ME 5374-ST

Machine Learning for Materials Science and Discovery

Fall 2025

Asst. Prof. Peter Schindler

Lecture 2 – Python Crash Course 1

- Python, IDEs, and Libraries/Modules
- Package Managers
- GitHub
- Markdown and Jupyter Notebooks
- *Interactive*: Setting up VSCode and UV
- *Interactive*: Data types in Python
- *Interactive*: Logic in Python (loops, statements, etc.)

Python Overview

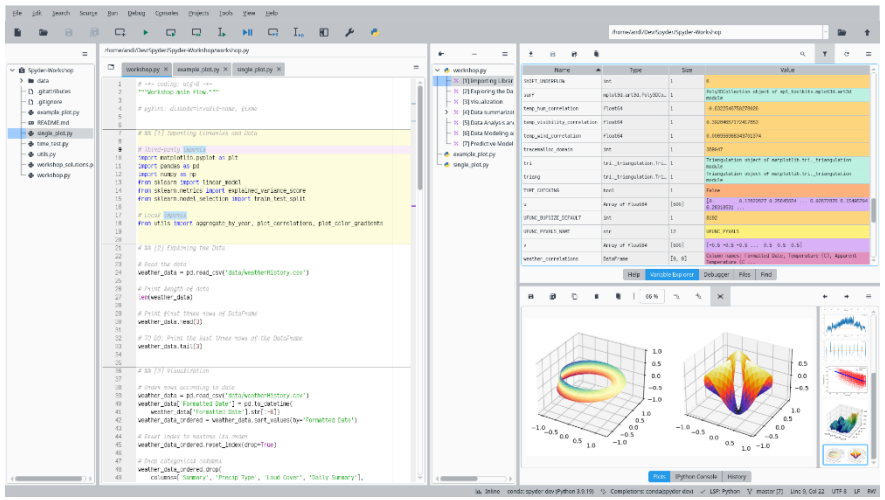
- General-purpose, high-level language
- **Interpreted** (not compiled) → runs line by line
- **Dynamically typed** → no need to declare variable types
- Beginner-friendly & readable, **lightweight syntax**
- Cross-platform & **open-source**
- *Massive ecosystem*: AI, data science, web, automation
- Large global community & industry adoption

Common Python IDEs (Integrated Development Environment)

PyCharm

- Great for Python-focused work
- Not as customizable

Spyder



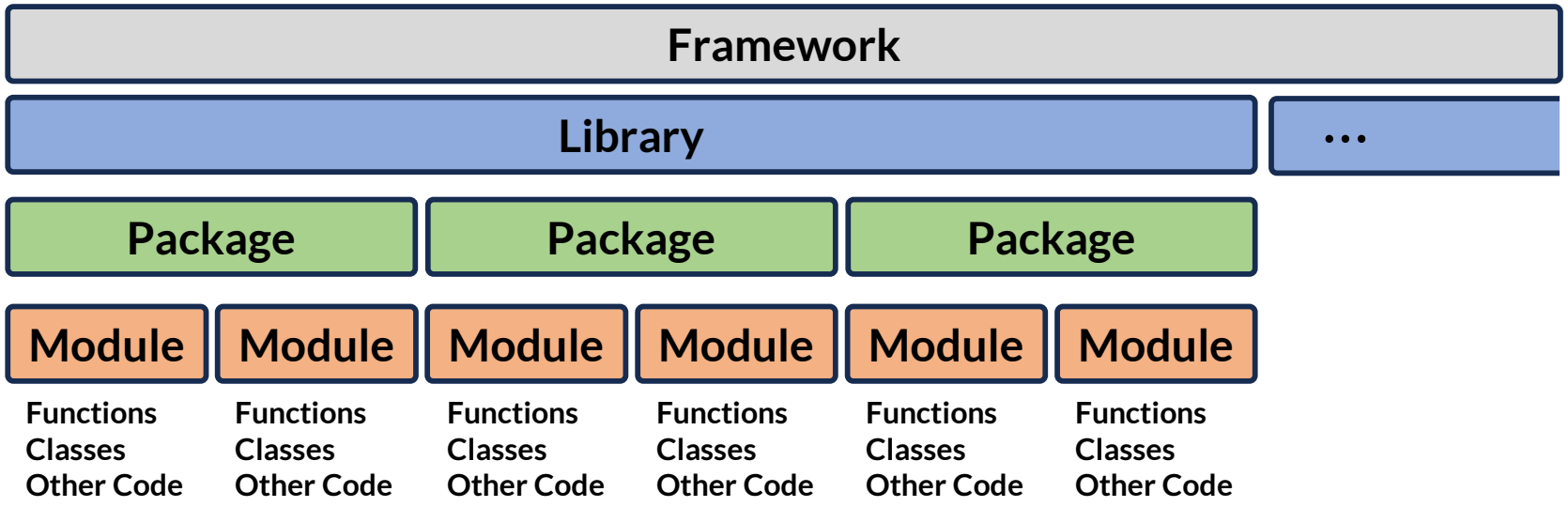
- MATLAB-like

Visual Studio Code (VS Code)

Or with AI-integration: Cursor, Winsurf, Void,...

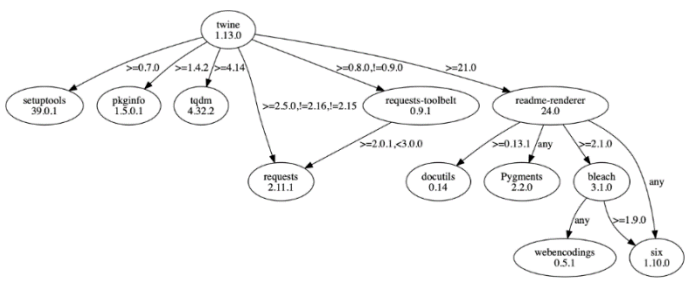
- Highly customizable through plugins
- Works also for other languages

Open-source Community: Python Libraries




```
numpy/   ← Library (top-level)
│
├── core/        ← Package
│   ├── numeric.py    ← Module
│   ├── shape_base.py ← Module
│   └── __init__.py
│
├── linalg/      ← Package
│   ├── linalg.py     ← Module
│   └── __init__.py
│
├── random/      ← Package
│   ├── bit_generator.py ← Module
│   ├── mt19937.py       ← Module
│   └── __init__.py
│
└── __init__.py
```

Main Python Ecosystem Challenge:
Each library/package typically depends on **specific versions** of other libraries/packages
→ Dependency tree



Package Managers: Dependency Management

pip

- Python's *default package manager*
- Installs packages from **PyPI** (Python Package Index)
- Works with **venv** or virtualenv for isolation
- Pros: Lightweight, standard, huge ecosystem, built-in with Python.
- Cons: Doesn't handle non-Python dependencies; slow; sometimes not reproducible

---

Conda

- Focused on **data science & ML**.
- Installs packages from Anaconda repo (or conda-forge distribution)
- Built-in environment management
- Pros: Handles scientific stack easily (NumPy, pandas, scikit-learn, TensorFlow, etc.), **including non-Python dependencies** (like C/C++ libraries, compilers, and CUDA)
- Cons: Heavy environments (~3GB), slower updates, larger footprint

---

UV

- Fast & reliable **replacement** for pip + venv (built in Rust)
- Pros: Super fast (10–100x faster than pip) and reproducible, works with pyproject.toml (modern packaging), operates **outside** of Python (*version independent*)
- Cons: Newer, no support for non-Python backend dependencies

---

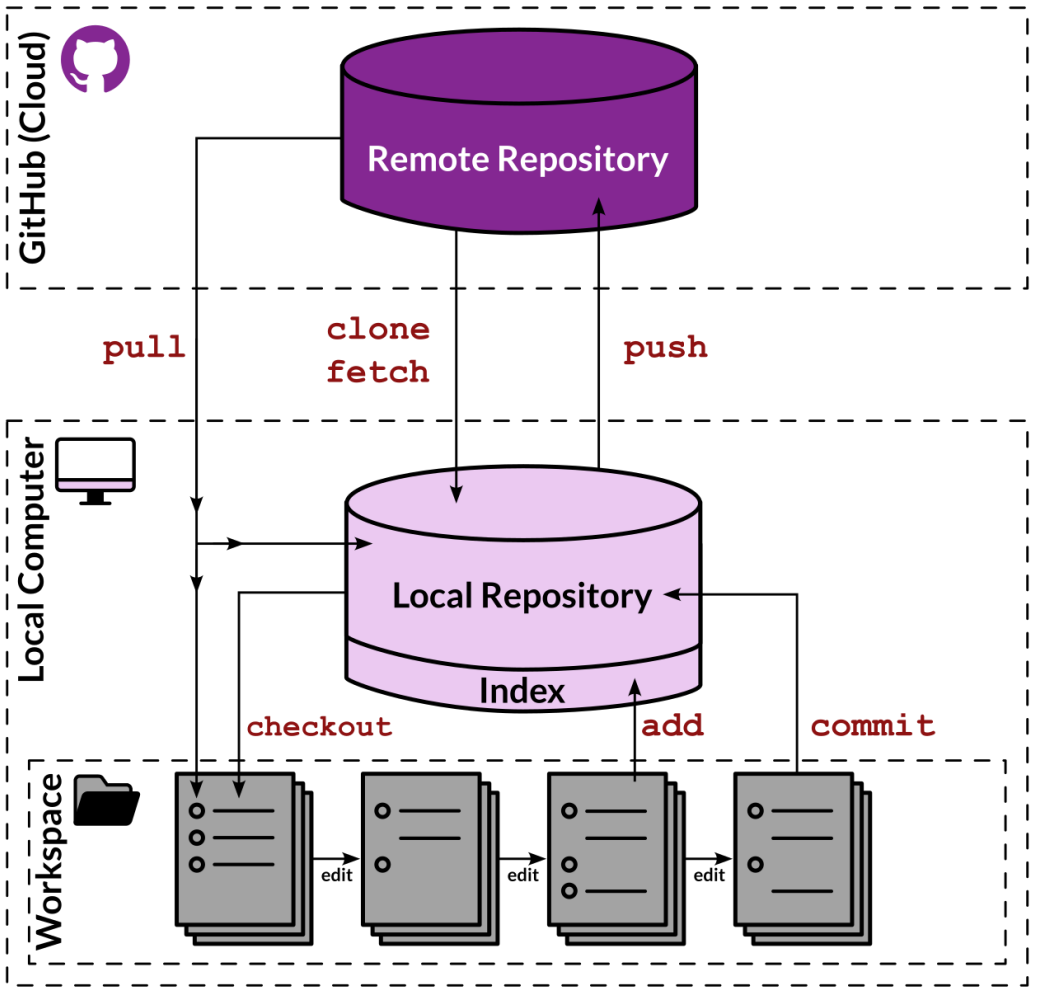
Others Micromamba, pipx, pixi

GitHub

- git is a software for **version control** of files
- On GitHub, you can host your code in a remote "*repository*"
- Keep **track of all changes** and ability to revert to any previous state
- Ideal for **collaborative** code development
- **.gitignore** file contains the types of files that git should not keep track of

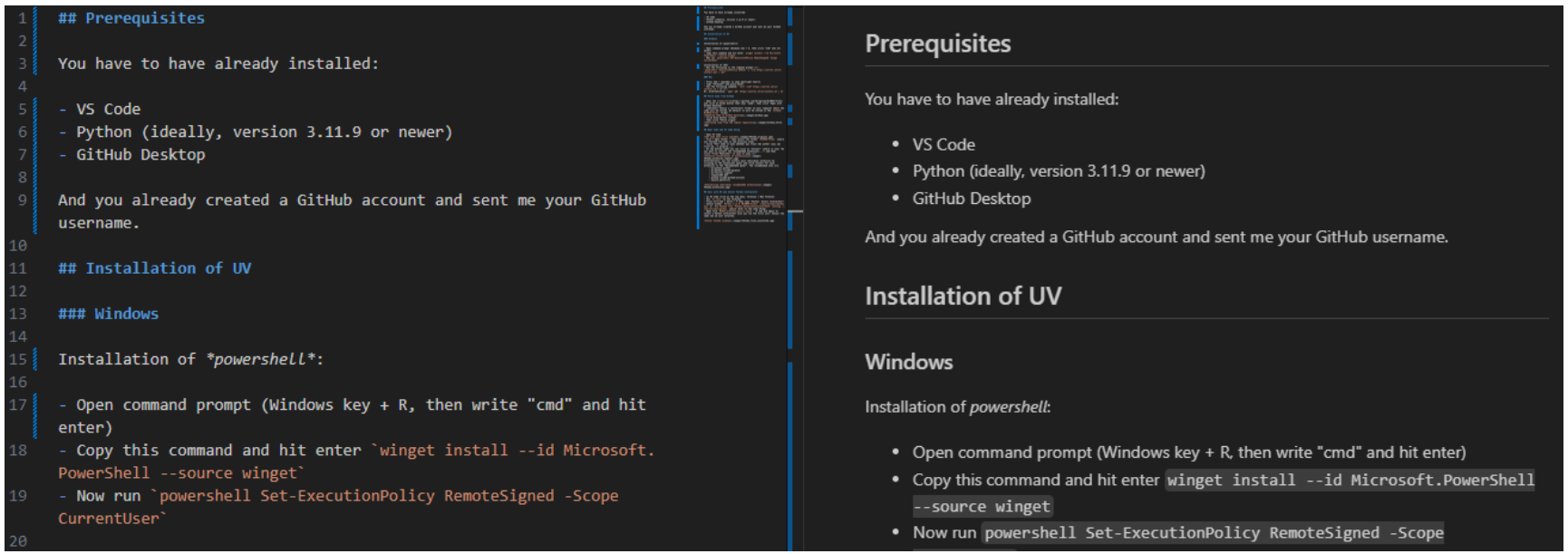
```
"git" can mean anything, depending on your mood.

- random three-letter combination that is pronounceable, and not
actually used by any common UNIX command.  The fact that it is a
mispronunciation of "get" may or may not be relevant.
- stupid. contemptible and despicable. simple. Take your pick from the
dictionary of slang.
- "global information tracker": you're in a good mood, and it actually
works for you. Angels sing, and a light suddenly fills the room.
- "goddamn idiotic truckload of sh*t": when it breaks
```



Markdown and Jupyter Notebooks

Markdown is similar to HTML but much simpler (used on GitHub for .md files)



```
## Prerequisites

You have to have already installed:

- VS Code
- Python (ideally, version 3.11.9 or newer)
- GitHub Desktop

And you already created a GitHub account and sent me your GitHub
username.

## Installation of UV

### Windows

Installation of *powershell*:

- Open command prompt (Windows key + R, then write "cmd" and hit
enter)
- Copy this command and hit enter `winget install --id Microsoft.
PowerShell --source winget`
- Now run `powershell Set-ExecutionPolicy RemoteSigned -Scope
CurrentUser`
```

Prerequisites

You have to have already installed:

- VS Code
- Python (ideally, version 3.11.9 or newer)
- GitHub Desktop

And you already created a GitHub account and sent me your GitHub username.

Installation of UV

Windows

Installation of *powershell*:

- Open command prompt (Windows key + R, then write "cmd" and hit enter)
- Copy this command and hit enter `winget install --id Microsoft.PowerShell --source winget`
- Now run `powershell Set-ExecutionPolicy RemoteSigned -Scope`

Jupyter is an interactive environment that combines markdown text, code, and visualizations in a *notebook*

Markdown Syntax

Here are a few main syntax examples for Markdown (as summarized by ChatGPT)

- **Headings:**

```md
# H1
## H2
### H3
```

- **Bold / Italic:**

```md
**bold** or __bold__
*italic* or _italic_
***bold italic***
```

- **Lists:**

```md
- Item 1
- Item 2
  - Subitem

1. First
2. Second
```

- **Blockquote:**

```md
> This is a quote
```

- **Horizontal rule:**

```md
---
```

- **Strikethrough:**

```md
~~strikethrough~~
```

- **Code:**

Inline: `` `code` ``

Block:

````
```python
print("Hello")
```
````

- **Links & Images:**

```md
[OpenAI](https://openai.com)
![Alt text](image.png)
```

Markdown Tables with Alignment

```md
| Name   | Age | City        |
|:-------|:---:|------------:|
| Alice  | 24  | New York    |
| Bob    | 30  | San Diego   |
| Carol  | 27  | Los Angeles |
```

Alignment explained:

- `:---` → left-aligned
- `:---:` → centered
- `---:` → right-aligned

Lecture Feedback

Please, scan the QR code and take a minute to let me know how the lecture was and mention any **feedback/questions**

This form is **anonymous!**