

Aalto ELEC-E7130 Internet Traffic Measurement and Analysis

Remote software development guide for assignments

Solving the weekly exercises may require you to analyze datasets (files) that are only available on Aalto's network drives. Unfortunately, computers outside the Aalto campus network do not have direct access to these network drives.

In theory, you *could* transfer the files to your personal computer (via SFTP through one of Aalto's public servers) and operate & analyze on them locally. However, this might be infeasible in practice, as some of the files may be simply too large to be downloaded. Furthermore, later in work life you might find yourself in a situation where you are not even authorized to download files off some remote server (e.g., due to privacy or data protection reasons).

Therefore, it would be beneficial for you to learn how to perform the analysis directly on the remote host (or network) where the data files reside.

On our course, there are a few suggested options for accomplishing this:

1. [EASY] Go to one of the Aalto computer classrooms (Ubuntu Linux) and do your work there. The classroom computers have direct access to the course data directory over the campus network. The Ubuntu OS should also provide all the tools and libraries you might need on this course. Most Aalto servers also run Ubuntu Linux.
2. [MEDIUM/HARD] Connect via SSH to an Aalto server, and work remotely over the terminal. If you are unfamiliar with terminal-based workflows, this might be quite challenging.
3. [EASY/Problematic] Use an Aalto cloud Virtual Machine via <https://vdi.aalto.fi> (this may not always work due to limited number of VM instances in the Aalto Cloud)
4. [EASY] Use a modern editor with remote development features, such as Visual Studio Code (VSCode) This option allows you to open a remote workspace, where you develop and run your scripts, all the while using a local graphical editor session/window. So, you experience the ease of "local" development, while using tools and libraries on the remote machine (Aalto server).
How cool is that?!?

The first three suggestions are self-explanatory or have been covered somewhere else. **Next, we shall demonstrate how you could set up VScode for remote development.**

Visual Studio Code setup guide for remote development

This guide will describe how to install VSCode on your personal computer and configure it for remote development with Aalto servers or classroom computers as the remote host.

Note: You should need to do the following setup only once. Later you can easily reconnect to a recent remote workspace through graphical menus (point and click, see Additional Notes section below).

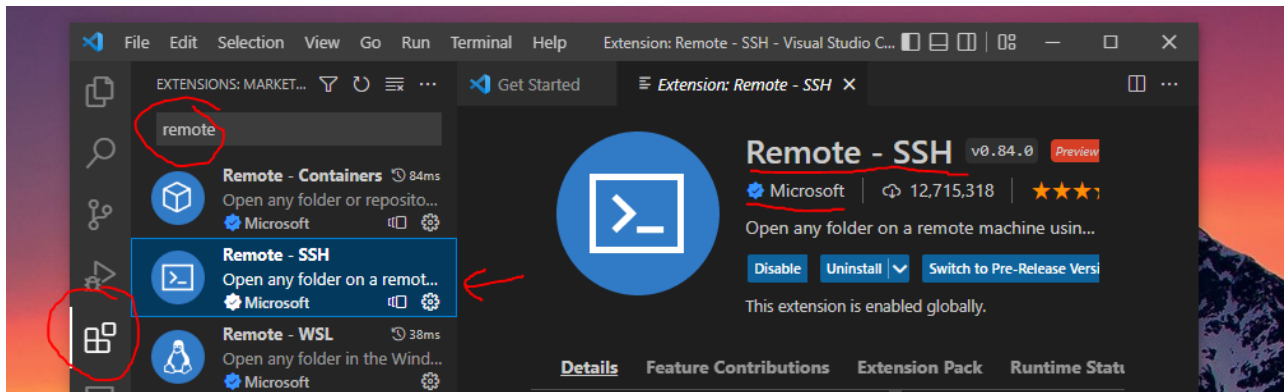
1. Download and install VSCode

Download and install [VSCode](#) on your **personal computer**

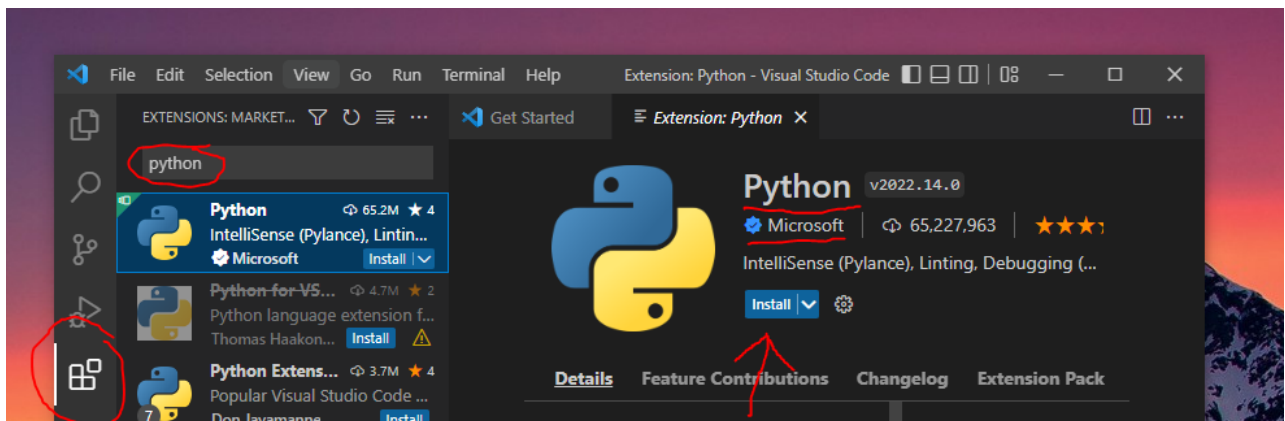
After the installation has finished, open the application

2. Install required extension(s)

In VSCode window, select "Extensions" view from the left menu bar. Then find and install the extension for Remote SSH:



You may also want to install support extension for Python development (optional):

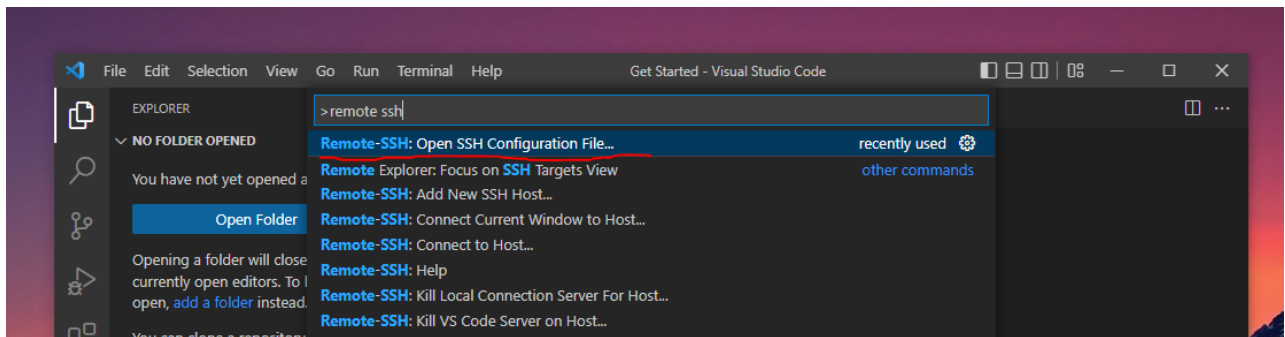


Alternatively, you can install the extensions via the web store:

- [Remote SSH extension \(by Microsoft\)](#)
- [Python Extension \(by Microsoft\)](#)

3. Edit local SSH config

Next you need to configure some SSH config parameters to make your life easier. Press **Ctrl+Shift+P** to open VSCode quick launch menu. Then type “ssh config” to locate the following action entry:



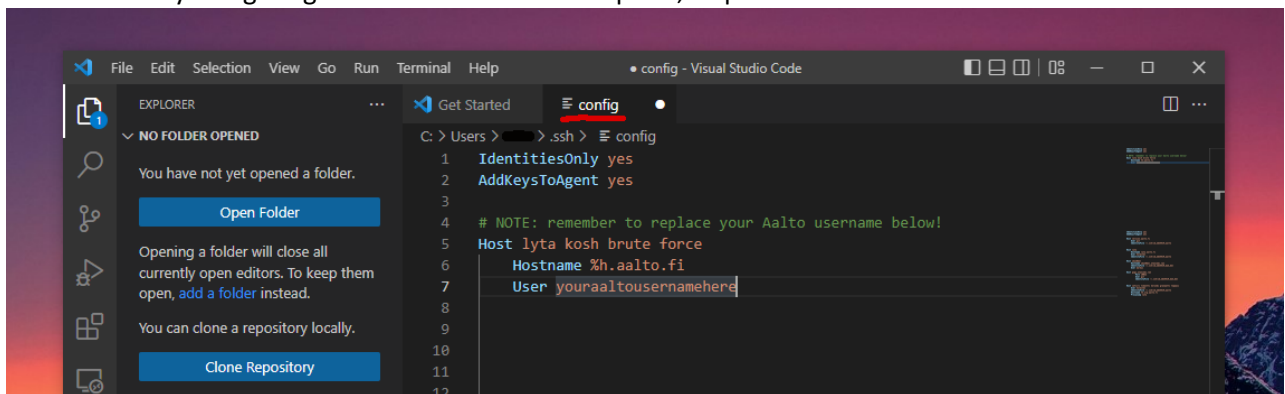
Click the first file option offered, typically something like “C:\Users\Myuser\.ssh\config”

Next, place the following contents into the config file:

```
Host lyta kosh brute force
  Hostname %h.aalto.fi
  User youraaltousernamehere
```

Remember to replace your Aalto username on the last line. The file should look something like below.

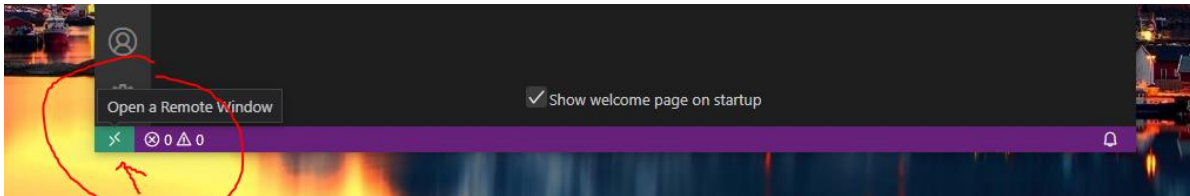
Save the file by navigating “File > Save” from the top bar, or press Ctrl+s.



4. Connect VSCode to remote server via SSH

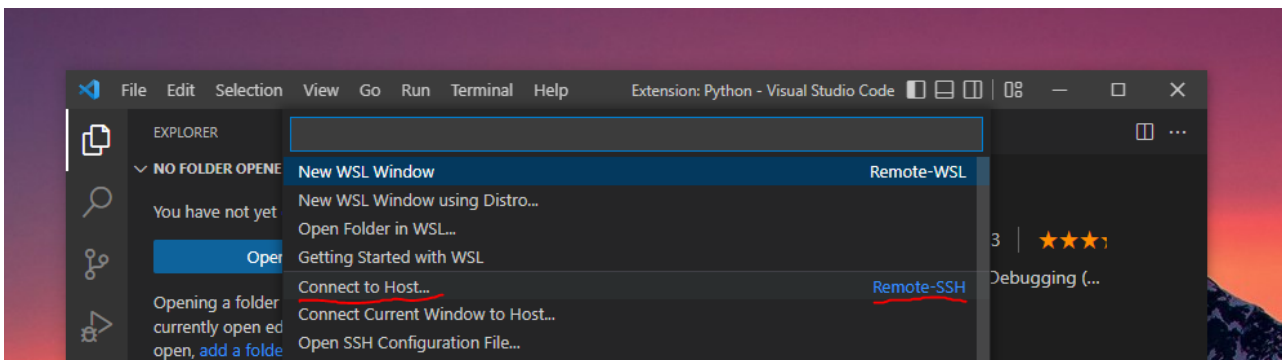
Now we can finally connect our local VScode to an Aalto server

Click the green button on bottom left corner:

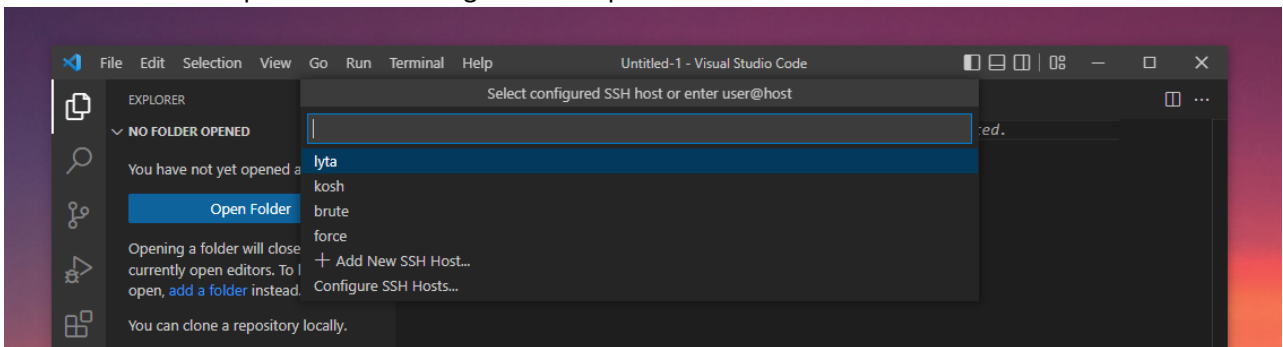


A menu should appear in the top-center of VSCode.

Select "Connect to Host..."

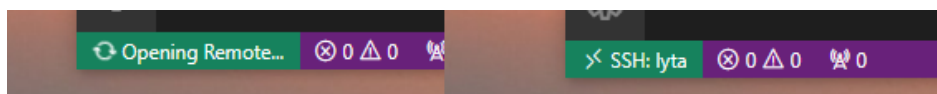


Next you need to select which remote host to connect to. VSCode should now suggest remotes that were defined in the previous SSH configuration step:



Select any one of them. Next VSCode may ask further details about the remote (e.g., which OS is running on the remote, which is "Linux" for Aalto servers). You may also need to enter your Aalto password when prompted.

A new VSCode window should open. Pay attention to its bottom left corner:



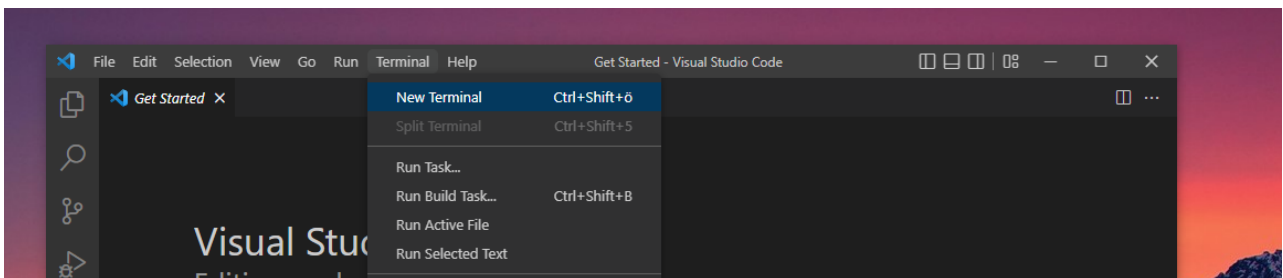
A successful connection will present the "SSH: <hostname>" as shown on the right. If you run into any problems, you may need to consult a course assistant. (Or just learn to google problems like everyone else 😊).

5. Create workspace on remote host

To edit files in VSCode, you need to open a “workspace”, which is just a fancy name for an ordinary directory. Since we are working remotely, the workspace must exist on the remote machine (Aalto network drive), and **not** locally. Next, we will create such a workspace directory on the remote (you can skip this step if you already have created one).

Unamusingly, VSCode does not seem to provide an easy option for creating a directory on the remote, *except* by opening a terminal and running the creation command there.

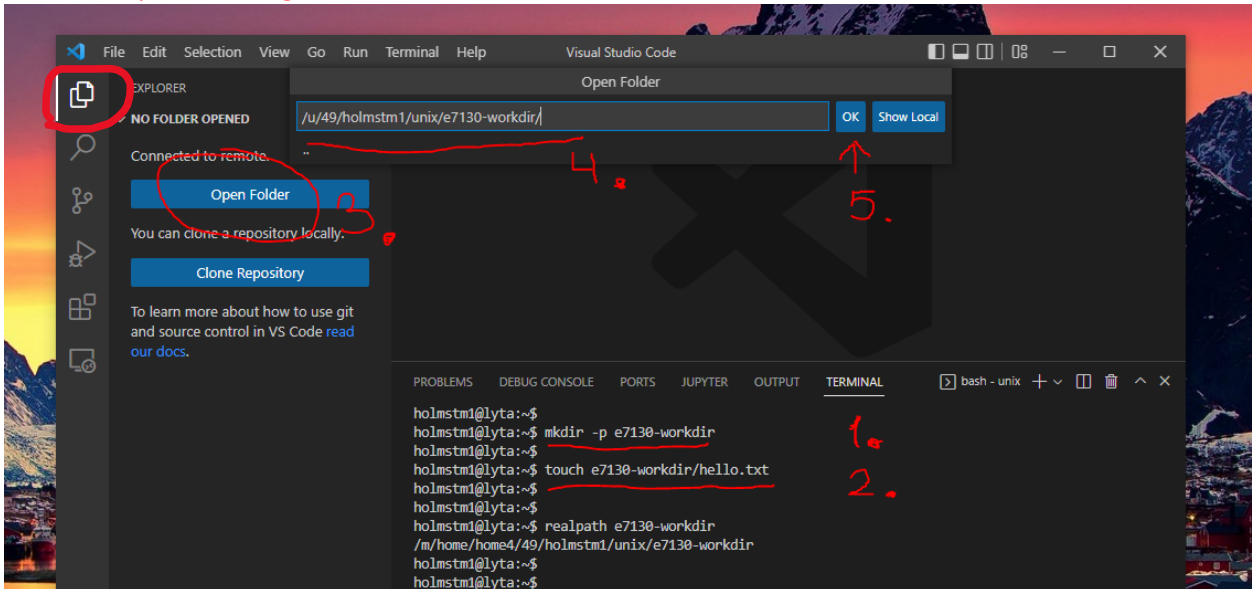
So, click "Terminal > New Terminal" on top bar of VSCode. A new terminal should open at the bottom of the screen.



Enter these commands to create the workspace directory, and an empty text file inside it (for demonstration purposes):

```
$ mkdir e7130-workdir
$ touch e7130-workdir/hello.txt
```

Make sure you are using the correct remote-connected VSCode window!



The last “realpath” command simply shows the full path to the directory we just created, which may be useful to know when we next open the directory.

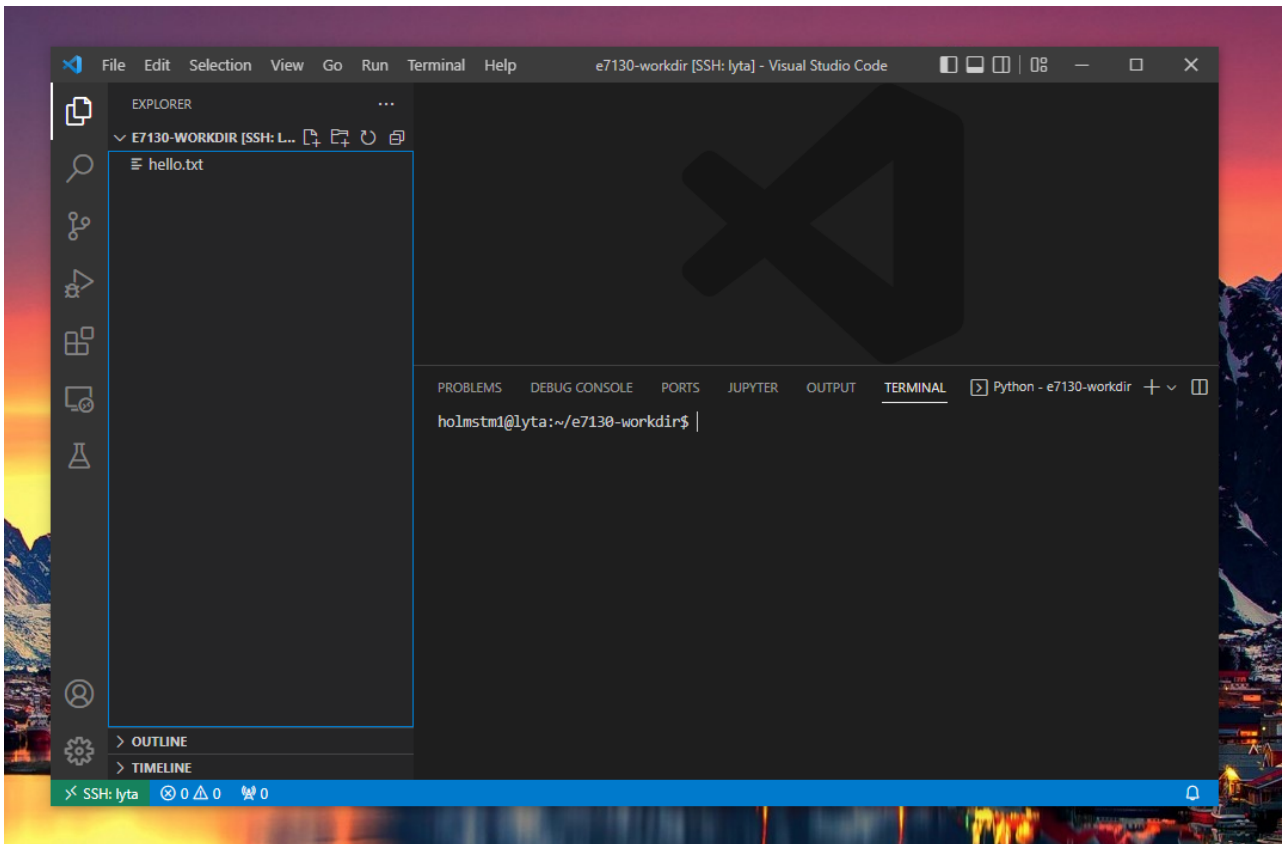
6. Open workspace on remote

Next, press the big blue button that says, “Open Folder”. Make sure you are in the “Explorer” view, as shown on the bar on the left.

Alternatively, if the button is not visible, click “File > Open Folder” from the top bar.

In either case you need to enter the directory path. Typically, if you write the first few letters of the directory, VSCode will suggest matching entries (e.g., write “e7130” and you should see our “e7130-workdir”). You can also copy-paste the full path shown in the previous step.

Press OK to open the workspace.



If everything went well, you should see the file "hello.txt" in the file explorer view (the one we created in the previous step).

We are now ready for action!

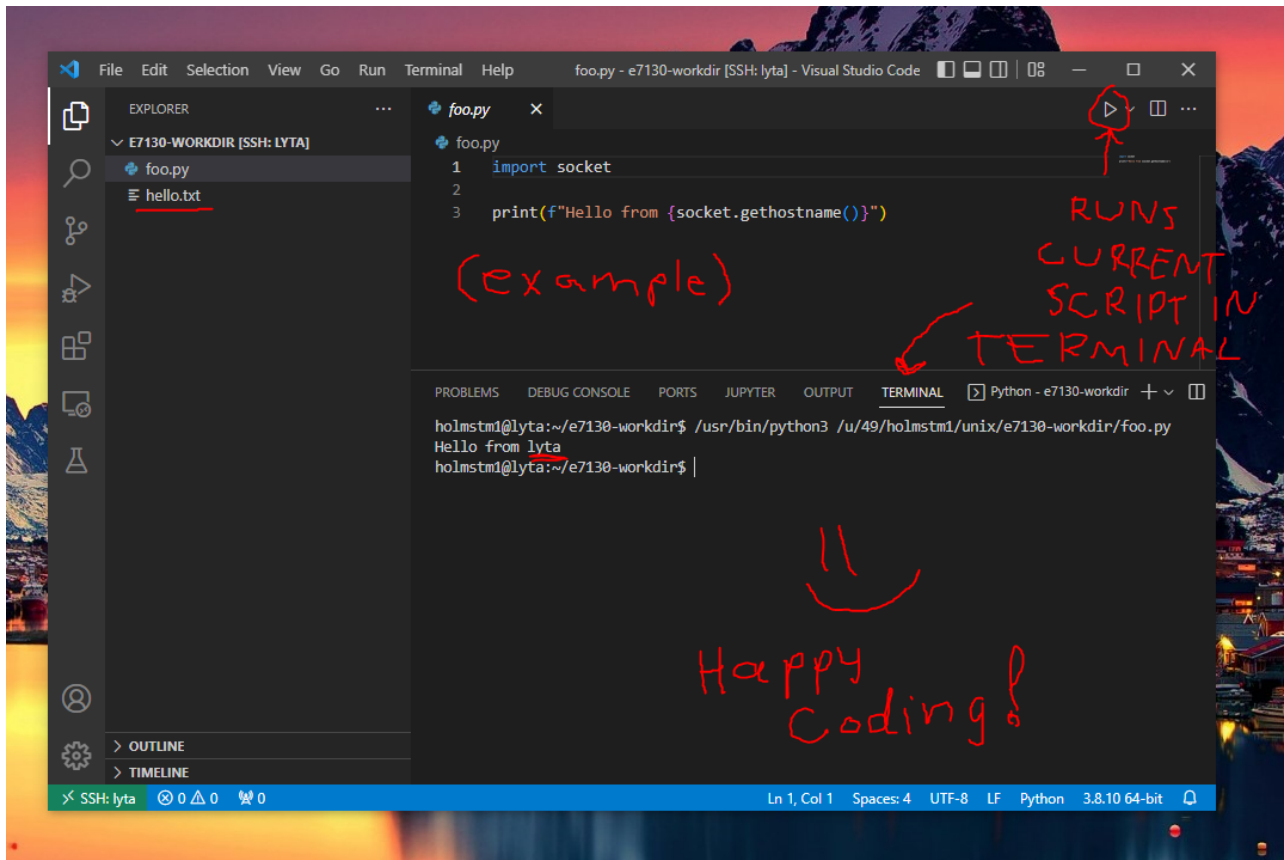
7. Create and run a simple example script on the remote

Create a new file in VSCode and enter the following content into it:

```
import platform
print(f"Hello from host {socket.gethostname()}")
```

save the file as "foo.py". Just make sure the file contents look exactly like in the screenshot below; Python is precise about whitespace and indentation!

Then you can run the script by pressing the "play button" in the top right corner. The script should run in the terminal area, and show output of the print() statement:



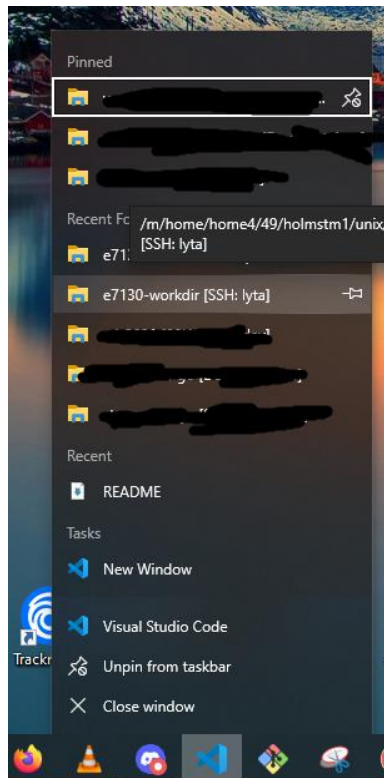
That's it! If you get similar output, you can proceed solving the homework assignments!

Additional hints

A) Reconnecting to previous workspace

The steps above describe setting up VSCode for the first time. In the future, you should be able to quickly open recent workspaces from the VSCode *Get started* view.

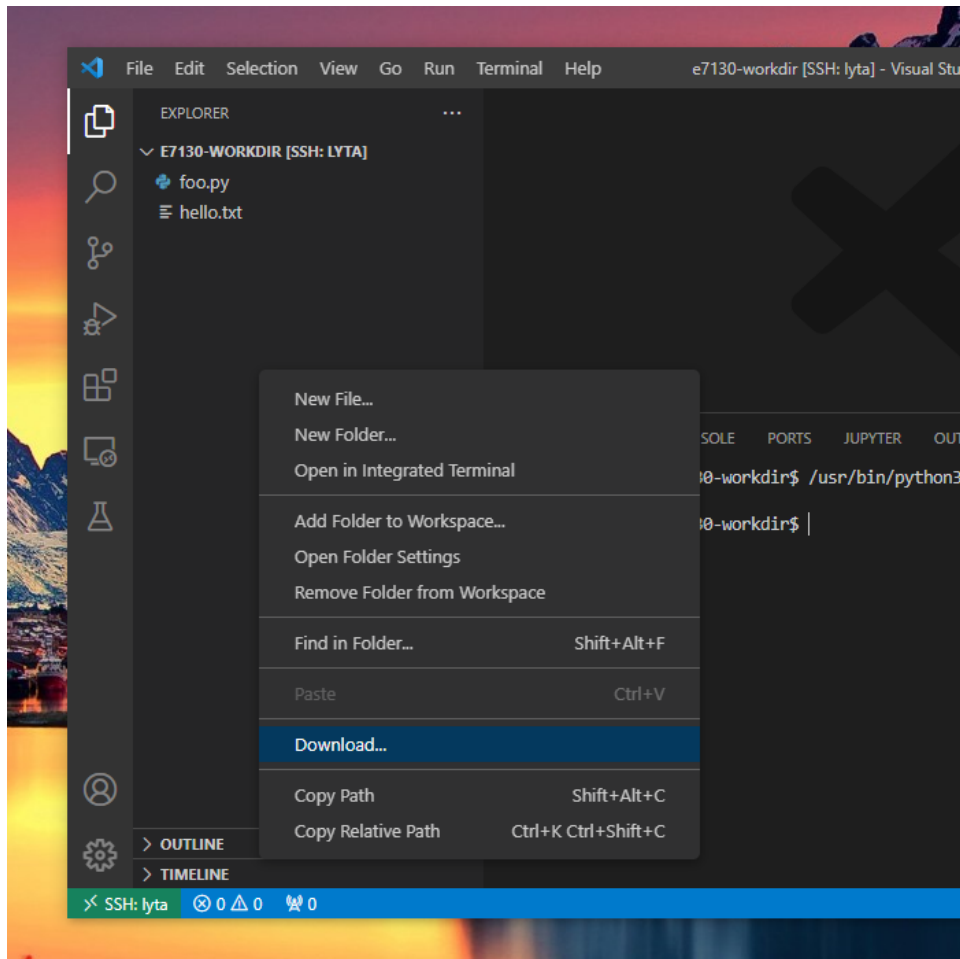
Furthermore, if you are using Windows, pin VSCode into the taskbar; now you can quick-launch previous workspace sessions by right-clicking the icon:



B) Download files from the remote to local machine

The files you create in the workspace reside on the remote host. You can download the workspace directory to your local machine by right-clicking on the Explorer view and selecting "Download...". You can also download files one-by-one with this method.

(This might be useful when you want to create a zip file for submission through MyCourses.)



C) Select appropriate remote host for your homework

The servers **lyta** and **kosh** are general light-duty servers, which are not intended for computationally heavy tasks. Instead, prefer **brute.aalto.fi** and **force.aalto.fi** for such tasks.

 Keep this in mind when running your homework assignment code on the remote machines.

For more information, see: <https://www.aalto.fi/en/services/servers-for-light-duty-calculation>

You are also welcomed to use the classroom computers (in Y342, for example) for your homework computations. However, there is no way to directly connect to them, so you need to use one of the public servers as a “jump” host. To achieve this, add a ProxyJump command in your local SSH config:

```
Host classroomcomputer1 classroomcomputer2
    User youraaltousernamehere
    Hostname %h.org.aalto.fi
    ProxyJump lyta
```

Now you should be able to directly SSH into “classroomcomputer1” (for example) from your local (personal) computer.

This feature is especially useful when working with VSCode Remote Development extension

Remember to replace the “classroomcomputer1”, “classroomcomputer2”, etc. with actual classroom computer names. The names are written on sticker labels attached to each computer’s main unit. A list of names is available online:

<https://www.aalto.fi/en/services/linux-computer-names-in-it-classrooms>

Pick one at random to avoid everyone congesting the same computer!

NOTE: All Aalto servers and classroom computers are configured to use the same shared home directory. Thus, if you create a directory on one server, it should be accessible on other servers, as well as on the classroom computers. This fact makes switching remote hosts in VSCode trivial!

D) Setup SSH keys for authentication

You might have noticed that VSCode frequently asks for your Aalto password. This can be overcome by using SSH keys. See:

<https://scicomp.aalto.fi/scicomp/ssh/#ssh-keys-better-than-just-passwords>

This task is not strictly necessary and is left as an exercise to the interested reader.