



**Amrita School of Artificial
Intelligence**

COIMBATORE CAMPUS

ONE-DAY WORKSHOP

AI in Material Science

Participant guide for the one-day
hands-on workshop at the University of Calicut

Prepared by **Dr. Abhijith Anandakrishnan**

Assistant Professor, Amrita School of Artificial Intelligence

DATE

Wednesday, 24 September 2026

VENUE

Senate hall, University of Calicut

SESSIONS

09:00 to 12:30 and 14:00 to 17:00

BRING

A laptop, set up as described

OFFERING

University of Calicut, 24 September 2026

Participant guide

The afternoon is hands-on and runs on your own laptop. Two hours of preparation at home is what makes the afternoon work; there is no time on the day to install software from scratch.

What to bring

- A laptop you are allowed to install software on, with at least 8 GB of RAM and 10 GB of free disk space. Any operating system will do, as long as you follow the matching route below.
- Its charger.
- The API key from Step 5, saved somewhere you can find it.

Morning: MATLAB or GNU Octave

The morning scripts run in MATLAB and in GNU Octave. If your department has a MATLAB licence you can use it. Otherwise install GNU Octave, which is free:

Windows and macOS	Download the installer from <code>octave.org/download</code> and accept the defaults. Version 8 or later.
--------------------------	---

Ubuntu and Debian Linux	<code>sudo apt install octave</code>
--------------------------------	--------------------------------------

Open Octave, type `pinv([1 2; 3 4])` and press Enter. If a 2 by 2 matrix prints, you are ready for the morning.

Afternoon: a Linux environment with Python

The afternoon uses Python on Linux. Pick the route for your machine. Do Steps 1 to 6 in order; each takes a few minutes and Step 6 tells you whether everything worked.

Step 1. Get a Linux terminal

Linux	Nothing to do. Open your usual terminal.
macOS	Nothing to do. Open Terminal (Applications, Utilities). Everything below works as is.
Windows 10 or 11	Install WSL 2 (Windows Subsystem for Linux), which runs Ubuntu inside Windows. Open PowerShell as Administrator (Start, type <code>powershell</code> , right-click, <i>Run as administrator</i>) and run <code>wsl --install</code> . Restart when asked. On restart an Ubuntu window opens and asks you to choose a username and password. From then on, "the terminal" means this Ubuntu window: find it as Ubuntu in the Start menu.
Windows, if WSL cannot be installed	Use a Linux live USB. Download Ubuntu 24.04 LTS from <code>ubuntu.com/download/desktop</code> , write it to an 8 GB or larger USB stick with Rufus (<code>rufus.ie</code>), and boot the laptop from the stick (usually F12, F2 or Esc at power-on). Choose <i>Try Ubuntu</i> . Everything installed in a live session is lost on shutdown, so do Steps 2 to 6 again on the morning of the workshop, or ask the organisers for the offline package. WSL 2 is the better route if it works for you.

NOTE

Windows users: the commands below are typed into the **Ubuntu** window, not into PowerShell or Command Prompt. If your prompt ends with `$` you are in the right place.

On Ubuntu (native, WSL or live USB), first make sure the basics are present:

```
sudo apt update && sudo apt install -y curl git build-essential
```

Step 2. Install `uv`

`uv` creates the Python environment and installs Python itself if you do not have a suitable version. Run this in the terminal:

```
curl -LsSf https://astral.sh/uv/install.sh | sh
```

Then close the terminal and open it again so the `uv` command is found. Check with `uv --version`.

Step 3. Create the workshop environment

```
mkdir -p ~/ai-materials && cd ~/ai-materials
uv venv --python 3.12
source .venv/bin/activate
uv pip install rdkit pandas scikit-learn matplotlib jupyter ase
uv pip install langchain langchain-google-genai langchain-groq langgraph
```

The two install lines download a few hundred megabytes and takes a few minutes on a good connection. Whenever you open a new terminal for the workshop, run `cd ~/ai-materials` and `source .venv/bin/activate` first; the prompt then starts with `(ai-materials)`.

Step 4. Install a code editor

We recommend **Visual Studio Code** (`code.visualstudio.com`) with the *Python* and *Jupyter* extensions from its Extensions panel. Windows users install VS Code on Windows as usual, add the WSL extension as well, and then open the workshop folder from the Ubuntu window with `code ~`
`~/ai-materials`.

Any editor you already use is fine (Sublime Text, PyCharm, Vim, Emacs). Notebooks can also be run from the terminal with `jupyter lab`, which opens in your browser.

Step 5. Create a free API key for a language model

Hands-on 2 sends requests to a hosted language model. Create one free key before the day. No card is needed and nothing is charged.

Google AI Studio (recommended)

Sign in at `aistudio.google.com` with a Google account, choose *Get API key*, then *Create API key*. Copy the key.

Groq (alternative)

Sign up at `console.groq.com`, open *API Keys*, then *Create API Key*. Copy the key.

Save the key in the workshop folder so the code can find it:

```
echo 'GEMINI_API_KEY=paste-your-key-here' > ~/ai-materials/.env
```

(For a Groq key, write `GRQ_API_KEY=` instead.) Treat the key like a password: do not post it in a group chat or commit it to a public repository.

Step 6. Check that everything works

With the environment active (Step 3), run:

```
python -c "import rdkit, sklearn, ase, langchain; from rdkit import Chem;
print(Chem.MolToSmiles(Chem.MolFromSmiles('CCO')), 'ready')"
```

If it prints `CCO ready`, your laptop is ready for the afternoon. If it prints an error, send a screenshot of the whole terminal window to `a_abhijith@cb.amrita.edu` with the subject *Calicut workshop setup*, and we will sort it out before the day.

Step 7 (optional). ChemGraph

The demonstration at the end of Hands-on 2 uses ChemGraph, a research-agent framework from Argonne National Laboratory. It is a large install (it pulls in PyTorch and a machine-learned potential, several gigabytes), so it is not required. If you have a fast connection and want to try it yourself, install it into its own environment:

```
mkdir -p ~/chemgraph && cd ~/chemgraph
uv venv --python 3.12 && source .venv/bin/activate
uv pip install "chemgraph==0.7.0"
```

On the day

Arrive with the laptop charged and the environment from Step 3 already created. Open the terminal, activate the environment, and keep the API key handy. The notebooks and data are handed out on a USB stick and on the workshop repository at the start of the afternoon, so nothing needs to be downloaded during the session apart from the language model's replies.

Questions on any of this: Dr. Abhijith Anandakrishnan, a_abhijith@cb.amrita.edu .