

2.1 Getting Started with Django

2. Django is so popular among web developers because Django's MVT architecture ensures development is fast and easy. Django is implemented in Python which is easy to read and has a lot of powerful computational features. Django is based on loosely coupled architecture, which makes it very simple to add new hardware and components, so scaling a Django project isn't a problem. Django is built with security in mind, allowing secure-by-design implementation with built-in automated encryption.

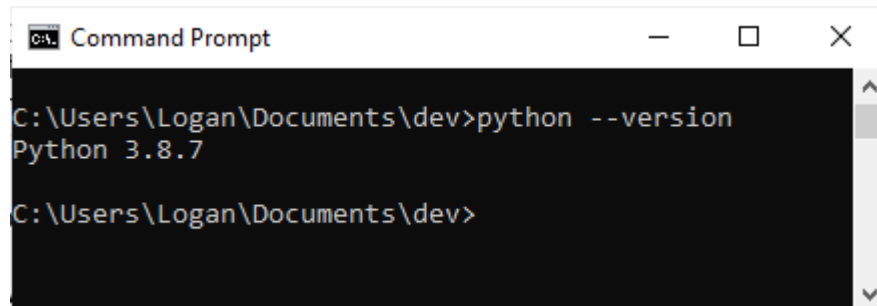
3. List Five large companies that use Django:

1. Pinterest - a social media platform for sharing images in order to give inspiration to their users for topics of interest. They use Django for their website and mobile app.
2. Instagram - a social media application for sharing photos and videos. They use Django for their website and mobile app.
3. National Geographic - a television network and popular magazine series focused on delivering educational content in subject areas like, science, culture, and history. They use Django for their education website.
4. Spotify - an audio streaming provider. Users listen to digital music and podcasts. Django is used for their back-end services and data analysis to increase its functionality.
5. Mozilla - is a global nonprofit which created the popular Firefox browser. They use Django to tackle large amounts of traffic and API hits more efficiently.

4. For each of the following scenarios, explain if you would use Django (and why or why not):

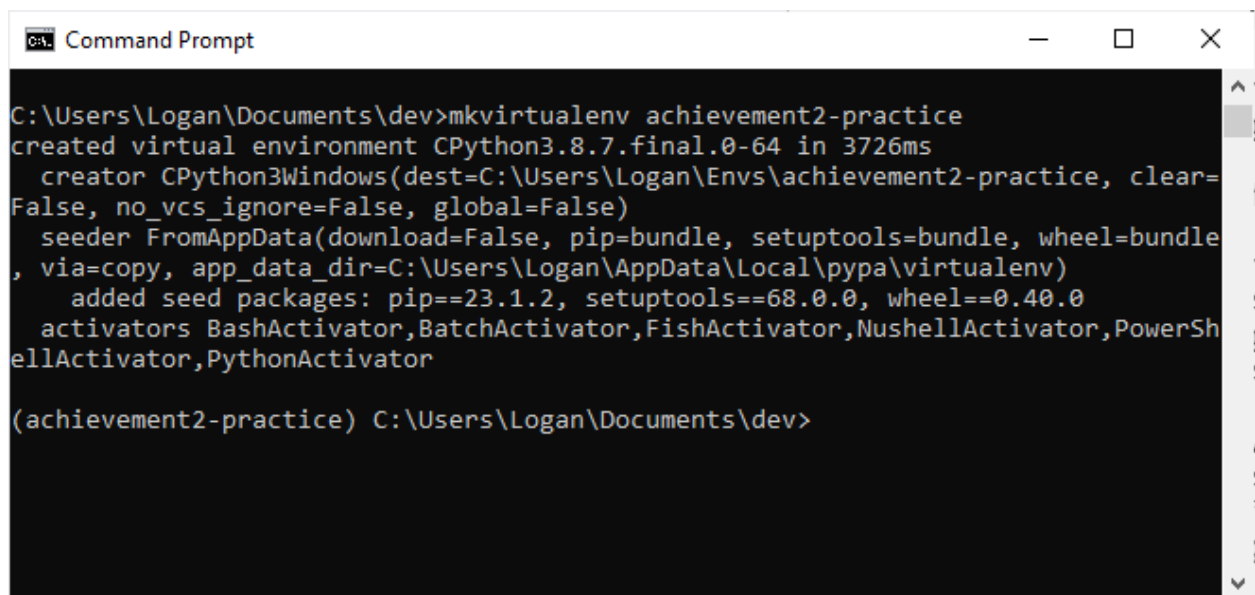
- You need to develop a web application with multiple users. Yes this seems like a perfect project to use Django because to store your users information you will need a database and Django comes ready with the features to interact with one.
- You need fast deployment and the ability to make changes as you proceed. Yes, Django would be a good fit because Django's MVT architecture ensures development is fast and easy and it's based on loosely coupled architecture, which makes it very simple to add new hardware and components.
- You need to build a very basic application, which doesn't require any database access or file operations. No, this is not a good time to use Django because Django comes with a lot of overhead and support for features such as database access so your application will be unnecessarily slow and bloated.
- You want to build an application from scratch and want a lot of control over how it works. No, this is not a good time to use Django because Django comes with a lot of pre-written code and you have to follow a very specific design pattern when using this framework.
- You're about to start working on a big project and are afraid of getting stuck and needing additional support. Yes, this is a good time to use Django because Django is good for large projects and is very popular so you will be able to find a lot of support online.

5.



```
Command Prompt
C:\Users\Logan\Documents\dev>python --version
Python 3.8.7
C:\Users\Logan\Documents\dev>
```

7.



```
Command Prompt
C:\Users\Logan\Documents\dev>mkvirtualenv achievement2-practice
created virtual environment CPython3.8.7.final.0-64 in 3726ms
  creator CPython3Windows(dest=C:\Users\Logan\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\Logan\AppData\Local\pypa\virtualenv)
  added seed packages: pip==23.1.2, setuptools==68.0.0, wheel==0.40.0
  activators BashActivator,BatchActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator

(achievement2-practice) C:\Users\Logan\Documents\dev>
```

8.

```
Command Prompt
(achievement2-practice) C:\Users\Logan\Documents\dev>py -m pip install Django
Collecting Django
  Using cached Django-4.2.3-py3-none-any.whl (8.0 MB)
Collecting asgiref<4,>=3.6.0 (from Django)
  Using cached asgiref-3.7.2-py3-none-any.whl (24 kB)
Collecting sqlparse>=0.3.1 (from Django)
  Using cached sqlparse-0.4.4-py3-none-any.whl (41 kB)
Collecting backports.zoneinfo (from Django)
  Using cached backports.zoneinfo-0.2.1-cp38-cp38-win_amd64.whl (38 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-4.7.1-py3-none-any.whl (33 kB)
Installing collected packages: tzdata, typing-extensions, sqlparse, backports.zoneinfo, asgiref, Django
Successfully installed Django-4.2.3 asgiref-3.7.2 backports.zoneinfo-0.2.1 sqlparse-0.4.4 typing-extensions-4.7.1 tzdata-2023.3

[notice] A new release of pip is available: 23.1.2 -> 23.2
[notice] To update, run: python.exe -m pip install --upgrade pip

(achievement2-practice) C:\Users\Logan\Documents\dev>django-admin --version
4.2.3

(achievement2-practice) C:\Users\Logan\Documents\dev>
```