

How to use Django, PostgreSQL, and Docker

Jun 27, 2018

In this tutorial we will create a new Django project using Docker and PostgreSQL. Django ships with built-in SQLite support but even for local development you are better off using a “real” database like PostgreSQL that matches what is in production.

It’s *possible* to run PostgreSQL locally using a tool like [Postgres.app](#) however the preferred choice among many developers today is to use [Docker](#), a tool for creating isolated operating systems. The easiest way to think of it is as a large virtual environment that contains everything needed for our Django project: dependencies, database, caching services, and any other tools needed.

A big reason to use Docker is that it completely removes any issues around local development set up. Instead of worrying about which software packages are installed or running a local database alongside a project, you simply run a Docker image of the entire project. Best of all, this can be shared in groups and makes team development much simpler.

Install Docker

The first step is to install the desktop Docker app for your local machine:

- [Docker for Mac](#)
- [Docker for Windows](#)
- [Docker for Linux](#)

The initial download of Docker might take some time to download. It’s a big file. Feel free to stretch your leas at this point!