Python code quality

Martijn Beenker
Lead Data Engineer @ Team Rockstars Data

Repo @ https://github.com/LowerKees/pyqa

THIS TALK IS FOR...
EVERYONE!

Not sure how to set up a project <> **Experience with setting up a project**

This talk is for you

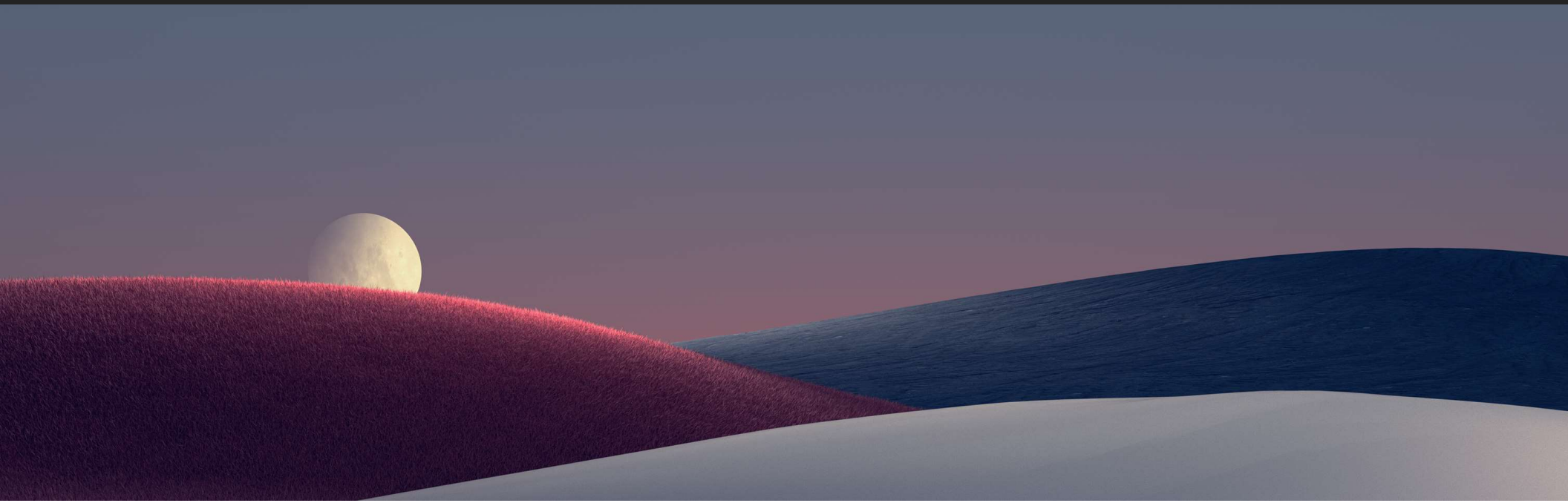
Share your own thoughts
and make this talk better

WELL STRUCTURED REPOS HELPS WHILE READING CODE.

MONO REPO vs POLY REPO

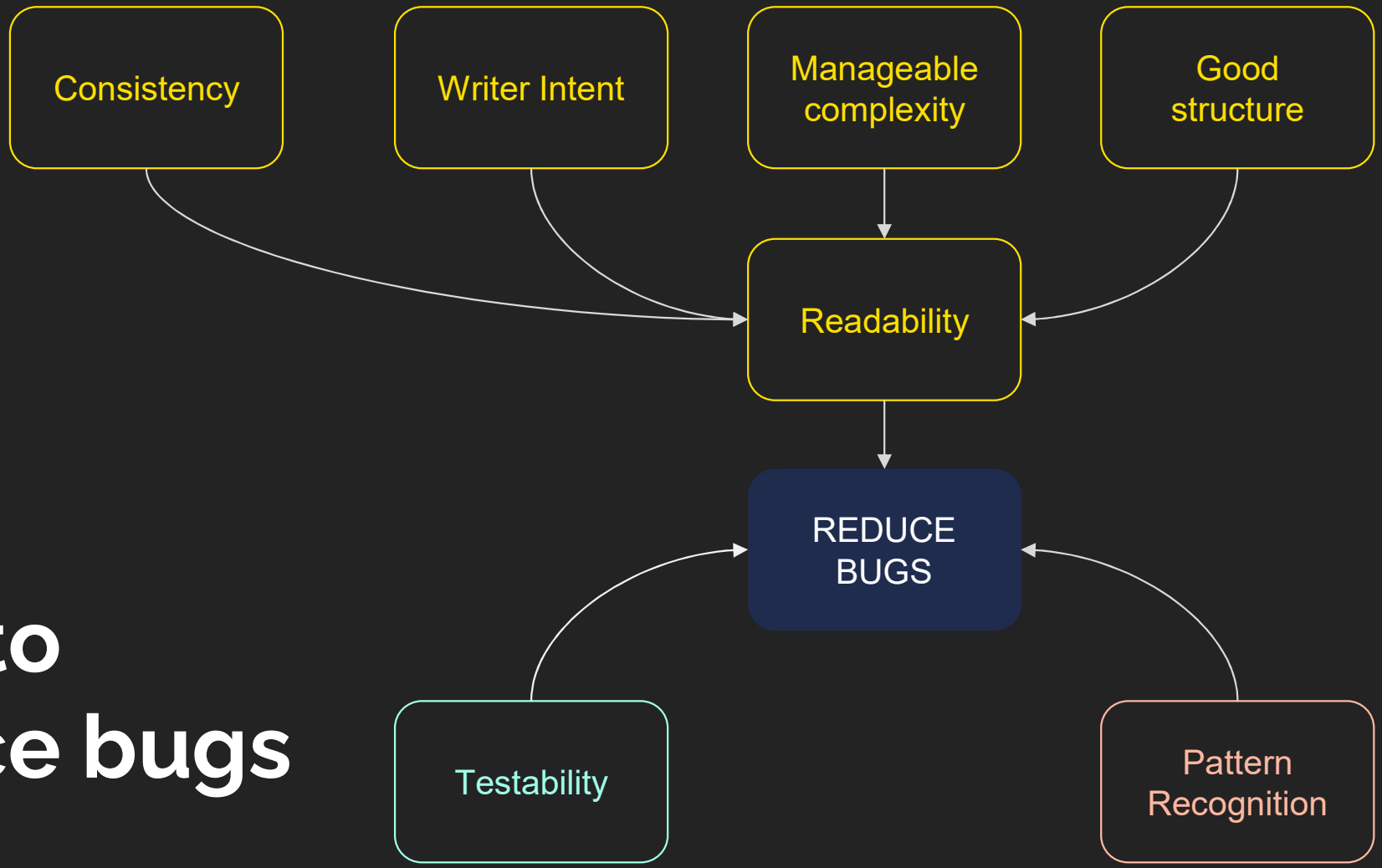
Mono repo	Poly repo	Considerations
Data product A Data product B Shared	Data product A Data product B Shared	+ Code visibility + Code sharing + **Faster workflow for making cross project changes** ± Standardization (unless) - Requires team maturity - Requires discipline - Can increase release complexity

What's code quality about?

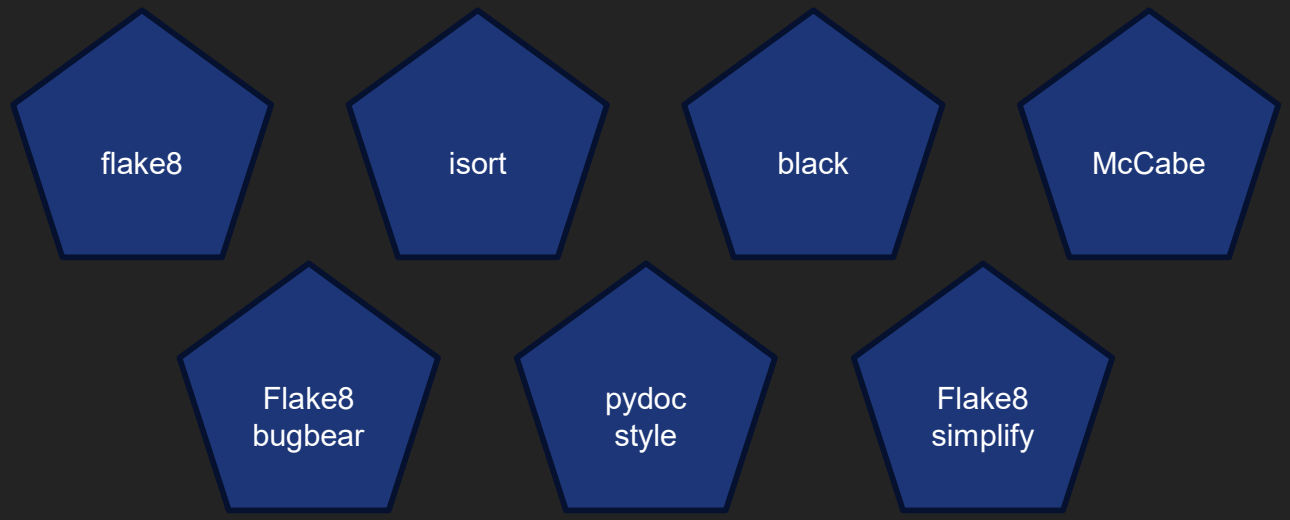


WRITE FUNCTIONING CODE, REDUCE BUGS & WRITE PERFORMANT CODE

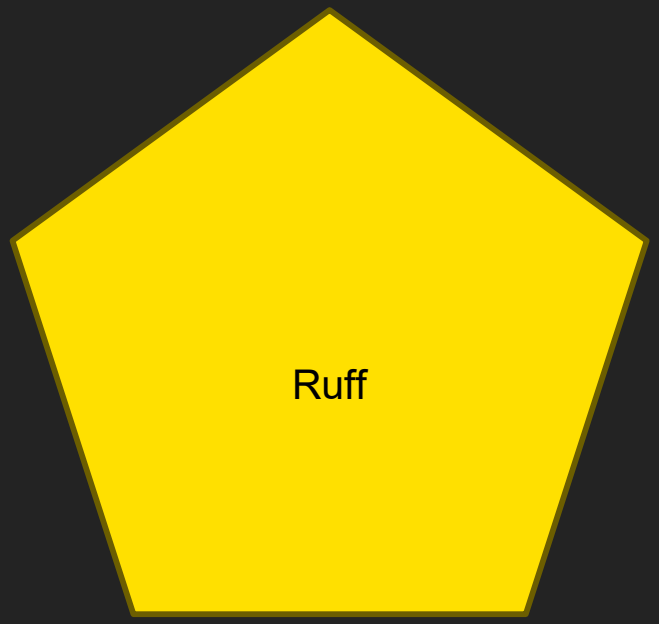
How to reduce bugs

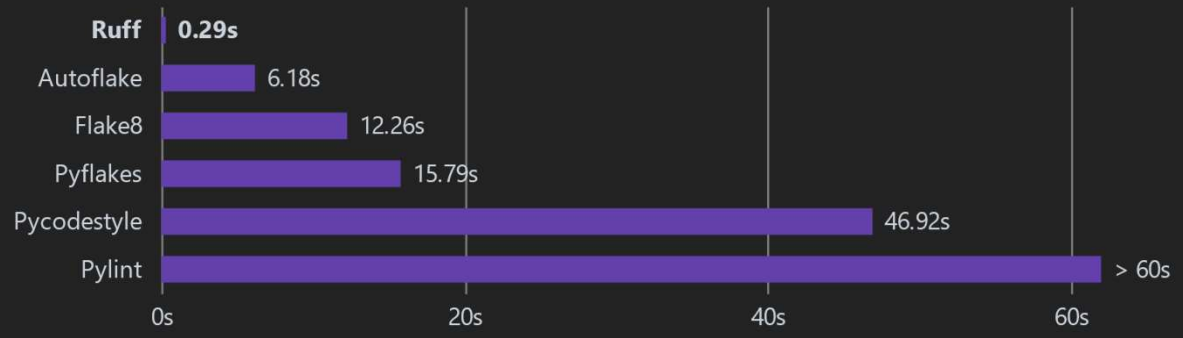


LINTING & EXTENSIONS



LINTING & EXTENSIONS





PREVENT BUGS FROM ENTERING THE CODE BASE INSTEAD OF GETTING THEM OUT

THIS IS MY FAVORITE REPO STYLE BECAUSE IT SHARES QA FUNCTIONALITY ACROSS PACKAGES & PROJECTS BUT LEAVES ROOM FOR EXCEPTIONS

APPENDIXES

Big matrix of purpose

Tool name	Ties into
MyPy	Clarifies writer intent
Pydocstyle (Ruff “D”)	Clarifies writer intent
Flake8 (Ruff defaults)	Consistency, clarifies writer intent
Flake 8 bug bear (Ruff “B”)	Pattern recognition
Repo layout	Good (and conventional) structure
PyCodeStyle (Ruff “E”)	Consistency
PyDocStyle (Ruff “D”	Clarifies writer intent
PyFlakes (Ruff “F”)	Pattern recognition
Black (Ruff with line length = 88)	Consistency
Isort (Ruff “I”)	Consistency
McCabe complexity (Ruff “C901”)	Manage complexity, increase testability

Docs

- Setup spark session with delta: https://docs.delta.io/latest/quick-start.html#python
- Rules of Ruff: https://docs.astral.sh/ruff/rules
- Python & spark compatibility matrix: https://community.cloudera.com/t5/Community-Articles/Spark-Python-Supportability-Matrix/ta-p/379144
- Testing pyspark 3.5: https://spark.apache.org/docs/latest/api/python/getting_started/testing_pyspark.html
- Python src repo layout: https://blog.ionelmc.ro/2014/05/25/python-packaging/#the-structure
- Python src repo layout 2: https://hynek.me/articles/testing-packaging/
- NASA using src layout: https://github.com/nasa/fprime/blob/035808df02706d405611b30efa396f8fb799e9a1/Gds/setup.py#L69
- What setuptools has to say about repo layout: https://setuptools.pypa.io/en/latest/userguide/package_discovery.html#automatic-discovery

Settings.json

```
  "[python]": {
    "editor.rulers": [88],
    "editor.formatOnSave": true,
    "editor.defaultFormatter":
"charliermarsh.ruff",
    "editor.codeActionsOnSave": {
      "source.organizeImports": "explicit"
    },
    "editor.formatOnType": true,
  },
```