



No , all

Introduction to PyTest

Simon Stone

Research Software Engineer for HPC and AI

Research Computing @ ITC, Dartmouth College

Introducing Research Software Engineering

Collaborative expertise in software engineering, designed to bridge the gap between innovative ideas and impactful outcomes. Our services include:

- 🤝 **Grant Proposal Consulting** to ensure accurate resource estimations and project feasibility.
- 🚀 **Rapid Prototyping** to refine concepts and explore solutions.
- 🛑 **Ongoing Application Support** and **Application Rehabilitation** for existing applications.
- 🌐 **Open-Source Releases** to share knowledge and contribute to the wider community.

Contact us today to discuss your project and discover how Research Software Engineering can be your trusted partner in innovation.



Why test your code?

🔥 If you don't test, you are testing in production



Why test

🔥 If you don't



Why test your code?

- 🔥 If you don't test, you are testing in production
- 😊 Ensure desired behavior
- 🐛 Discover and squash bugs
- 🌱 Grow your codebase without breaking existing things
- 📄 Tests document how your code can be used
- 💡 Tests can help discover awkward interfaces or coupling

Test-Driven Development

👉 Core flow:

✍️ Write tests first

Fail all tests

🧒 Implement the simplest code to pass all tests

Pass all tests

Refactor the implementation as needed

🤔 See also: “Code towards an interface”, Backward Design

Beck, K. (2003). *Test-driven development : by example* (1st edition). Addison-Wesley.

Why PyTest?

 PyTest is an open-source testing framework for Python

 Easy and unintrusive to write simple tests

 Elegantly scales to more complex testing scenarios

 A community standard



pytest



Let's get started!





Hands-on



Summary

- 🌱 Tests are great!
- 📖 PyTest lets you automate and scale testing in an unintrusive way
- 🌱 Test-Driven Development is a useful paradigm to thoughtfully grow your codebase

What's next?

-  Explore the PyTest documentation
-  Automate testing with GitHub Actions
-  Learn about Mocking
-  Actually write (any kind of) tests!



Thank you

dartgo.org/intro-to-pytest