

VS Code Debugging on the Compute Node of a Remote Server

1. Install Plugin in Local VS Code.

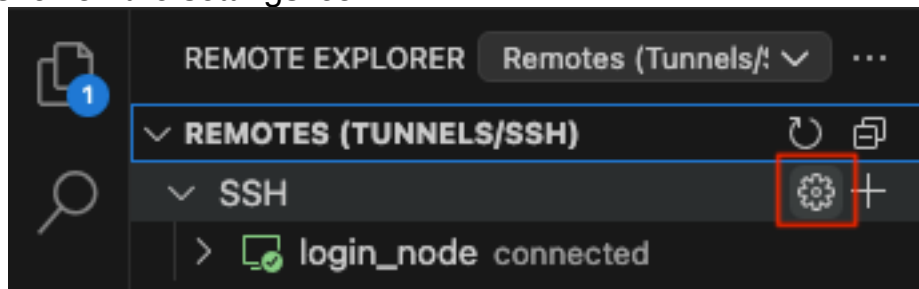
Click the Extensions panel on the left side of VS Code to install two plugins — Remote SSH and Python.



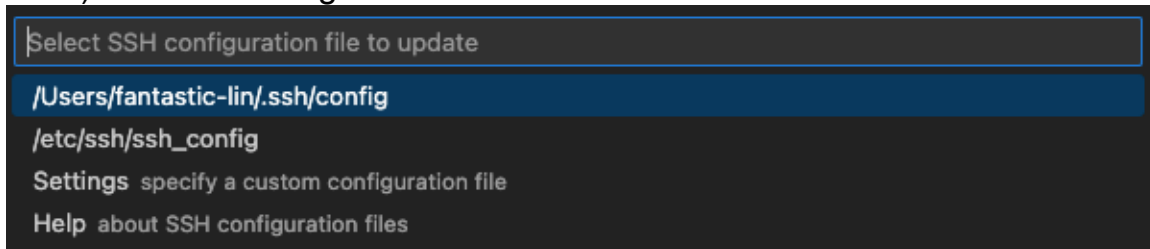
2. Set Up Remote Connection.



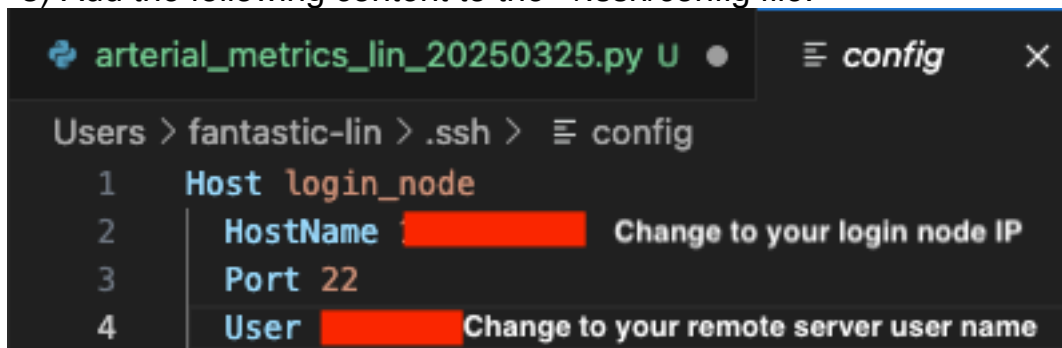
1) Click on the settings icon.



2) Edit .ssh/config.

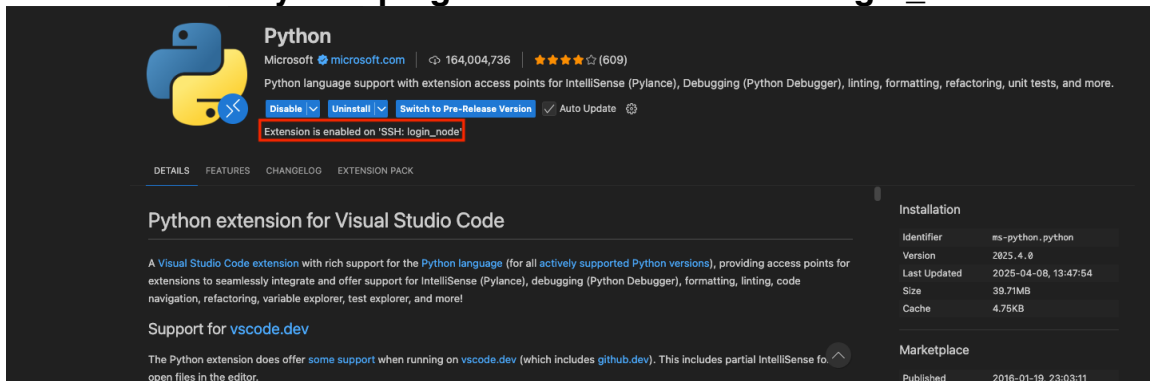


3) Add the following content to the ~/.ssh/config file.

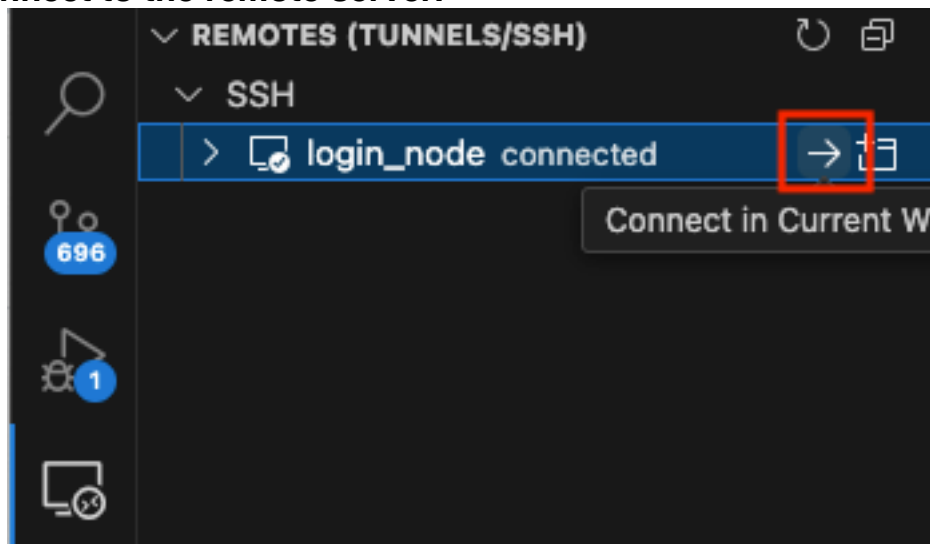


4) Save.

3. Ensure the Python plugin is enabled on “SSH:login_node”.



4. Connect to the remote server.



5. Implement port forwarding between the local port [localhost:8899] and the remote computing node [178.187.0.3:8899]

Enter the following command in the local terminal.

ssh -L [port]:[compute node ip]:[port][remote server user name]@[login node ip]

e.g.: ssh -L 8899:178.187.0.3:8899 linlin@10.8.0.41

6. Open the remote server terminal in VS Code. Use qsub to apply for compute node resources, and then you can access the compute node.

qsub -l -q batch -l nodes=node01

explanation:

-l submit an **interactive job**

-q specify queue

-l specify the compute nodes whose IP address correspond to the

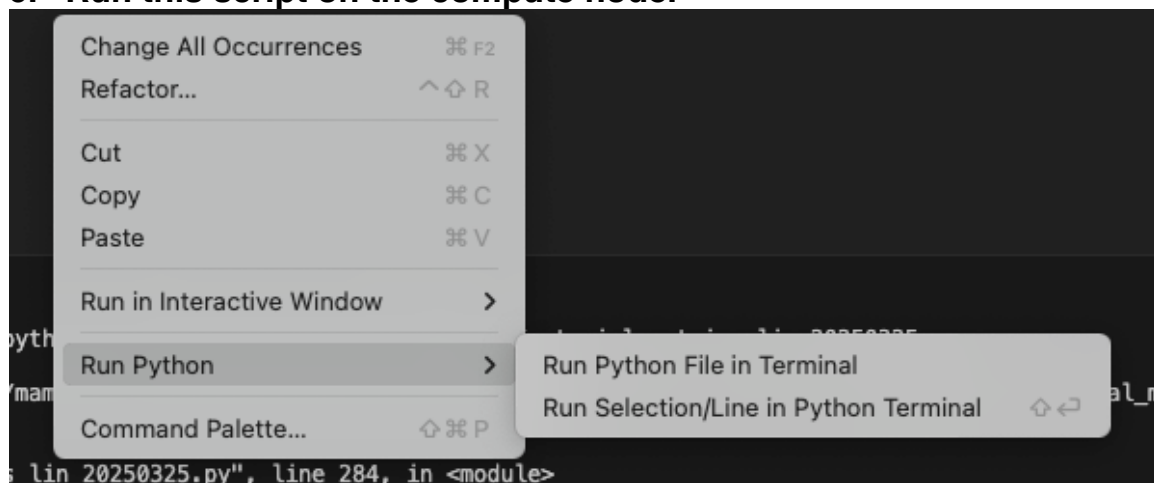
[compute node IP] used above for port forwarding

7. Pip install debugpy.

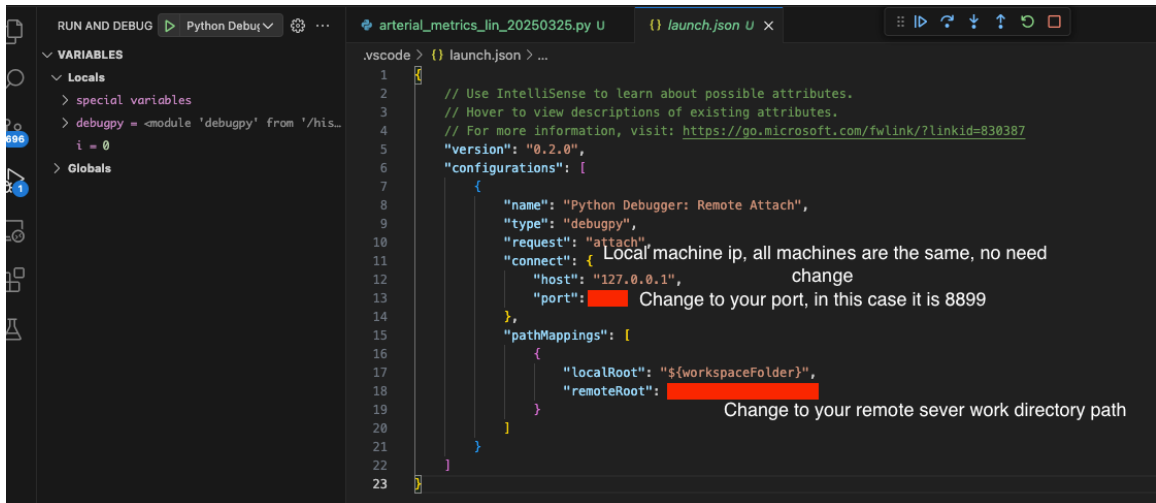
8. Add the following line at the beginning of your Python script.

```
test > arterial_metrics_lin_20250325.py
277
278 # if __name__ == "__main__":
279 #     main()
280
281 import debugpy
282 debugpy.listen(address=('0.0.0.0', 8899))
283 print("✅ Waiting for debugger to attach...")
284 debugpy.wait_for_client()
285
286
287 for i in range(3):
288     print(f"Count: {i}")
289
```

9. Run this script on the compute node.



10. Click the debug settings icon. This will open a file named launch.json. Edit it.



11. Then you can debug on the compute nodes like this. 🎉

