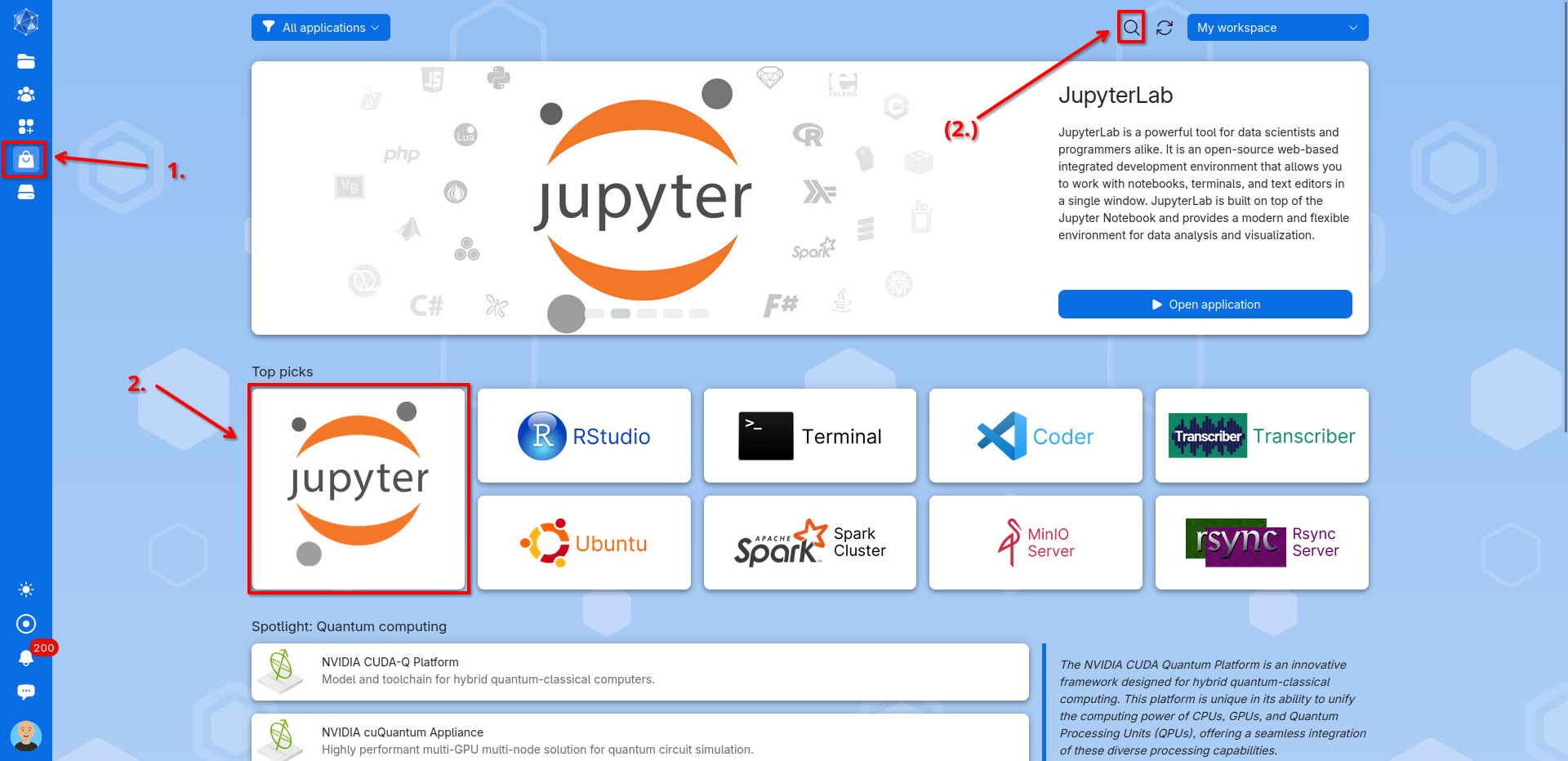
**Tutorial: Using Jupyter on the Server**

**1. Add Environment to kernel**

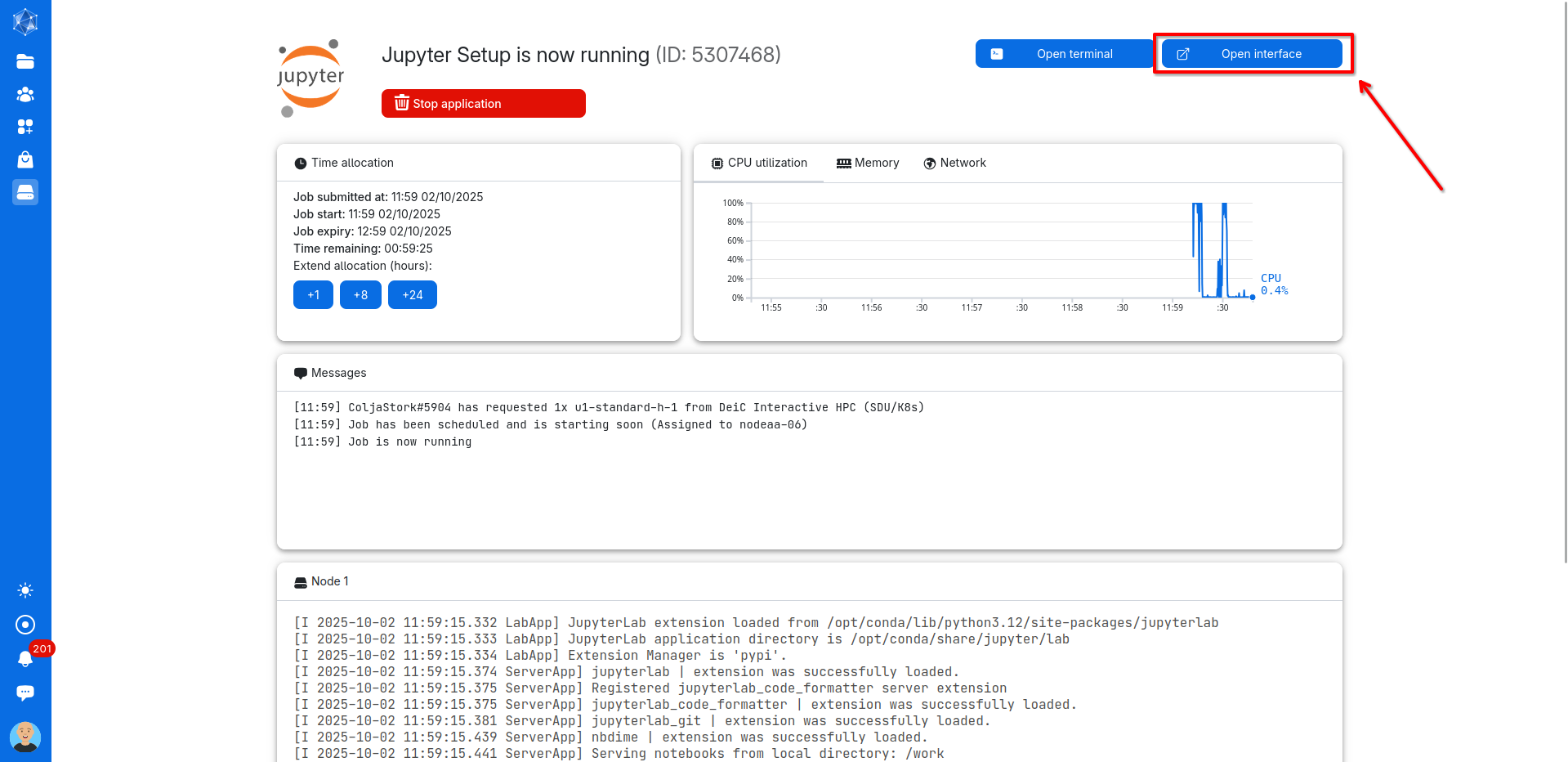
To use Jupyter on the Server cluster in combination with your custom python environments, it is necessary to manually connect Jupyter with these. To do so open up a new Jupyter server process by navigating to ***Applications*** and then finding ***JupyterLab***or searching for it in the search bar and click on it.



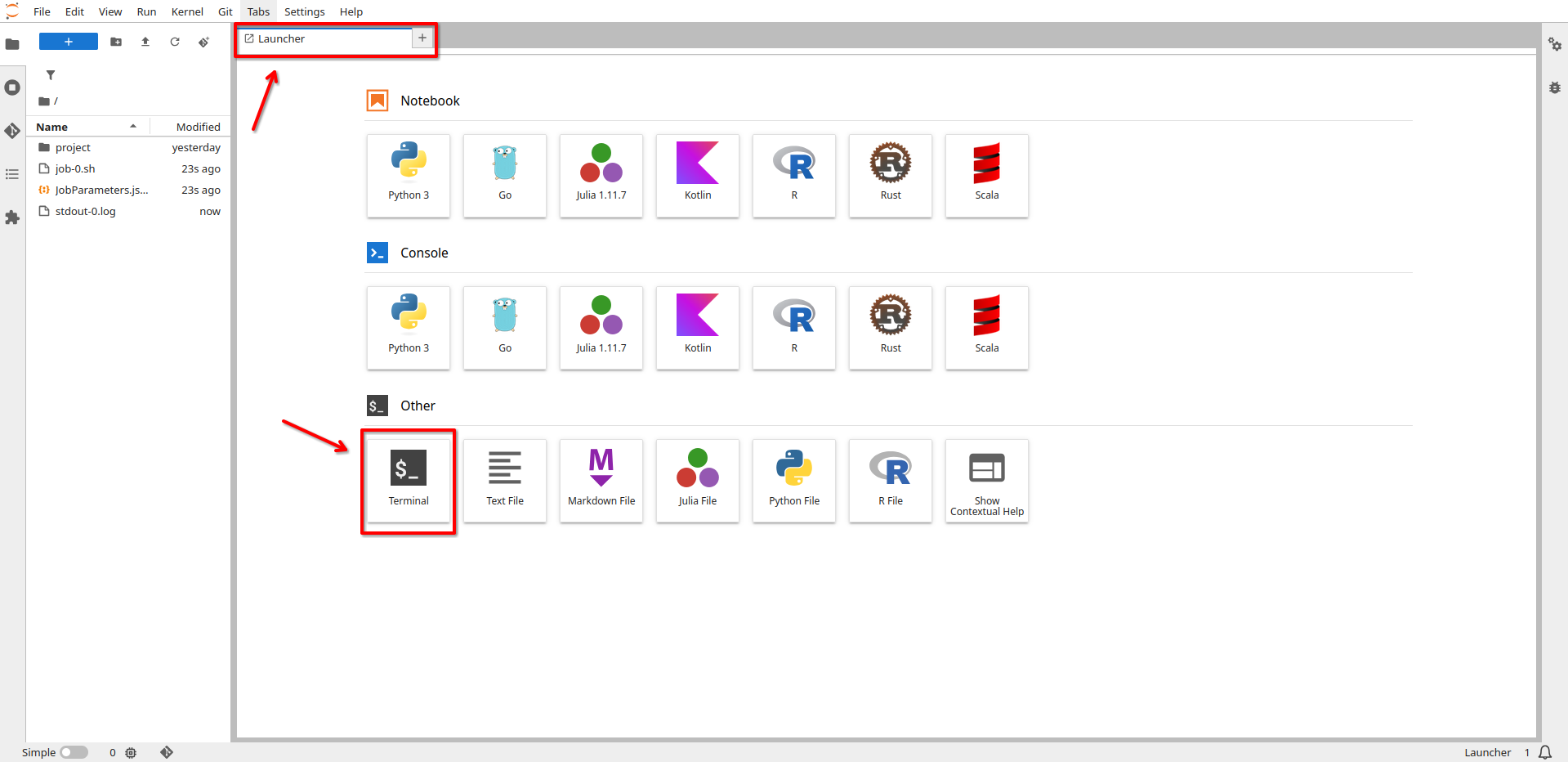
The JupyterLab Process setup window will then open. Here you need to configure the following options:

1. **Process name**: Choose a name for your process.
2. **Machine type:** Select the smallest machine, since the setup does not require much processing power or resources.
3. **Runtime duration:** Set how long the process should run. One hour is usually more than enough. *(Note: Any unused time after the process ends will not be deducted from your balance.)*
4. **Working folder:** Select your project directory in which your environment is installed as the starting folder.
5. *(Optional):* Select and use a bash script to initialize conda and the environment like the one provided in the Github repository. This tutorial will do without.
6. **Submit**: Click Submit to launch the process.

In the newly opened Process window click **Open Interface** to access JupyterLab.



In the launcher tab find the terminal and start it by clicking on it.



In the terminal you can now initialize Conda by running:

*eval "$(/work/project/miniforge3/bin/conda shell.bash hook)"*

With Conda initialized, activate your environment:

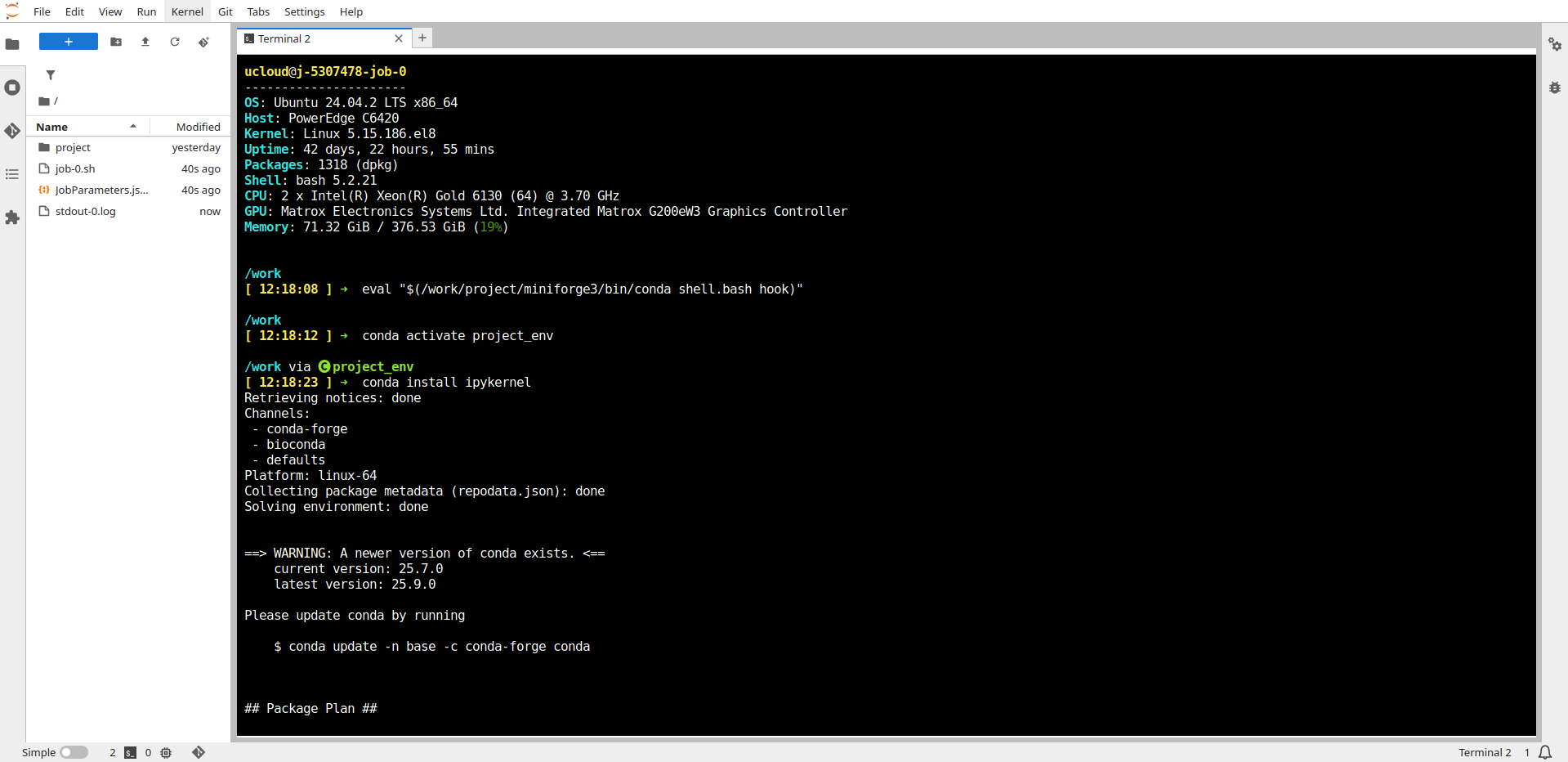
*conda activate project\_env*

If the ***ipykernel*** is not installed in your environment (should be when the environment file provided in the Github Repository was used), you can do so by running while the environment is still activated:

*conda install ipykernel*

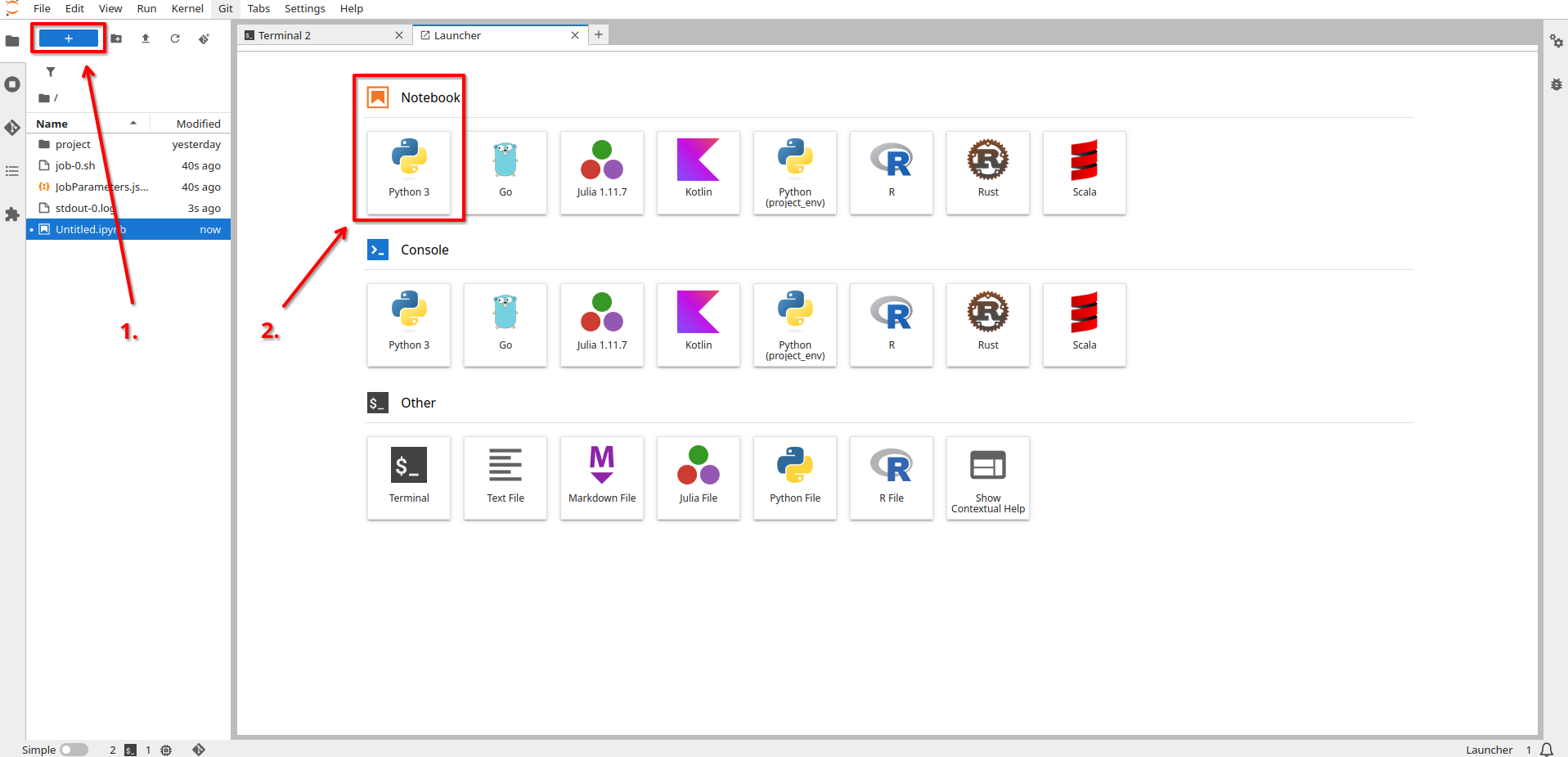
You can then add your environment to the Jupyter kernel list through:

*python -m ipykernel install --user --name project\_env --display-name "Python (project\_env)"*

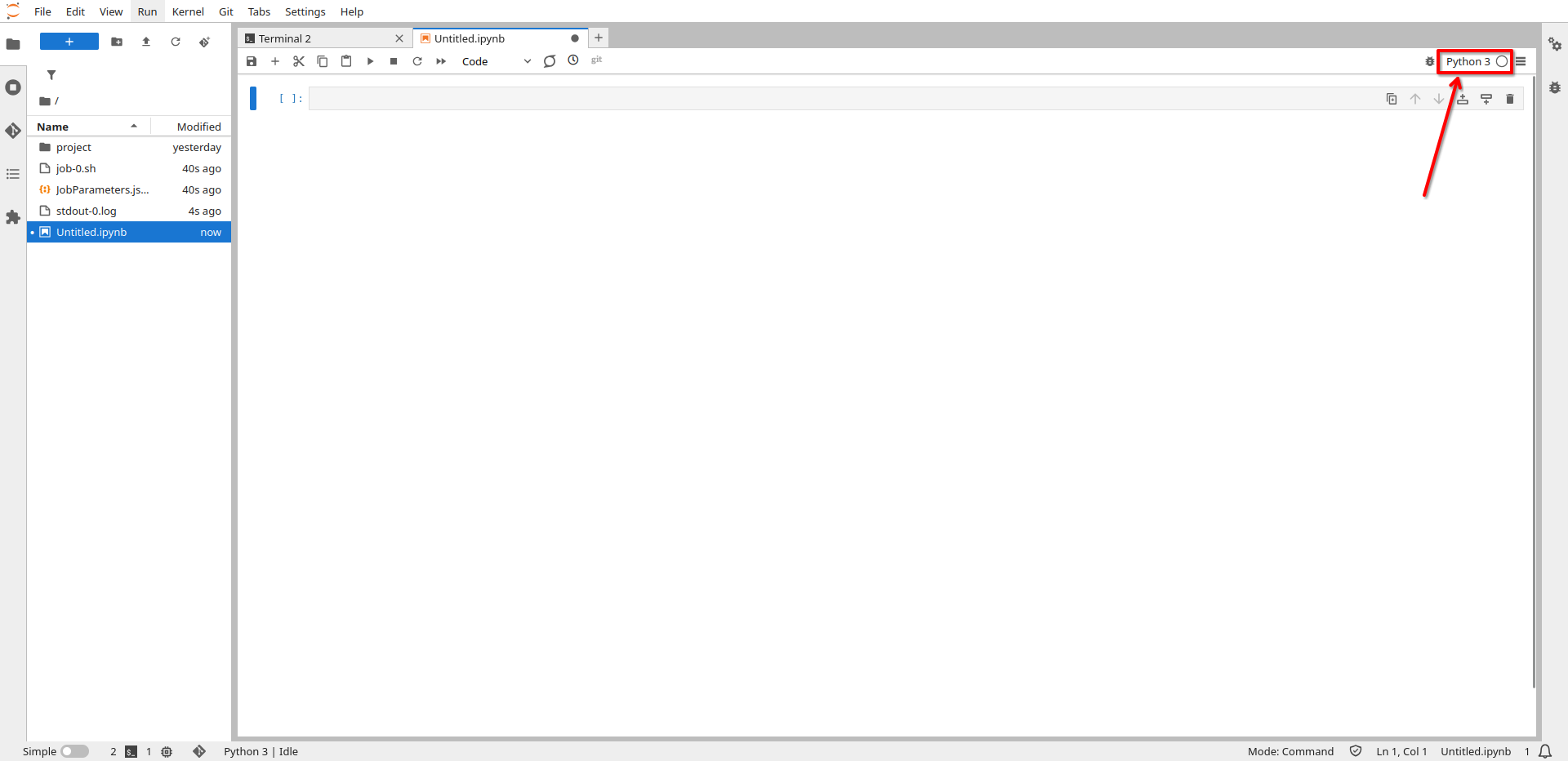


**1. Select Kernel in Jupyter Notebook**

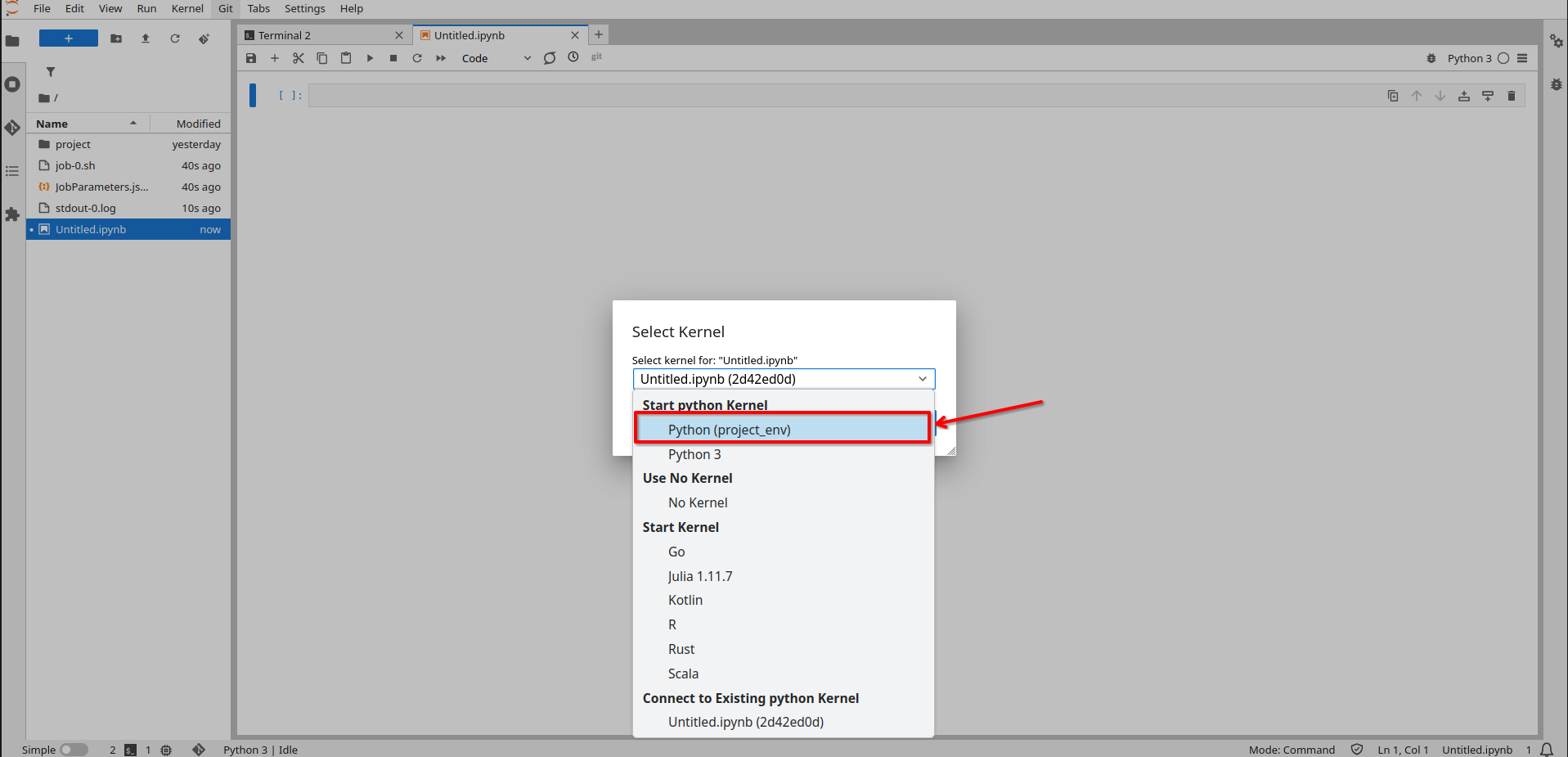
You can then open a Jupyter notebook by opening a new launcher by clicking on the ***blue plus*** and then clicking on ***Pyhton 3 Notebook.***



Here click on the Kernel in the top right corner, by standard it should be called Python 3.



You should now be able to find your custom environment in the Kernel list and select it. Now your Jupyter Notebooks will recognize all the packages from your environment.



For future JupyterLab server processes you can easily just add the the commands to a initialization bash script which will then allow you to always find your environment in the kernel list:

*# setup Jupyter kernel*

*python -m ipykernel install --user --name project\_env --display-name "Python (project\_env)"*

The initialization script available at the Github repository includes this command. See the ***Server Setup Tutorial*** on how to set initialization scripts.