

We use cookies to make interactions with our websites and services easy and meaningful, to better understand how they are used and to tailor advertising. You can read more (https://www.salesforce.com/company/privacy/full_privacy.jsp#nav_info) and make your cookie choices here (https://www.salesforce.com/company/privacy/full_privacy.jsp#nav_info). By continuing to use this site you are giving us your consent to do this.

✕

Language Support (/categories/language-support) > Python (/categories/python-support)

Deploying Python and Django Apps on Heroku

🕒 Last updated 11 December 2018

☰ Table of Contents

- Expected files for Python
- Python deployment flow
- Python versions and upgrades



For a step-by-step introduction to deploying Python apps on Heroku, see [Getting Started on Heroku with Python](https://devcenter.heroku.com/articles/getting-started-with-python) (<https://devcenter.heroku.com/articles/getting-started-with-python>).

Expected files for Python

Heroku automatically identifies your app as a Python app if *any* of the following files are present in its root directory:

- `requirements.txt`
- `setup.py`
- `Pipfile`

If none of these files is present in your app's root directory, the Python buildpack (<https://github.com/heroku/heroku-buildpack-python>) will fail to identify your application correctly.

Python deployment flow

When you deploy to Heroku (<https://devcenter.heroku.com/articles/git>), the dependencies you specify in your `requirements.txt` file are automatically installed before app startup.

If you're using Django, the `collectstatic` (<https://docs.djangoproject.com/en/2.1/howto/static-files/#deployment>) command also runs automatically during the deployment process. This command can be tricky to configure properly. To make it easier, add the Django-Heroku Python package

(<https://github.com/heroku/django-heroku>), which sets up everything for you.

To automatically perform other tasks (such as any required database migrations) before your app is deployed, you can add a release phase (<https://devcenter.heroku.com/articles/release-phase>) command to your app.

Python versions and upgrades

By default, your app continues to use whatever version of Python was used when it was first deployed. You can specify a different Python runtime (<https://devcenter.heroku.com/articles/python-runtimes>) to use in your app's `runtime.txt` file.

You will be notified in your app's build output if the app's Python version is no longer up to date.