

16 – Code quality

Bálint Aradi

Scientific Programming in Python (2026)

<https://atticlectures.net/scipro/python-2026/>

Prerequisites

- pylint
- mypy
- black

```
conda install mypy pylint black
```

Code analysis with pylint and mypy

Pylint

- checks for possible **convention breaches**, **inconsistencies** and **errors**
- It produces a score for “**code quality**” (how much code aligns to pylint’s guidelines)

```
pylint mymath.py
```

```
-----  
Your code has been rated  
at 10.00/10  
(previous run: 10.00/10, +0.00)
```

Mypy

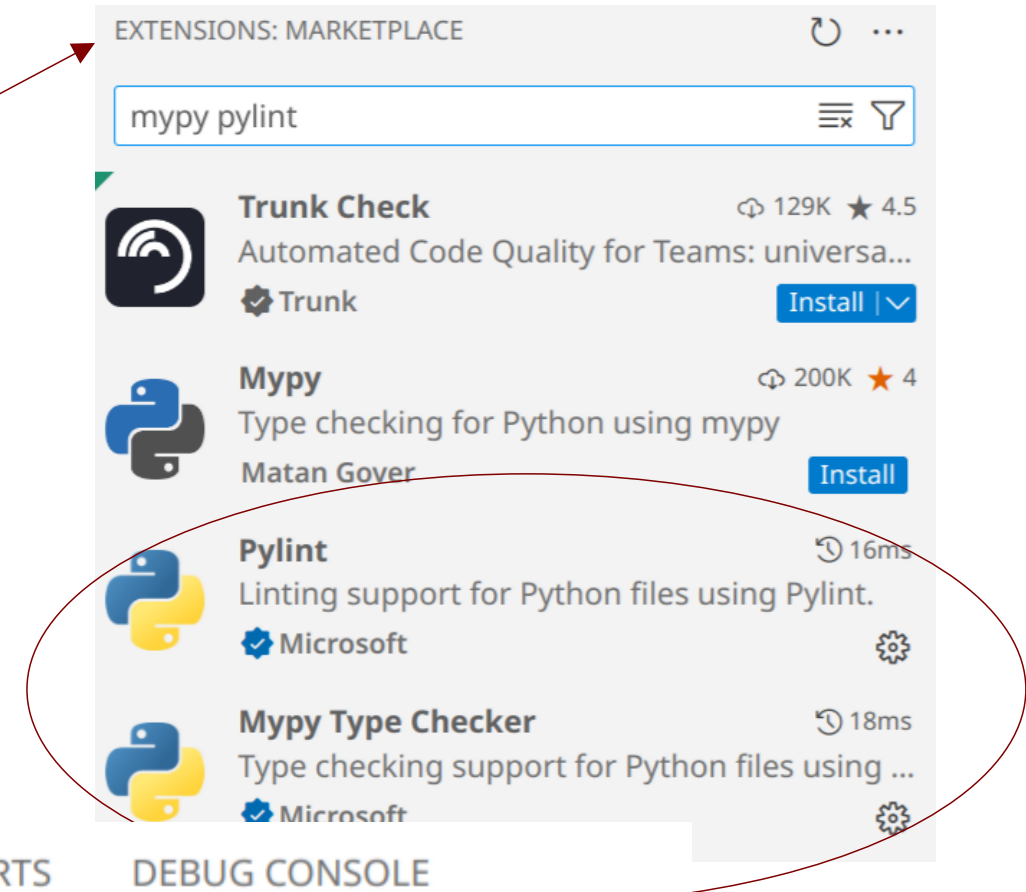
- checks for **type inconsistencies** and **type hinting correctness**

```
mypy mymath.py
```

```
Success: no issues  
found in 1 source file
```

Adding linters to Visual Studio Code

- Visual Studio Code invokes **Pyright** by default
- Further linters (mypy, pylint) can be added via **Extensions Marketplace (Ctrl-Shift-X)**
- All installed linters will be run (disable extension to exclude one)
- **Ctrl-Shift-M**: Opens **Problems window** with linter messages



PROBLEMS 3 OUTPUT TEST RESULTS TERMINAL PORTS DEBUG CONSOLE

test_solvers.py test 3

💡 Name "s" is not defined Mypy(name-defined) [Ln 23, Col 1]

⊗ Undefined variable 's' Pylint(E0602:undefined-variable) [Ln 23, Col 1]

⚠ Statement seems to have no effect Pylint(W0104:pointless-statement) [Ln 23, Col 1]

Python coding standard (PEP 8) & Black formatter

PEP 8 coding standard

- Python has a **widely accepted coding style guide**
- It has been documented in the **Python Enhancement Proposal 8 (PEP 8)**
- Most Python projects stick to that standard
- **Do not deviate from it** without very-very good reasons (maybe except for default line length) 😊

Black formatter

- Reformats code to be PEP 8 compatible (and makes some stylistic choices)

```
black -l 100 mymath.py
```

Custom line length
(default: 88)

```
reformatted mymath.py  
All done!  
1 file reformatted.
```

- Best time for reformatting: Before adding the file to the stage (git add)