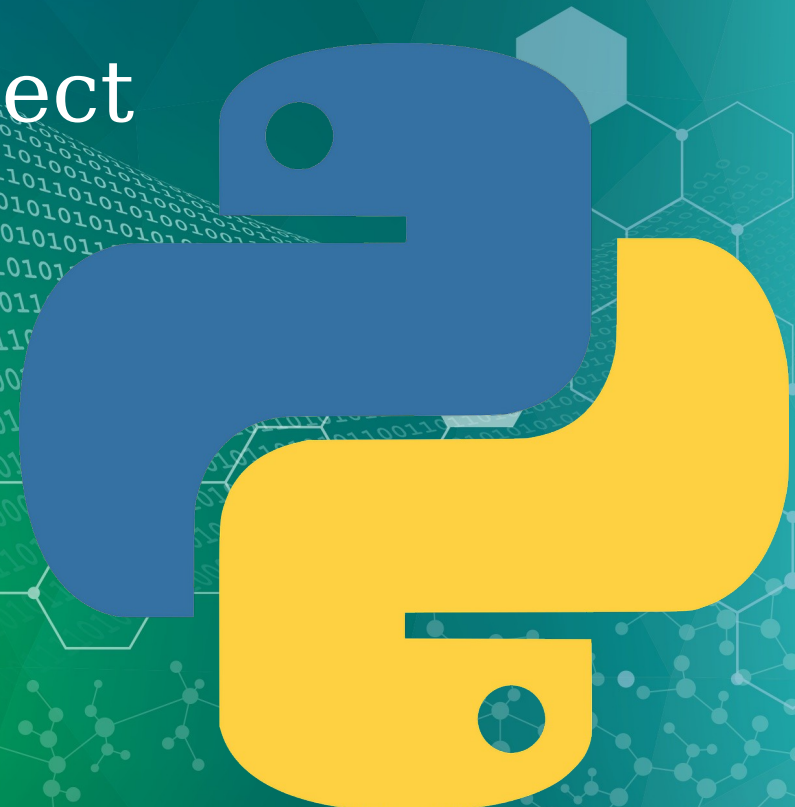


# Preferred Practices Through a Project Template

Peter F. Peterson, Chen Zhang, Jose M.  
Borreguero-Calvo, and Kevin A. Tactac



ORNL is managed by UT-Battelle, LLC for the US Department of  
Energy

# Preferred Practices



# Choose an open source license

An open source license protects contributors and users. Businesses and savvy developers won't touch a project without this protection.

{ Which of the following best describes your situation? }



**I need to work in a community.**

Use the **license preferred by the community** you're contributing to or depending on. Your project will fit right in.

If you have a dependency that doesn't have a license, ask its maintainers to **add a license**.



**I want it simple and permissive.**

The **MIT License** is short and to the point. It lets people do almost anything they want with your project, like making and distributing closed source versions.

**Babel**, **.NET**, and **Rails** use the MIT License.



**I care about sharing improvements.**

The **GNU GPLv3** also lets people do almost anything they want with your project, *except* distributing closed source versions.

**Ansible**, **Bash**, and **GIMP** use the GNU GPLv3.

# GitHub template repository





## General

---

### Repository name

### **Template repository**

Template repositories let users generate new repositories with the same directory structure and files. [Learn more about template repositories.](#)

### **Require contributors to sign off on web-based commits**

Enabling this setting will require contributors to sign off on commits made through GitHub's web interface. Signing off is a way for contributors to affirm that their commit complies with the repository's terms, commonly the [Developer Certificate of Origin \(DCO\)](#). [Learn more about signing off on commits.](#)

## Default branch

---

The default branch is considered the “base” branch in your repository, against which all pull requests and code commits are automatically made, unless you specify a different branch.



# Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

---

*Required fields are marked with an asterisk (\*).*

## Repository template

 neutrons/python\_project\_template ▾


Start your repository with a template repository's contents.

**Include all branches**

Copy all branches from neutrons/python\_project\_template and not just the default branch.

---

**Owner \***

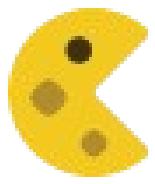
 neutrons ▾

**Repository name \***

/

Great repository names are short and memorable. Need inspiration? How about **verbose-octo-disco** ?

**Description (optional)**



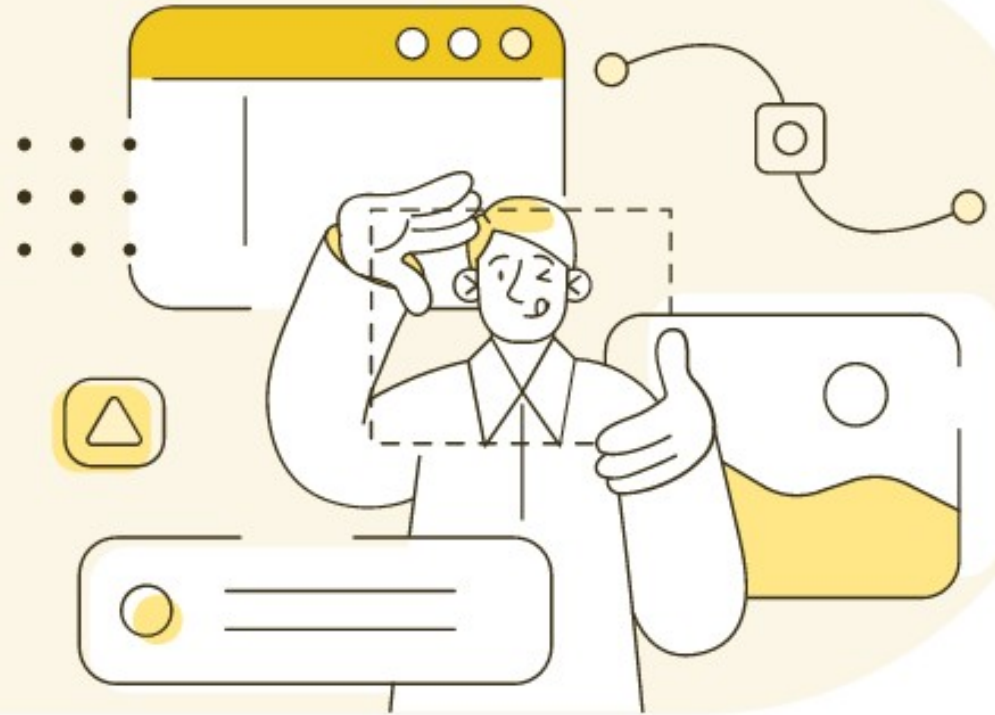
# COOKIECUTTER

## Save time and stay consistent

Cookiecutter is an open source library for building coding project templates

[Browse templates](#)

[Learn more](#)



# Template contents





# OpenSSF Best Practices Badge

- Basics
- Change Control
- [Bug and Vulnerability] Reporting
- Quality
- Security
- [Static and Dynamic] Analysis



# Static analysis configuration

python\_project\_template / .pre-commit-config.yaml

pre-commit-ci[bot] [pre-commit.ci] pre-commit autoupdate ✓

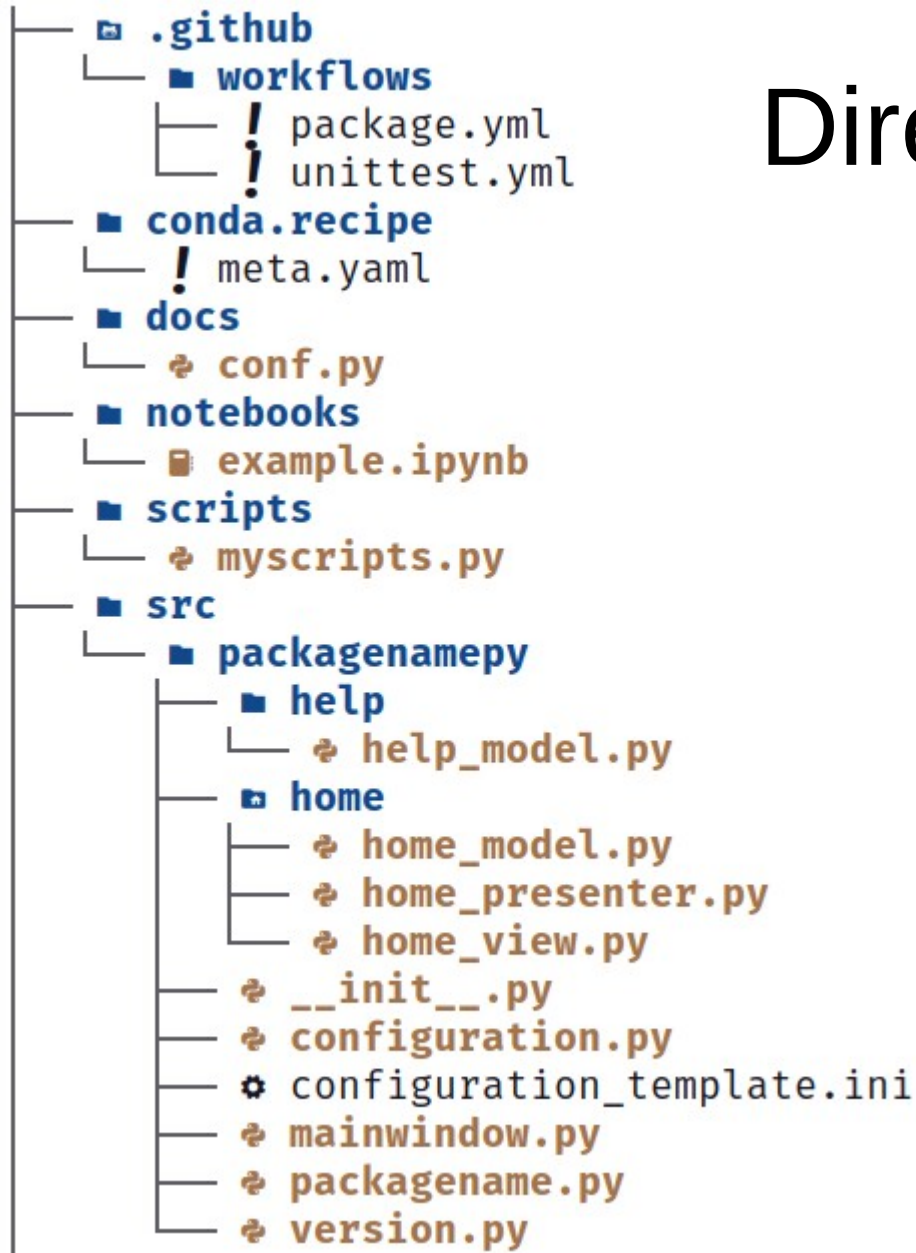
Code Blame 22 lines (22 loc) · 638 Bytes

```
1 repos:
2   - repo: https://github.com/pre-commit/pre-commit-hooks
3     rev: v4.6.0
4     hooks:
5       - id: check-added-large-files
6         args: [--maxkb=8192]
7       - id: check-merge-conflict
8       - id: check-yaml
9         args: [--allow-multiple-documents]
10        exclude: "conda.recipe/meta.yaml"
11       - id: end-of-file-fixer
12        exclude: "tests/cis_tests/*.*"
13       - id: trailing-whitespace
14        exclude: "tests/cis_tests/*.*"
15   - repo: https://github.com/astral-sh/ruff-pre-commit
16     rev: v0.4.4
17     hooks:
18       - id: ruff
19         args: [--fix, --exit-non-zero-on-fix]
20        exclude: "tests/cis_tests/*.*"
21       - id: ruff-format
22        exclude: "tests/cis_tests/*.*"

```



# Directory layout



# How is this used?



# Modernization and Documentation



# Domain scientists



what's the  
opposite of  
misdirection?



direction, that-a-way,  
sealed order, point of compass,  
spec, lowdown, specs



 Thesaurus.plus

[https://github.com/neutrons/python\\_project\\_template](https://github.com/neutrons/python_project_template)