



MAGEMin framework

(Mineral Assemblage Gibbs Energy Minimization)


MAGEMin



- MPI-parallel C code
- Point-wise minimization at given P-T-X



- Metapelite (White et al., 2014)
- Metabasite (Green et al., 2016)
- Igneous (Holland et al., 2018)
- Ultramafic (Evans & Forst, 2021)
- Mantle (Stixude & Lithgow-Bertelloni, 2010)

MAGEMin_C




- Julia wrapper of the C code
- Flexible programming interface
- Database calibration
- Geodynamic coupling

MAGEMinApp



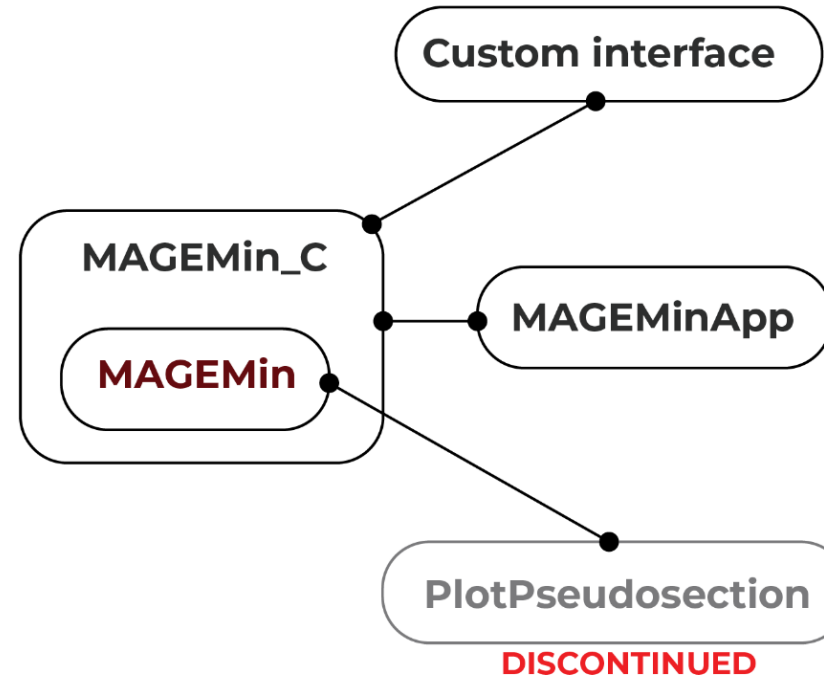
- Web browser app (graphic user interface)
- Parallel point-wise minimization
- PT, PX, TX and PT-X phase diagrams
- Auto labelling, contouring
- Fractional melting/crystallization paths

PlotPseudosection

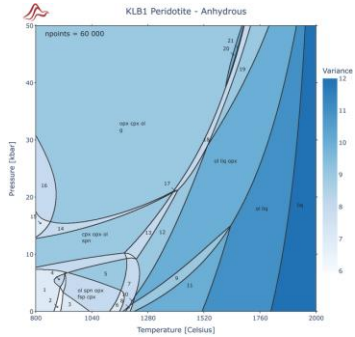


- Matlab app (Graphic user interface)
- Parallel point-wise minimization
- PT, PX, TX diagrams
- Contouring
- PT paths
- Trace element partitioning for mafic to ultramafic systems

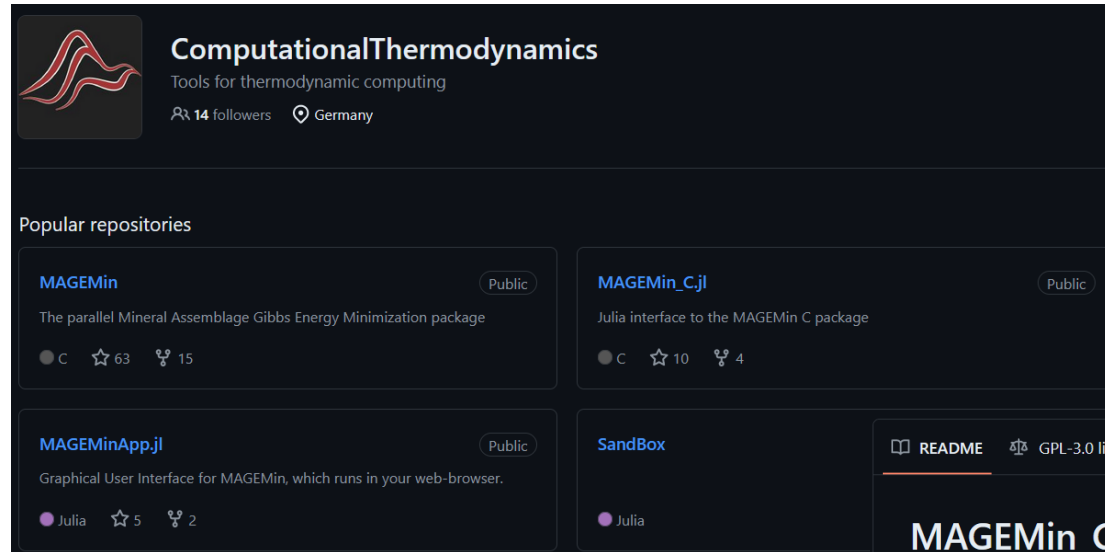
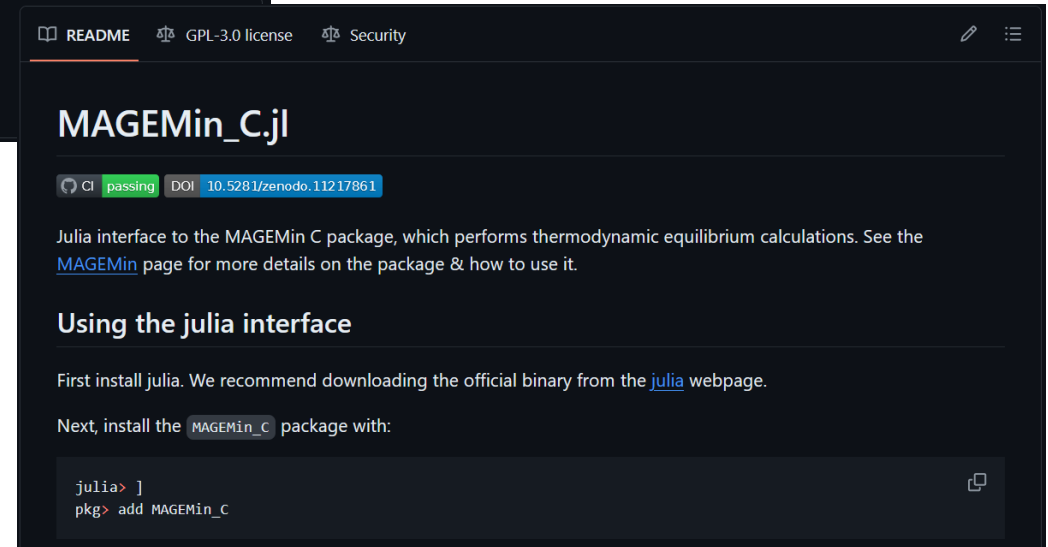
E.G., GEODYNAMIC COUPLING



MAGEMin github: ComputationalThermodynamics



nriel@uni-mainz.de

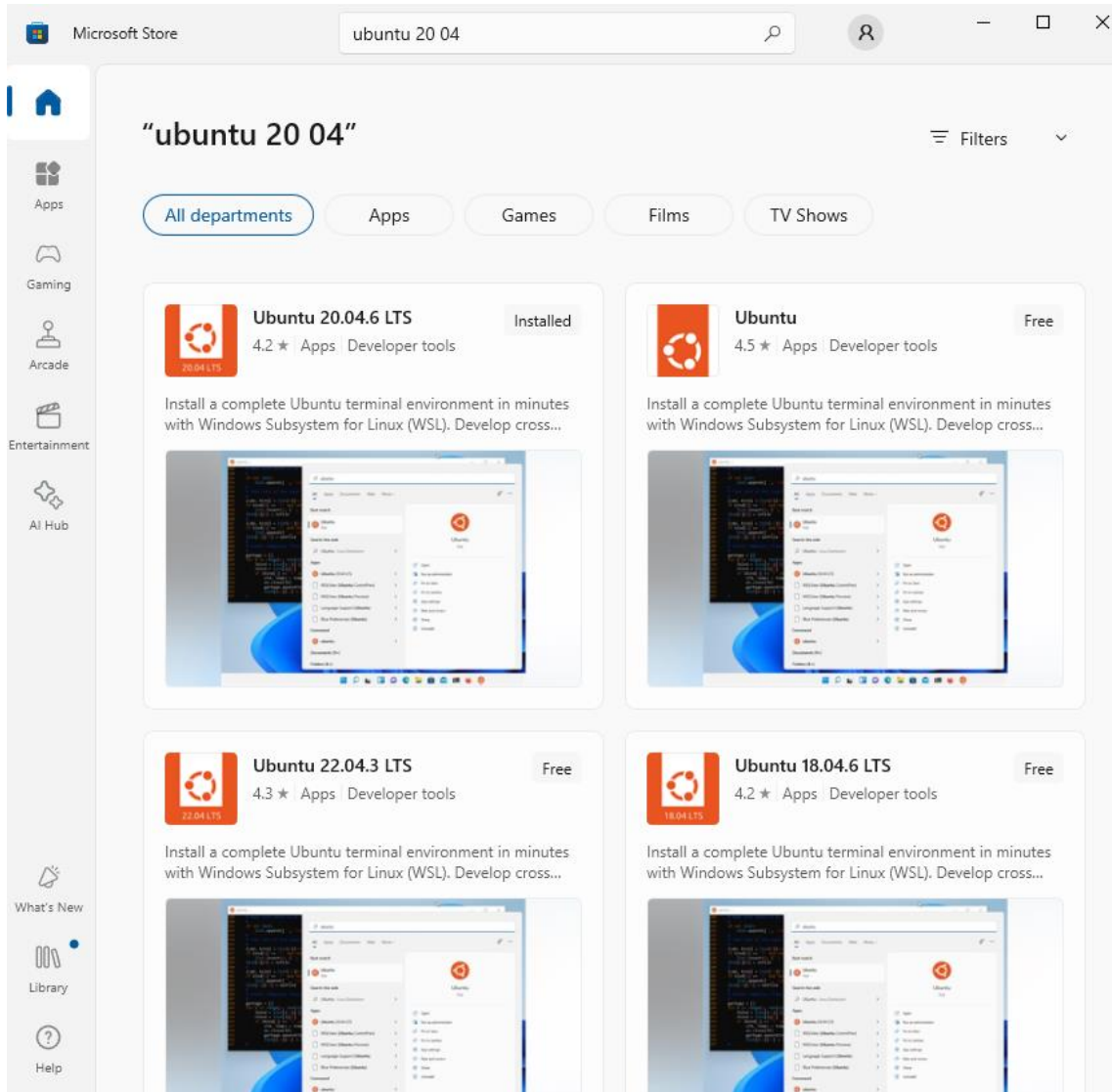
A screenshot of the GitHub profile for ComputationalThermodynamics. The profile shows the repository name, a description "Tools for thermodynamic computing", 14 followers, and location "Germany". Below, it lists popular repositories: MAGEMin (The parallel Mineral Assemblage Gibbs Energy Minimization package, 63 stars, 15 forks), MAGEMin_C.jl (Julia interface to the MAGEMin C package, 10 stars, 4 forks), MAGEMinApp.jl (Graphical User Interface for MAGEMin, which runs in your web-browser, 5 stars, 2 forks), and SandBox (Julia).A screenshot of the README for the MAGEMin_C.jl repository. It shows the repository name, a badge for CI passing, a DOI link, and a description: "Julia interface to the MAGEMin C package, which performs thermodynamic equilibrium calculations. See the MAGEMin page for more details on the package & how to use it." Below, it has a section "Using the julia interface" with instructions to install julia and then the MAGEMin_C package using the command: `julia>]
pkg> add MAGEMin_C`

<https://github.com/ComputationalThermodynamics>

https://github.com/ComputationalThermodynamics/MAGEMin_C

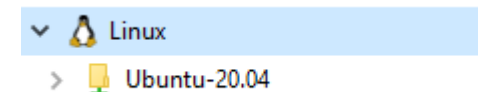
<https://github.com/ComputationalThermodynamics/MAGEMinApp>

MAGEMinApp installation (Windows with WSL)



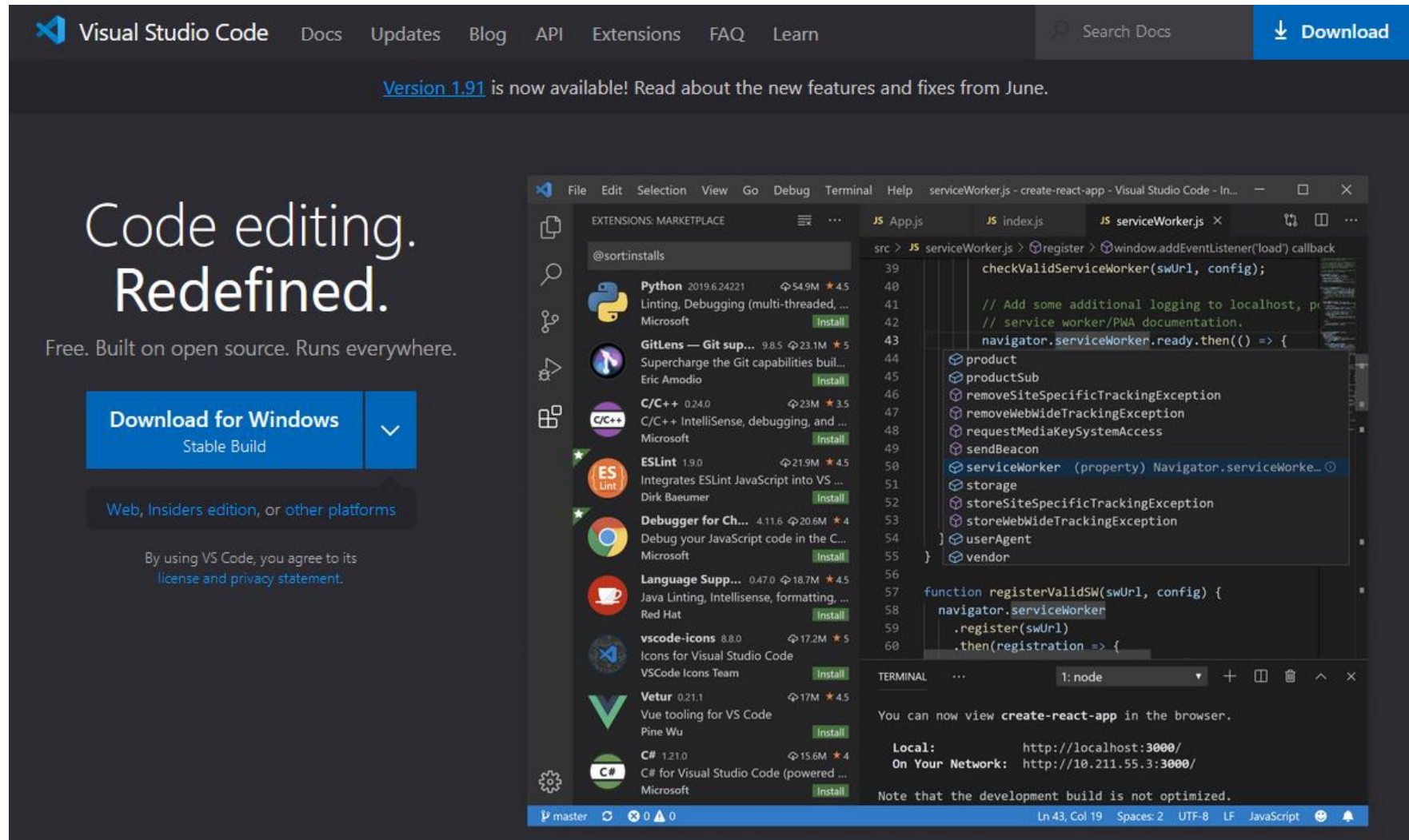
- Open Microsoft store (start-up menu)
- Look for Ubuntu 20 04
- Install Ubuntu 20 04
- Restart computer
- A terminal will open and ask for setting up a Linux username and password to your Ubuntu

- A new folder in the explorer should appear:



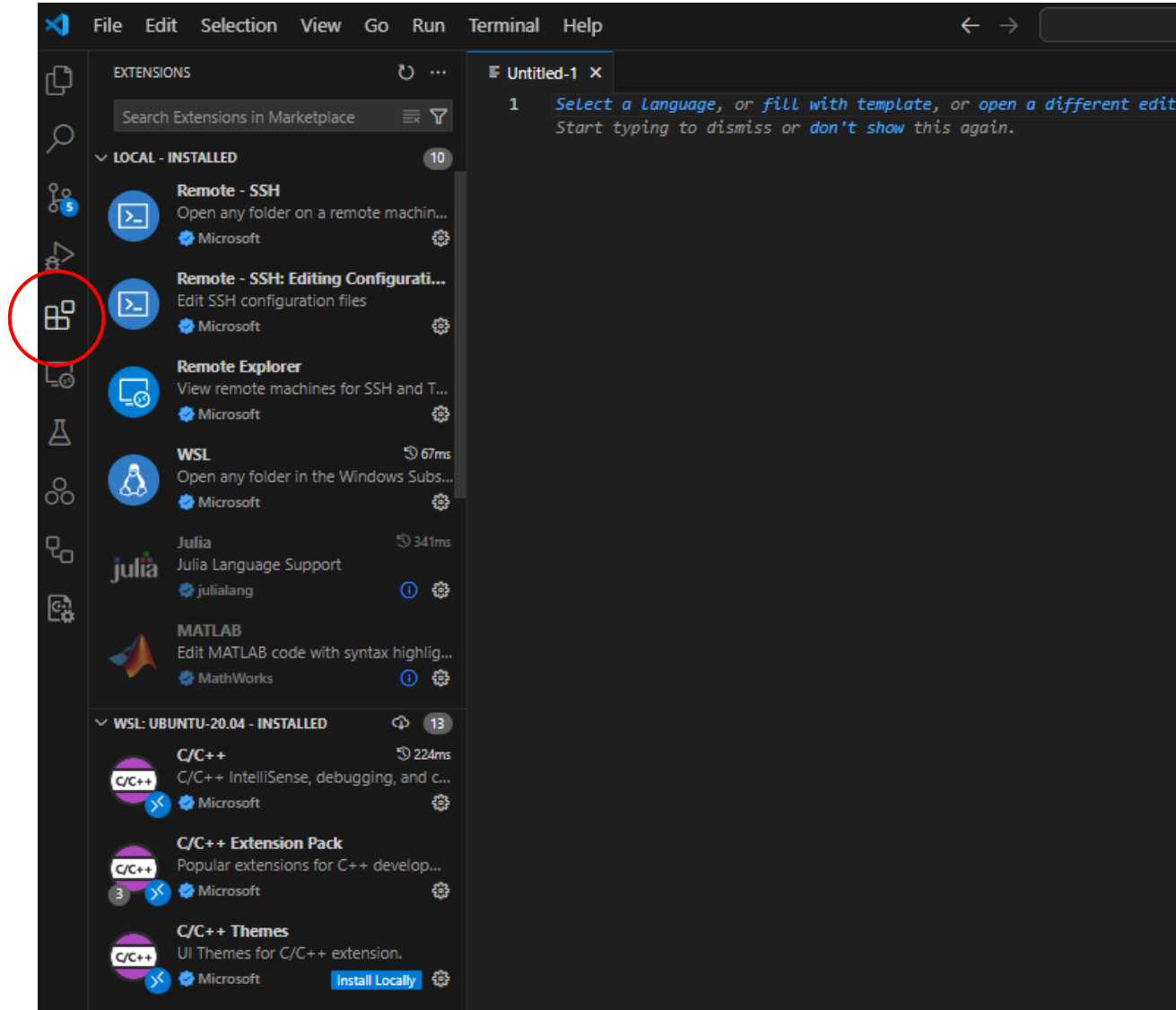
Visual Studio Code installation

- <https://code.visualstudio.com/>
- Download and install



Visual Studio Code installation

- Install WSL plugin



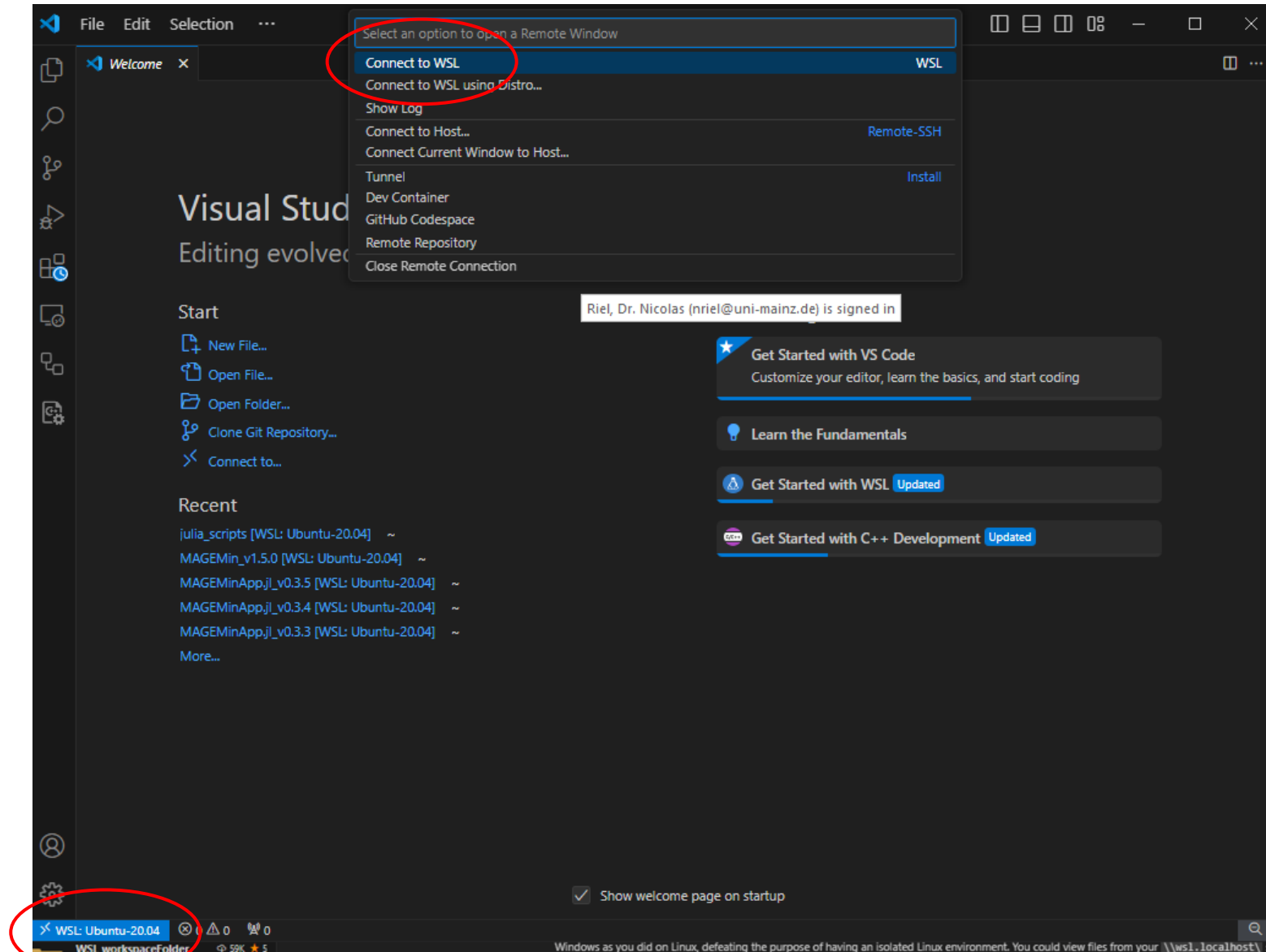
Visual Studio Code installation

- Install WSL plugin

The screenshot shows the Visual Studio Code interface with the Extensions Marketplace open. On the left, a list of extensions is shown, with 'WSL' by Microsoft at the top. The main panel displays the details for the 'WSL' extension (version 0.88.2). The extension description states: 'Open any folder in the Windows Subsystem for Linux (WSL) and take advantage of Visual Studio Code's full feature set.' Below the description, there are buttons for 'Disable' and 'Uninstall'. The 'Uninstall' button is circled in red. The right panel shows the 'Visual Studio Code WSL' documentation, which includes sections like 'Why do I need the WSL extension?' and 'Why WSL?'. The documentation explains that WSL allows running a Linux environment on Windows without a traditional virtual machine, and that the WSL extension enables using VS Code in WSL just as you would from Windows.

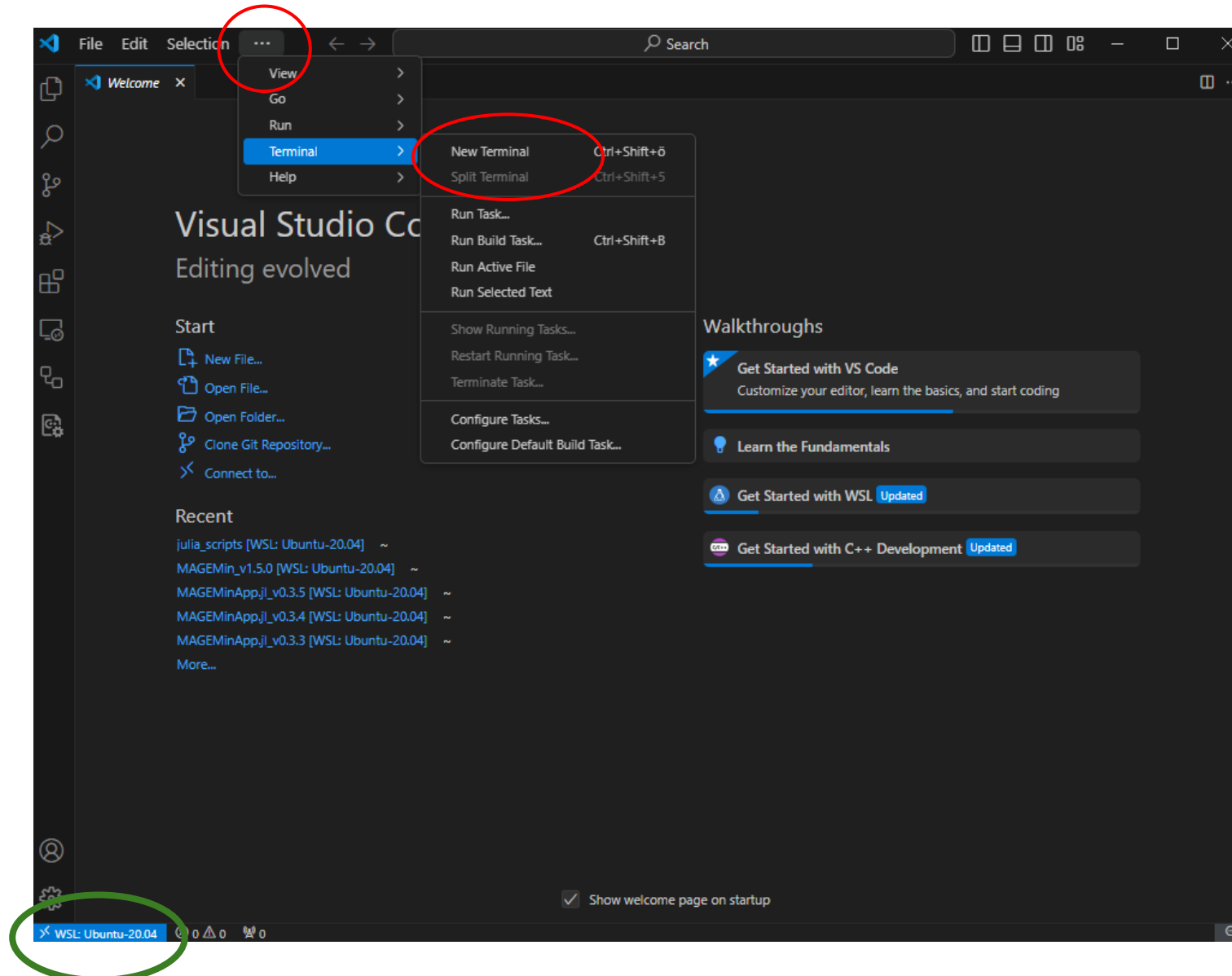
Visual Studio Code installation

- Connect vscode to WSL



Visual Studio Code installation

- Open a new terminal



Julia installation

- Get the command to download Julia:
`curl -fsSL https://install.julialang.org | sh`

<https://julialang.org/downloads/>



Download

Documentation

Learn

Blog

Community

Contribute

JSOC

Install julia

Install the latest Julia version (v1.10.4 June 4, 2024) by running this in your terminal:

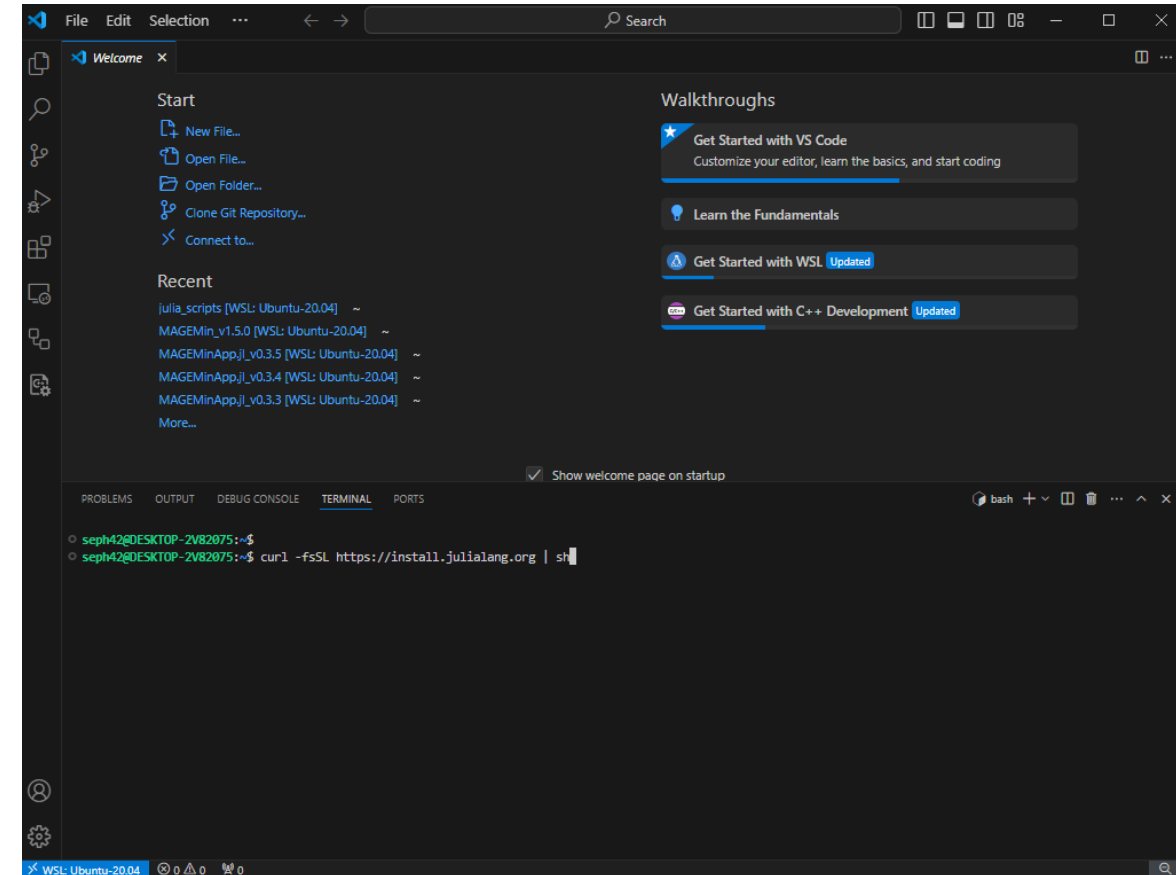
```
$ curl -fsSL https://install.julialang.org | sh
```

For Windows instructions [click here](#)

Once installed **julia** will be available via the command line interface.

This will install the **Juliaup** installation manager, which will automatically install julia and help keep it up to date. The command is installed. To install different julia versions see **juliaup --help**.

Please star us [on GitHub](#). If you use Julia in your research, please [cite us](#). If possible, do consider [sponsoring us](#).



Please do not use the version of "Julia" shipped by unix package managers

Many unix package managers ship broken and/or significantly out of date versions of Julia. Please use juliaup or download the c

- Execute the command in the terminal

Install MAGEMinApp

- Close and open a new terminal (this makes Julia available)
- Type:
julia -t 6 # where 6 is the number of core you want to use (depends on your machine, type
 # 'versioninfo()' to get more informations)

```
seph42@DESKTOP-2V82075:~$  
seph42@DESKTOP-2V82075:~$ julia -t 6  
  
      _  
   _(_)_  
  / __ \ |__| |  
 | | | | | | | |  
 | |_| | | | | |  
  \___/ |__|_|  
       /_/_/  
Documentation: https://docs.julialang.org  
Type "?" for help, "]"? for Pkg help.  
Version 1.10.0 (2023-12-25)  
Official https://julialang.org/ release  
  
(v1.10) pkg> add MAGERMinApp
```

- In the terminal type ']', this will open the package manager
- Type 'add MAGEMinApp', this will download and install MAGEMinApp
- Once installed, quite the package manager by typing 'BACKSPACE' key

Launch MAGEMinApp

- In the Julia terminal, type:
‘using MAGEMinApp’
Then
‘App()’
- The following text will be displayed in the terminal:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
seph42@DESKTOP-2V82075:~$ julia -t 6

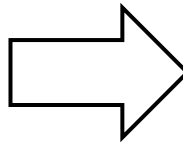
Documentation: https://docs.julialang.org
Type "?" for help, "]" for Pkg help.

Version 1.10.0 (2023-12-25)
Official https://julialang.org/ release

(julia) pkg> add MAGEMinApp
Resolving package versions...
No Changes to `~/julia/environments/v1.10/Project.toml`
No Changes to `~/julia/environments/v1.10/Manifest.toml`

julia> using MAGEMinApp
Using libMAGEMin.dylib from MAGEMin_jll

julia> App()
[ Info: Listening on: 127.0.0.1:8050, thread id: 1
```



- Copy and paste the address in your web-browser:

