
Demo of PDF Generation

Author A. Name

Aug 10, 2022

CONTENTS

SPHINX-PDF-WITH-MUDKIP

Sandbox repo to create a PDF with Sphinx (not that easy!)

1.1 Repo goals

1. Manage Sphinx project with `mudkip`
 - no `conf.py`, use ``[mudkip.override]1` instead
 - make latex with `mudkit build --preset latex --output-dir docs/_latex`
2. Create a PDF in Github Codespaces
 - create a bash script to install latex and other dependencies using `apt-get`
 - make script to invoke PDF build

Non-goal:

- Automate PDF creation (e.g. in Github Actions)
- Switch to MyST `index.md`
- Configure VS Code for TOML viewing, etc

1.2 Failure points ()

- `:`,
- latex `.mp4`

2to3

A tool that tries to convert Python 2.x code to Python 3.x code by handling most of the incompatibilities which can be detected by parsing the source and traversing the parse tree.

2to3 is available in the standard library as `lib2to3`; a standalone entry point is provided as `Tools/scripts/2to3`. See [2to3 — Automated Python 2 to 3 code translation](#).

abstract base class

Abstract base classes complement [duck-typing](#) by providing a way to define interfaces when other techniques like `hasattr()` would be clumsy or subtly wrong (for example with [magic methods](#)). ABCs introduce virtual subclasses, which are classes that don't inherit from a class but are still recognized by `isinstance()` and `issubclass()`; see the [abc](#) module documentation. Python comes with many built-in ABCs for data structures (in the `collections.abc` module), numbers (in the `numbers` module), streams (in the `io` module), import finders and loaders (in the `importlib.abc` module). You can create your own ABCs with the [abc](#) module.