

# **Effective Programming Practices for Economists**

## **Installation and execution**

### **Environment files and environments**

Janoś Gabler and Hans-Martin von Gaudecker

# Environment files vs. environments

- An **environment.yml** file:
  - A recipe to create an environment
  - Under version control
  - Shared with team members
- A conda/mamba **environment**:
  - A collection of Python packages stored on your computer
  - Independent of other environments you might have
- You create an environment from an environment file by typing

```
mamba env create -f environment.yml
```

# Anatomy of environment.yml

```
name: mini-env
channels:
  - conda-forge
  - nodefaults
dependencies:
  - python==3.11
  - pandas
  - pip:
    - pdbp
```

- Name can be anything, except that it must not contain spaces
- Channels: You rarely need more than conda-forge
- Dependencies: Python, list of Python packages
  - Can have equality or inequality constraints on versions
  - Only use pip for packages that are not conda/mamba installable
- Environment files should be hand-written and maintained

# Dependencies installed automatically

- The mini environment from above will contain the following packages
- Most are low-level dependencies you don't have to care about
- Some exact versions are OS specific
- Listing all of them would create a non-portable environment

<code>_libgcc_mutex</code>	<code>0.1</code>	<code>numpy</code>	<code>1.26.2</code>
<code>_openmp_mutex</code>	<code>4.5</code>	<code>openssl</code>	<code>3.2.0</code>
<code>bzip2</code>	<code>1.0.8</code>	<code>pandas</code>	<code>2.1.4</code>
<code>ca-certificates</code>	<code>2023.11.17</code>	<code>pdbp</code>	<code>1.5.0</code>
<code>ld_impl_linux-64</code>	<code>2.40</code>	<code>pip</code>	<code>23.3.1</code>
<code>libblas</code>	<code>3.9.0</code>	<code>pygments</code>	<code>2.17.2</code>
<code>libcblas</code>	<code>3.9.0</code>	<code>python</code>	<code>3.11.0</code>
<code>libffi</code>	<code>3.4.2</code>	<code>python-dateutil</code>	<code>2.8.2</code>
<code>libgcc-ng</code>	<code>13.2.0</code>	<code>python-tzdata</code>	<code>2023.3</code>
<code>libgfortran-ng</code>	<code>13.2.0</code>	<code>python_abi</code>	<code>3.11</code>
<code>libgfortran5</code>	<code>13.2.0</code>	<code>pytz</code>	<code>2023.3.post1</code>
<code>libgomp</code>	<code>13.2.0</code>	<code>readline</code>	<code>8.2</code>
<code>liblapack</code>	<code>3.9.0</code>	<code>setuptools</code>	<code>68.2.2</code>
<code>libns1</code>	<code>2.0.1</code>	<code>six</code>	<code>1.16.0</code>
<code>libopenblas</code>	<code>0.3.25</code>	<code>tabcompleter</code>	<code>1.3.0</code>
<code>libsqlite</code>	<code>3.44.2</code>	<code>tk</code>	<code>8.6.13</code>
<code>libstdcxx-ng</code>	<code>13.2.0</code>	<code>tzdata</code>	<code>2023c</code>
<code>libuuid</code>	<code>2.38.1</code>	<code>wheel</code>	<code>0.42.0</code>
<code>libzlib</code>	<code>1.2.13</code>	<code>xz</code>	<code>5.2.6</code>
<code>ncurses</code>	<code>6.4</code>		

# How to add a package

- Just installing packages into an environment from the shell is not reproducible
- Instead:
  - Add the package to `environment.yml`
  - Use `mamba env update -f environment.yml`
- If you have any problem:
  - `mamba env remove -n mini-env`
  - `mamba env create -f environment.yml`