

# Theory: Introduction to Django

🕒 7 minutes    0 / 3 problems solved

Skip this topic

Start practicing

2409 users solved this topic. Latest completion was about 1 hour ago.

Django is one of the most-used Python frameworks in web programming. When you are browsing through boards on Pinterest, reviewing code on Bitbucket, or making comments with the help of Disqus, you are using Django products. You can find out about more well-known services that use Django on this [page](#).

The name of the framework is inspired by the stage name of a famous jazz guitarist Django Reinhardt, so the developers who created many handy add-ons for the framework call their group Jazzband. At first, you won't need all of the tools, but whenever you need more, you can find them at this [Github](#) account.

Django framework provides an API for templating HTML pages, making connections to databases, and using HTTP backend services. Django has many useful web development shortcuts and utilities in one place. To start working with the framework, choose the version you'll use. For Django in its A.B.C version, A.B stands for a feature release and C for a patch release. When choosing an appropriate version, pay attention to its feature release number. When you're done choosing, download the latest patch release for this version. To get the most of Django, start with the latest version.

The first stable release of Django was issued in 2008. The first Long-Term Support (LTS) release with version 1.4 came four years later, in 2012.

## §1. LTS

LTS is a well-known standard in software development. It means that developers will support this version of the framework for an extended period of time (for Django, it's usually 3 years or more). You can safely update your version to a newer patch release without fear of breaking compatibility with the source code. For this period, all bugs and security losses will be fixed as soon as possible. Conversely, non-LTS versions are supported only until a newer feature release comes out (note that Django developers support the last two feature releases at a time).

For example, the LTS version 1.11 is supported until April 2020, but the later, non-LTS version 2.0 is already not supported by the developing team.

In this course, we will use the latest LTS release of Django as of April 2019, [version 2.2](#).

Surely you can't wait to finally install Django and get to work. There are two ways to install the package: you can install it globally, which is simpler, or get it in a Python virtualenv.

## §2. Global Installation

To install Django, you need a Python package manager `pip`. Django is no different from any other Python package, so if you want to install it on your system, run:

```
1 | pip install Django==2.2
```

To use the framework, you will probably need a database. For the sake of simplicity, you can [install SQLite](#) unless it's already present on your system.

## §3. Installation in Virtual Environment

To isolate your development environment from other Python packages on your computer, you need to create and activate Python `virtualenv`. Then install the framework to keep it separate:

Current topic:

[Introduction to Django](#)

Stage 1

Topic depends on:

✗ [Frameworks](#)

Stage 1

✗ [Pip](#)

Stage 1

Topic is required for:

[Django MVC](#)

Stage 1

Table of contents:

[1 Introduction to Django](#)

[§1. LTS](#)

[§2. Global Installation](#)

[§3. Installation in Virtual Environment](#)

[§4. Check Installation](#)

[Feedback & Comments](#)

```
1 # If you don't have virtualenv yet
2 pip install virtualenv
3
4 # Unix
5 python -m venv hyperskill-django
6 source hyperskill-django/bin/activate
7 pip install Django==2.2
8
9 # Windows
10
11 python -m venv hyperskill-django
12
13 hyperskill-django\Scripts\activate
14
15 pip install Django==2.2
16
17
18
19
20 # If you want to deactivate a virtual environment
21
22 deactivate
```

A virtual environment is also helpful if you want to keep different versions of a package on your system. You can install each one of them in separate `virtualenv`. Thus if you already have the older version of Django, you can try out the new one with `virtualenv`.

## §4. Check Installation

After you've installed Django, you'll get `django-admin` command in your shell (if you've installed Django in a virtual environment, do not forget to activate it). Django-admin is a helper that can create a template layout for a new project or add an application to an existing project. You can do this manually, but it's much easier to use `django-admin` for this purpose.

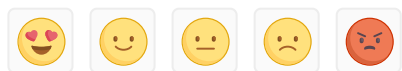
Now you only need to check the version of the installed package:

```
1 django-admin version
2 # 2.2
```

It means that your installation was successful and you can start using Django! You've got the basic essentials to create your first project, so good luck!

 Report a typo

**184** users liked this theory. **3** didn't like it. What about you?



Start practicing

[Comments \(12\)](#)

[Hints \(4\)](#)

[Useful links \(2\)](#)

[Show discussion](#)