

# More Python Programming Leftovers

By *Roy Schestowitz*

Created 22/01/2020 - 7:14pm

Submitted by Roy Schestowitz on Wednesday 22nd of January 2020 07:14:48 PM Filed under [Development](#) [1]

- [Hello Word in Django: How to start with Django](#) [2]

In this article, we will learn how to develop and run a python-Django app in less than 5 minutes.

- [Python GUI Programming With Tkinter](#) [3]

Python has a lot of GUI frameworks, but Tkinter is the only framework that's built into the Python standard library. Tkinter has several strengths. It's cross-platform, so the same code works on Windows, macOS, and Linux. Visual elements are rendered using native operating system elements, so applications built with Tkinter look like they belong on the platform where they're run.

Although Tkinter is considered the de-facto Python GUI framework, it's not without criticism. One notable criticism is that GUIs built with Tkinter look outdated. If you want a shiny, modern interface, then Tkinter may not be what you're looking for.

However, Tkinter is lightweight and relatively painless to use compared to other frameworks. This makes it a compelling choice for building GUI applications in Python, especially for applications where a modern sheen is unnecessary, and the top priority is to build something that's functional and cross-platform quickly.

- [The contextmanager Decorator](#) [4]

Context managers provide a cool programming pattern, especially if you're forgetful or just

have too much to keep track of and you want to simplify your life.



### [URLs Lead The Way](#) [5]

In the last article in the Understand Django series, we saw how a user's browser request goes from their browser to Django's "front door." Now it's time to look at how Django processes those requests.

An HTTP request coming from a browser includes a URL describing which resource Django should produce. Since URLs can come in many forms, we must instruct Django on the kinds of URLs that our web application can handle. This is what the URL configuration is for. In the Django documentation, the URL configuration is called a URLconf, for short.

Where is the URLconf? The URLconf is at the module path set by the `ROOT_URLCONF` setting in your project's settings file. If you ran the `startproject` command, then that setting will be named like `project.urls` where "project" is the name given as an argument to the command. In other words, the URLconf is placed right next to the settings.py file in `project/urls.py`.

## [Development](#)

---

**Source URL:** <http://www.tuxmachines.org/node/133210>

### **Links:**

[1] <http://www.tuxmachines.org/taxonomy/term/145>

[2] <https://www.pythoncircle.com/post/26/hello-word-in-django-how-to-start-with-django/>

[3] <https://realpython.com/python-gui-tkinter/>

[4] <https://reachtim.com/articles/the-contextmanager-decorator.html>

[5] <https://www.mattlayman.com/understand-django/urls-lead-way/>