Practical session 1

***Everybody has to start somewhere* - Haruki Murakami**

Development Tools for Scientific Computing 2024/2025

Pasquale Claudio Africa, Dario Coscia

13 Feb 2025

Part 1: Setting up your development environment

1. **Install required tools**:
    - Ensure that Python is installed on your system.
    - Install Git, and create a GitHub account if you haven't already.
    - Install Miniconda on your system (refer to the installation guide here).

2. **Set up a `git` repository and virtual environment**:
    - Open your terminal and create a conda environment named `devtools_scicomp` with Python 3.9 by running:

      ```
      conda create --name devtools_scicomp python=3.9
      ```

    - Install essential packages by running:

      ```
      python -m pip install pytest
      ```

      Note: Always install packages inside the conda environment using `python -m pip`, which ensures that you're using the Python interpreter and `pip` package manager within the environment.

    - Open GitHub in your web browser and follow these steps:
        a. Create a new empty repository titled `devtools_scicomp_project_2025`, and select a specific license and the Python `.gitignore` file.
        b. Open your terminal or command prompt, clone the repository, create a `README.md` file, and push your changes to the origin with the commit message `first commit`.

Part 2: Structuring the package

- Inside your cloned GitHub repository, create the following directories: `src/pyclassify/`, `scripts`, `test`, `shell`, and `experiments`.
- Inside the `src/pyclassify/` directory, create an `__init__.py` and `utils.py` file.
- Inside the `scripts/` directory, create a `run.py` file.
- Inside the `shell/` directory, create a `submit.sbatch` and a `submit.sh` file.
- Inside the `experiments/` directory, create a `config.yaml` file.
- Inside the `test/` directory, create a `test_.py` file.
- Generate a `requirements.txt` file from the `devtools_scicomp` conda environment by running:

  ```
  python -m pip freeze > requirements.txt
  ```

  Add the `requirements.txt` file to the root of your project folder (`devtools_scicomp_project_2025/`).

- Create a `pyproject.toml` file for your project from this template, and complete the `[INSERT]` placeholders.
- Add in the `.gitignore` file the removal of `.dat` and `.data` files.
- Add, commit, and push these changes to the origin repository with the commit message `structuring the package`.

Solutions

The repository with the right structure and commits is available here: GitHub repo