:

Symbol	Stands for
`~`	Home directory
`/`	Root directory
`.`	Current directory
`..`	Parent directory

1.1. Changing working directory to home directory

```
cd ~
```

Copy the command above into the terminal and press Enter to run the command.

This will change your current working directory to the home directory .`~`

Note: (In our lab environment, your user's home directory is =).`~/home/theia`

1.2. Changing working directory to parent

```
cd ..
```

This will change your current working directory to the parent of the current working directory.

If your working directory was , then it will become .`/home/theia/home`

1.3. Changing working directory to root directory

```
cd /
```

This will change your current working directory to the root directory .`/`

1.4 Changing working directory to child

```
cd bin
```

This will change your current working directory to the directory.`/bin`

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

1.5. Changing working directory back to home directory

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

This will change your current working directory back to your home directory.

Of course, a simpler way to do this would be:

```
cd ~
```

1.6. Changing working directory back to project directory

```
cd ../project
```

This will change your current working directory back to your project directory.

The project directory is a special empty directory we provide for your work.

Exercise 2 - Browsing Directories

`ls`

In this exercise, you will explore browsing the content of directories using the command.`ls`

`ls` is a special command that the shell knows by default. You will learn about many more of these commands in the future.

2.1. Viewing files in the current working directory

```
ls
```

Typing by itself will show all files inside the current working directory.`ls`

Because you're in the directory (which is empty) ls will return nothing.`/home/project`

2.2. Viewing files in any directory

If you know the path to a directory, you can view its contents by typing:

`ls [PATH TO DIRECTORY]`

For example:

```
ls /
```

This will show the contents of the root directory.

Recall some of the directories you've learned in prior video(s):

Directory	Contains
`/bin`	System libraries
`/sbin`	Binaries that require root privileges
`/usr`	User programs and data
`/home`	Home directory
`/media`	Removable media device directories

```
ls /bin
```

This will show the contents of the directory.`/bin`

You might notice one of these files is called . That's because the command runs using the file .`"ls"ls/bin/ls`

Exercise 3 - Updating and Installing Packages

In your lab environment, we provide access to the `sudo` command. Be careful not to break your system!

3.1 Getting latest packages information

```
sudo apt update
```

This will fetch the lastest package information from trusted sources.

`apt update` doesn't actually update your packages; instead, it finds if any packages *can* be upgraded.

3.2. Updating nano

`nano` is a simple command that enables you to use the terminal as a text editor.

To get the latest supported version of , type:`nano`

```
sudo apt upgrade nano
```

You may be prompted: "`Do you want to continue? [Y/n]`"

Type "" and press Enter to continue. Updating will take time and will not affect this lab.`nnano`

Note: The capital in means it's the default - if you press enter without typing anything it uses the default .`YY/ny`

3.3. Installing vim

Another popular text-editing program is .`vim`

Because doesn't come with your system, you will need to install it:`vim`

```
sudo apt install vim
```

As with upgrading, you may be prompted: "`Do you want to continue? [Y/n]`"

In this case, type "" and press Enter to continue. You will be using in a later exercise.`yvim`

Exercise 4 - Creating and Editing Files

For the purpose of this lab, you will be use to create and edit files.`nano`

This is because is known as simple to use and easy to master.`nano`

On the other hand, vim can be harder to learn - though it has many more features.

4.1 Navigating to the project directory

We provide a clean project directory at . Ensure you're working in this folder using:`/home/project`

```
cd /home/project
```

Try auto-completing the path by typing and pressing TAB.`cd /home/pr`

If you type here, you should see no files.`ls`

4.2 Creating and editing a file

```
nano myprogram.py
```

This will create a (Python) file called and enable you to begin editing it using the text editor.`.pymyprogram.pynano`

Type the following to the file:

```
print('Learning Linux is fun!')
```

Now:

1. Press "CTRL-X" to exit
2. You will be prompted with:

```
Save modified buffer (ANSWERING "No" WILL DESTROY CHANGES) ?
 Y Yes
 N No           ^C Cancel
```

Press "y" to save.

3. Press "ENTER" to confirm the file name.

You should now be back at the terminal's command prompt.

4.3 Running the Python file you made

```
ls
```

You should now see that the file was created in your current working directory.`myprogram.py`

You can now run your Python file using:

```
python3 myprogram.py
```

Try auto-completing the command by typing and pressing TAB.`python3 my`

You should see the output:

```
Learning Linux is fun!
```

Otherwise, you may have had a typo in your program.

Report

Make a report about this work (**Screeshots 1-5**) and send Report to send to Moodle.

Week 3-Assignment 2 – Practical work 1 - task 2

USING SSH/RDP FOR REMOTE LINUX/UNIX/MAC/WINDOWS SERVERS MANAGEMENT

1. PURPOSE OF WORK

Get the initial skills of working with a remote host via ssh protocol. Secure SHell is the primary means of remotely managing networked computers running UNIX/Linux.

The Linux/UNIX commands are used: **ssh, scp, uname, passwd, date, who, pwd, mkdir, mc, exit, logout**

2. TASKS FOR WORK

2.1. Install **Secure Shell Extension** on your Chrome browser in the classroom or at home.

Using the Secure Shell Extension, connect to the remote server with your training account (login and password).

Use the **uname** command (with the appropriate parameters: -o, -I, -m, -p, -r, -v, -s, -n) to determine the type of operating system, hardware platform, release, version & name of the kernel, node name for remote Linux (make a Screenshot 1).

2.2. Download **Putty.exe** on your Windows computer in the classroom or at home.

Using the **putty.exe**, connect to the remote server with your training account (login and password).

Use the **passwd** command for change your password on remote server.

Use the **date** command (with the appropriate parameters) to determine the date and time of the remote server (make a Screenshot 2).

2.3. Start Your **Linux Virtual Machine** on VirtualBox on your Windows computer (show task 2 in week 2). Start Linux Terminal.

2.4. Use the **uname** command (with the appropriate parameters: -o, -I, -m, -p, -r, -v, -s, -n) to determine the type of operating system, hardware platform, release, version & name of the kernel, node name for local Linux (make a Screenshot 3).

2.5. Use the **ssh** command on Linux Virtual Machine (without loading the remote server shell) to determine the date and time (date command with the appropriate parameters) for remote server (make a Screenshot 4).

2.6. Using the **ssh** client on Linux Virtual Machine, connect to the remote server with your training account. Determine which users, in whichterminals and from which ip-addresses are connected to the server (**who** command) (make a Screenshot 5).

2.7. Using the **ssh** client on Linux Virtual Machine, connect to the remote server with your training account.

Determine which directory is current on the remote server (**pwd**). If it is different from /home/stYYNN, then go to /home/stYYNN. In theYour home directory, create subdirectory (**mkdir**) as your transliteration_surname (for example, ivanov).

Run the **mc** file manager on remote server, view the contents of the current directory (make a Screenshot 6).Finish working with the mc file manager (F10 or **exit**). End the ssh session (**logout**).

2.8. Using the **mc** command on Linux Virtual Machine configure **SFTP connection** to remote server. Copy any file from local Linux to remoteserver and back from remote to local (make a Screenshot 7).

2.9. Use the **ssh** command on Linux Virtual Machine (without loading the remote server shell), upload an arbitrary file (**scp** command) to thedirectory you previously created on the remote server. Without entering an ssh session on the remote server, view the contents of the serverdirectory /home/stYYNN (make a Screenshot 8).

2.10. Yourself need learn how to transfer files from/to a remote server using the **Secure Shell Extension** for Chrome and **WinSCP.exe** forWindows.

3. REPORT

Make a report about this work (Screenshots 1-8) and send it to the Moodle

REPORT FOR TASK 2: USING SSH/RDP FOR REMOTE LINUX/UNIX /MAC/WINDOWS SERVERS MANAGEMENT

Student Name Surname	Student ID	Date

Screenshot 1	Screenshot 2
…	…
Screenshot 7	Screenshot 8