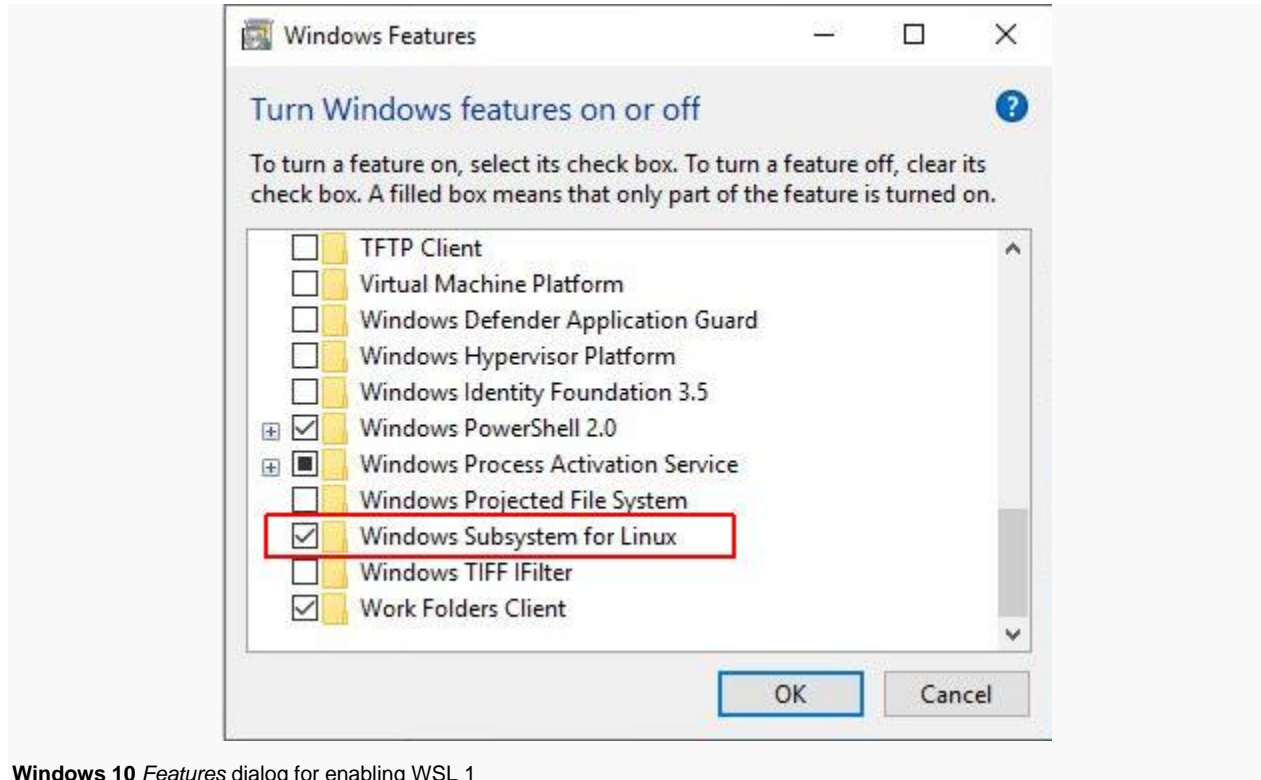


# Windows Subsystem For Linux

## Enabling WSL 1 (recommended for Windows 10)

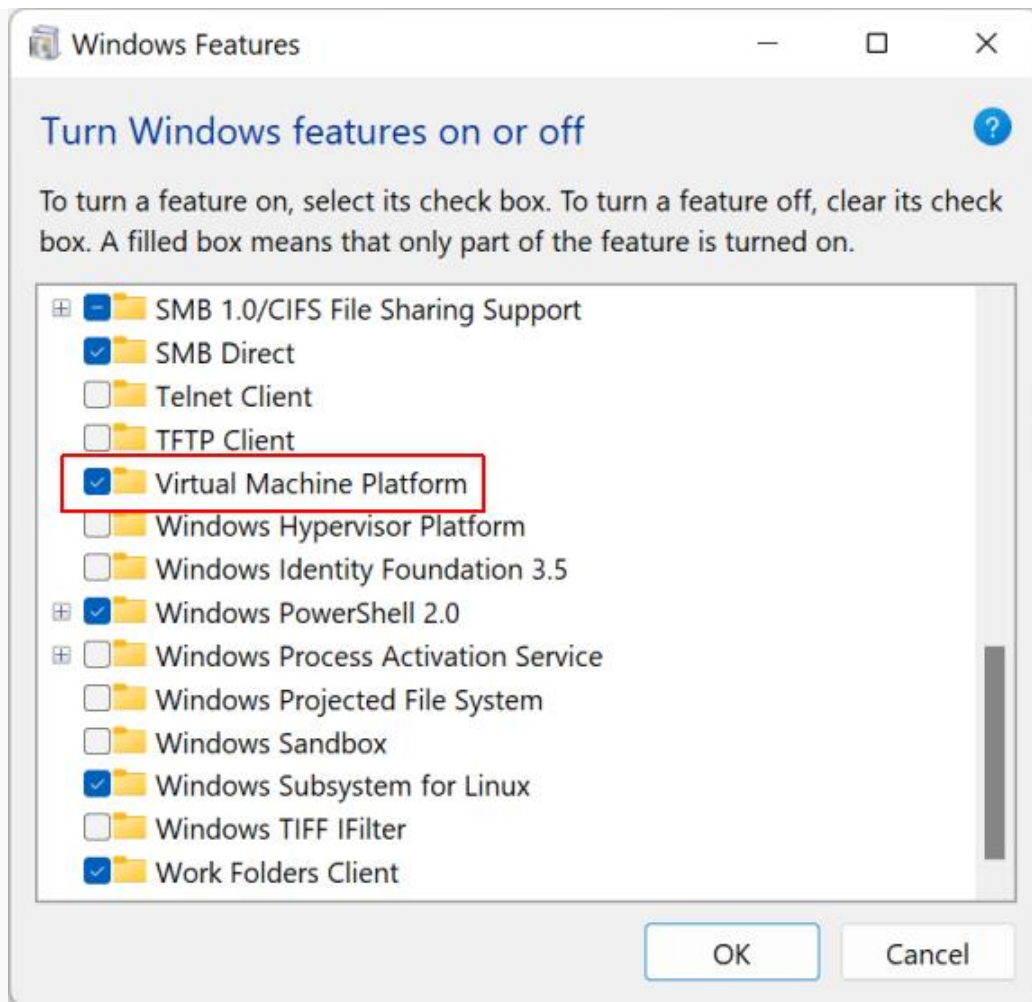


Windows 10 Features dialog for enabling WSL 1

For enabling WSL 1 on Windows 10 the *Windows Subsystem for Linux* feature must be activated using the following steps:

1. Right-click on the Windows Start menu icon, choose *Search* and type **Windows Features**. Select the top entry (category *Control panel*) to enable or turn off Windows-Features. The Windows-Features dialog will be opened.
2. Select in the upcoming dialog the check box for **Windows Subsystem for Linux** from the bottom of the list and press the *OK* button. Applying the changes may take a few minutes. Finally, press the *Restart now* button to reboot the computer.

## Enabling WSL 2 (recommended for Windows 11)



**Windows 11** *Features* dialog for enabling WSL 2

**Important:** WSL 2 requires virtualization technology on the computer. For AMD CPUs this technology is called AMD Secure Virtual Machine (AVM-SVM) or AMD Virtualization (AMD-V), for Intel CPUs it is called Intel VT-X. Instructions how to enable virtualization technology can be found here: [Enable virtualization on Windows 11 PCs](#)

After this, for enabling WSL 2 the *Virtual Machine Platform* feature must be activated using the following steps:

1. Right-click on the Windows Start menu icon, choose *Search* and type **Windows Features**. Select the top entry (category *Control panel*) to enable or turn off Windows-Features. The Windows-Features dialog will be opened.
2. Select in the upcoming dialog the check box for **Virtual Machine Platform** from the bottom of the list and press the *OK* button. Applying the changes may take a few minutes. Finally, press the *Restart now* button to reboot the computer.

After that, use the Microsoft Store app and look for the Linux distribution you want to use. Install the Linux distro of your choice.

- The Linux distribution can be launched from the **Start menu**.

## Installing Miniconda on Linux

- Download the latest Miniconda script for Linux

```
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86\_64.sh
```

- Install Miniconda

```
bash Miniconda3-latest-Linux-x86_64.sh
```

- To create an environment with a specific version of Python

```
conda create -n int2ai python=3.9
```

- Verify that the new environment was installed correctly

```
conda env list
```

- To activate this environment

```
conda activate int2ai
```

- To install JupyterLab run

```
pip install jupyterlab
```

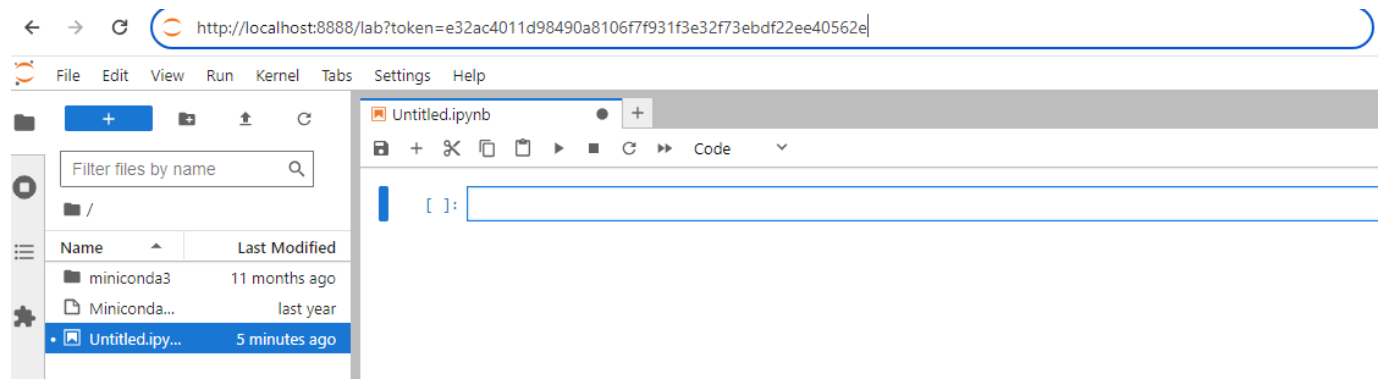
- We'll access the Notebook interface remotely with a browser running in Windows Host. So, launch the Notebook server running on WSL with *--no-browser* flag.

```
jupyter-lab --no-browser
```

- Copy the link shown on terminal to your browser

```
(int2ai) maged@Maged-PC:~$ jupyter-lab --no-browser
[I 2024-02-15 09:54:40.097 ServerApp] jupyter_lsp | extension was successfully linked.
[I 2024-02-15 09:54:40.104 ServerApp] jupyter_server_terminals | extension was successfully linked.
[I 2024-02-15 09:54:40.113 ServerApp] jupyterlab | extension was successfully linked.
[I 2024-02-15 09:54:40.120 ServerApp] Writing Jupyter server cookie secret to /home/maged/.local/share/jupyter/runtime/jupyter_cookie_secret
[I 2024-02-15 09:54:40.476 ServerApp] notebook_shim | extension was successfully linked.
[I 2024-02-15 09:54:40.508 ServerApp] notebook_shim | extension was successfully loaded.
[I 2024-02-15 09:54:40.512 ServerApp] jupyter_lsp | extension was successfully loaded.
[I 2024-02-15 09:54:40.514 ServerApp] jupyter_server_terminals | extension was successfully loaded.
[I 2024-02-15 09:54:40.517 LabApp] JupyterLab extension loaded from /home/maged/miniconda3/envs/int2ai/lib/python3.9/site-packages/jupyterlab
[I 2024-02-15 09:54:40.517 LabApp] JupyterLab application directory is /home/maged/miniconda3/envs/int2ai/share/jupyter/lab
[I 2024-02-15 09:54:40.518 LabApp] Extension Manager is 'pypi'.
[I 2024-02-15 09:54:40.576 ServerApp] jupyterlab | extension was successfully loaded.
[I 2024-02-15 09:54:40.577 ServerApp] Serving notebooks from local directory: /home/maged
[I 2024-02-15 09:54:40.577 ServerApp] Jupyter Server 2.12.5 is running at:
[I 2024-02-15 09:54:40.577 ServerApp] http://localhost:8888/lab?token=647c52a273391eeeb99b5f1e239cbfd47e9961f7d0ef4e29
[I 2024-02-15 09:54:40.577 ServerApp] http://127.0.0.1:8888/lab?token=647c52a273391eeeb99b5f1e239cbfd47e9961f7d0ef4e29
[I 2024-02-15 09:54:40.577 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 2024-02-15 09:54:40.583 ServerApp]

To access the server, open this file in a browser:
file:///home/maged/.local/share/jupyter/runtime/jpserver-382-open.html
Or copy and paste one of these URLs:
http://localhost:8888/lab?token=647c52a273391eeeb99b5f1e239cbfd47e9961f7d0ef4e29
http://127.0.0.1:8888/lab?token=647c52a273391eeeb99b5f1e239cbfd47e9961f7d0ef4e29
```



To change the home directory for Jupyter

```
cd /mnt/d/  
jupyter-lab --no-browser
```