

1. Django is popular with web developers because it is built with Python, which is one of the most popular and growing languages in the world. It also comes out of the box with most of the libraries and tools needed for common use cases. Django is designed to ensure the DRY (Don't Repeat Yourself) principle.
2. Instagram uses Django due to its simplicity and practicality. It allows scaling and efficiency. National Geographic eases the development of data-driven, complex websites using Django as Django has its own content management system (CMS). Mozilla uses Django to tackle large amounts of traffic and API hits in a more efficient manner. Eventbrite uses Django for benefits such as URL routing, form building, unit tests, and more. Spotify uses it for its back-end services and data analysis.
3. *To develop a web application with many users:* I will use Django due to its scalability. Django's MVT architecture is based on a loosely coupled architecture which makes it easy to add new hardware and components.
To make fast deployments and changes as I proceed: I will use Django as its MVT architecture ensures development is fast and easy.
To build a very basic application, which doesn't require any database access or file operations: I will not use Django due to the huge amount of out-of-the-box solutions which might be too heavy for small applications.
To build an application from scratch and want a lot of control over how it works: It has a highly structured way of doing things and that might be too rigid for when building from scratch as there is no room to diverge from the rules.
About to start working on a big project and are afraid of getting stuck and needing additional support: I will use Django as it has a large community of contributors which enables getting support easily. It also has very good documentation.

4.

```
(web-dev) C:\Users\user>django-admin --version
4.2.7

(web-dev) C:\Users\user>_
```

5.

```
C:\Users\user>mkvirtualenv achievement2-practice
created virtual environment CPython3.10.9.final.0-64 in 11749ms
creator CPython3Windows(dest=C:\Users\user\Envs\achievement2-practice, clear=False, no_vcs_ignore=False, global=False)

seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy, app_data_dir=C:\Users\user\AppData\Local\pypa\virtualenv)
added seed packages: pip==23.3.1, setuptools==68.2.2, wheel==0.41.2
activators BashActivator,BatchActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator

(achievement2-practice) C:\Users\user>workon

Pass a name to activate one of the following virtualenvs:
=====
achievement2-practice
base-new
cf-python-base
cf-python-copy
web-dev
wed-dev

(achievement2-practice) C:\Users\user>workon achievement2-practice
(achievement2-practice) C:\Users\user>_
```

6.

```
(achievement2-practice) C:\Users\user>workon achievement2-practice
(achievement2-practice) C:\Users\user>pip install django
Collecting django
  Using cached Django-4.2.7-py3-none-any.whl.metadata (4.1 kB)
Collecting asgiref<4,>=3.6.0 (from django)
  Using cached asgiref-3.7.2-py3-none-any.whl.metadata (9.2 kB)
Collecting sqlparse>=0.3.1 (from django)
  Using cached sqlparse-0.4.4-py3-none-any.whl (41 kB)
Collecting tzdata (from django)
  Using cached tzdata-2023.3-py2.py3-none-any.whl (341 kB)
Collecting typing-extensions>=4 (from asgiref<4,>=3.6.0->django)
  Using cached typing_extensions-4.8.0-py3-none-any.whl.metadata (3.0 kB)
Using cached Django-4.2.7-py3-none-any.whl (8.0 MB)
Using cached asgiref-3.7.2-py3-none-any.whl (24 kB)
Using cached typing_extensions-4.8.0-py3-none-any.whl (31 kB)
Installing collected packages: tzdata, typing-extensions, sqlparse, asgiref, django
Successfully installed asgiref-3.7.2 django-4.2.7 sqlparse-0.4.4 typing-extensions-4.8.0 tzdata-2023.3

(achievement2-practice) C:\Users\user>django-admin --version
4.2.7

(achievement2-practice) C:\Users\user>
```