

Effective Programming Practices for Economists

Basic Python

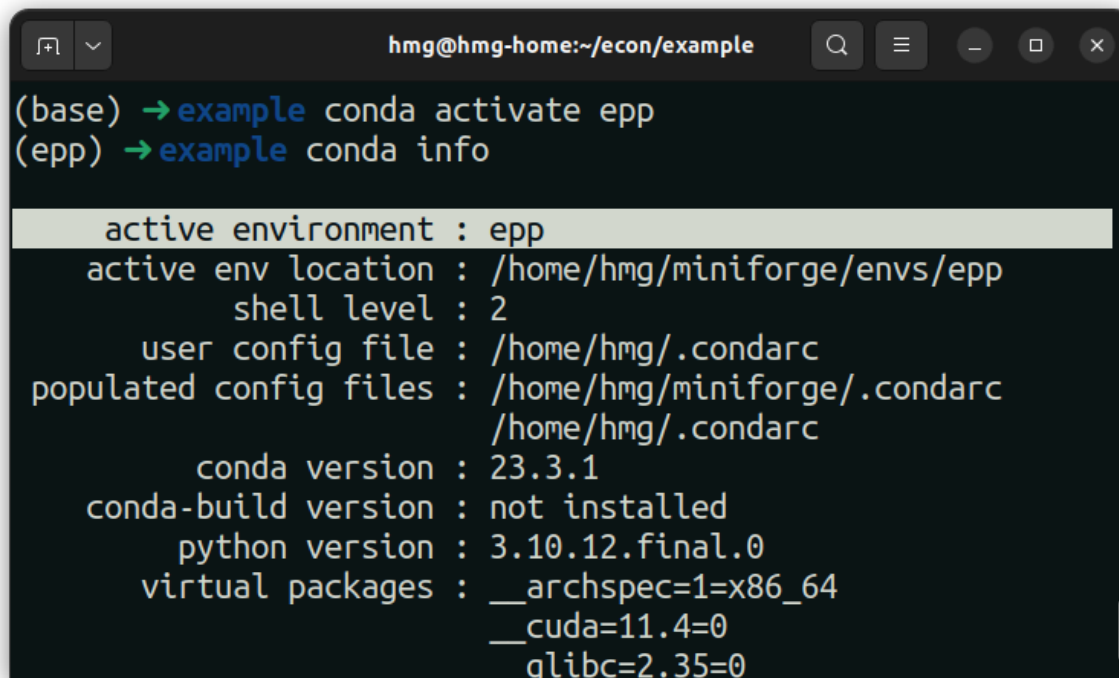
Running Python code via pytest

Janoś Gabler and Hans-Martin von Gaudecker

Preparation

- We assume you have installed anaconda and created the course environment
- Open a shell in the root directory of your project
 - On Windows, use the anaconda prompt or the powershell
 - If conda is not recognized in the powershell, check out this [stackoverflow post](#)
- Activate the environment using `conda activate epp`
- Confirm the activation worked using `conda info`

0: Activate and Info



```
hmg@hmg-home:~/econ/example
(base) → example conda activate epp
(epp) → example conda info

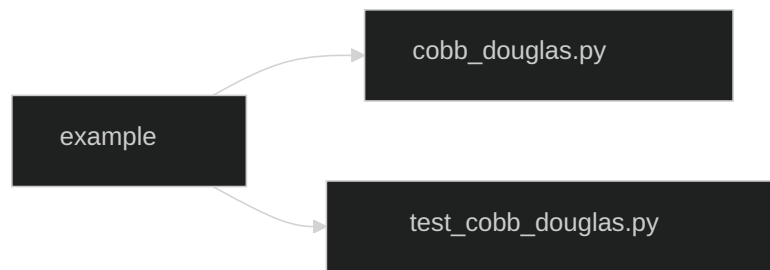
active environment : epp
  active env location : /home/hmg/miniforge/envs/epp
    shell level      : 2
  user config file   : /home/hmg/.condarc
populated config files : /home/hmg/miniforge/.condarc
                      /home/hmg/.condarc
   conda version     : 23.3.1
conda-build version  : not installed
  python version     : 3.10.12.final.0
 virtual packages    : __archspec=1=x86_64
                      __cuda=11.4=0
                      __libc=2.35=0
```

How does pytest execute code?

- Executing .py files: Run the entire file
- Executing notebooks: Run individual cells
- Pytest: Run individual functions in multiple .py files

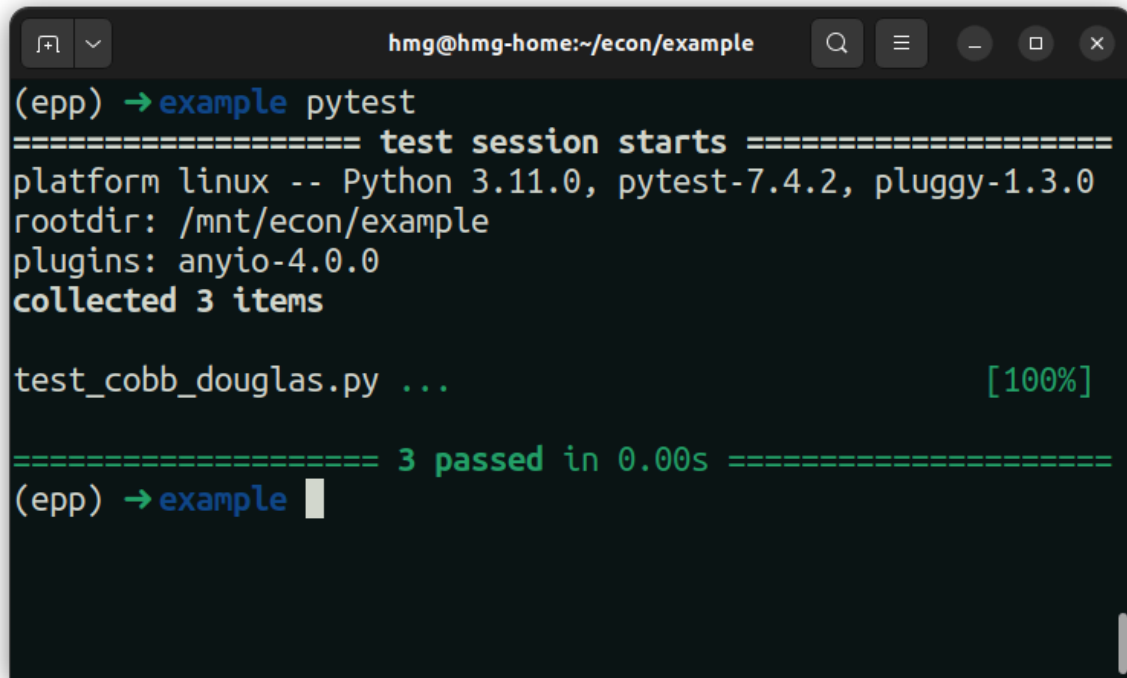
Very useful for automating test execution across an entire project

Example Project Structure



- Our shell is in the `example` directory
- We want to run all functions that start with `test_` in `test_cobb_douglas.py`
- Could be spread across multiple test files, across different subdirectories
- Command is `pytest`

1: Execute

A terminal window with a dark background and light-colored text. The window title is 'hmg@hmg-home:~/econ/example'. The terminal shows the command '(epp) → example pytest' being executed. The output includes a separator line '==== test session starts ===', environment details like 'platform linux -- Python 3.11.0, pytest-7.4.2, pluggy-1.3.0', the root directory '/mnt/econ/example', the plugin 'anyio-4.0.0', and 'collected 3 items'. A progress bar for 'test_cobb_douglas.py' shows '[100%]'. The final result is '==== 3 passed in 0.00s ==='. The prompt '(epp) → example' is followed by a cursor.

```
hmg@hmg-home:~/econ/example
(epp) → example pytest
==== test session starts ====
platform linux -- Python 3.11.0, pytest-7.4.2, pluggy-1.3.0
rootdir: /mnt/econ/example
plugins: anyio-4.0.0
collected 3 items

test_cobb_douglas.py ... [100%]

==== 3 passed in 0.00s ====
(epp) → example
```