

Jordan Nelson

## Exercise 2.1

---

Write two to three sentences on why Django is so popular among web developers.

Django is popular among web developers due to its simplicity, scalability, and robustness. It follows the "batteries-included" philosophy, offering a wide range of functionalities out-of-the-box, such as an admin panel, user authentication, and database schema migrations. Additionally, its emphasis on reusable components and a clean, pragmatic design makes it easier for developers to build and maintain high-quality web applications efficiently.

After some research, list five large companies that use Django.

1. **Pinterest:** An image sharing and social media service, uses Django to handle vast amounts of content and user interactions.
2. **Instagram:** A social networking service for sharing photos and videos, utilizes Django for its user-friendly interface and ability to scale efficiently.
3. **Mozilla:** Known for the Firefox browser, uses Django for its website and several internal tools.
4. **Spotify:** A music streaming service employs Django to manage complex backend operations and user data processing.
5. **YouTube:** A video sharing platform, leverages Django for certain features and server-side operations.

For each of the following scenarios, explain if you would use Django:

- **You need to develop a web application with multiple users:**

Yes, use Django. It has built-in support for handling user authentication, permissions, and roles, making it ideal for multi-user applications.

- **You need fast deployment and the ability to make changes as you proceed:**

Yes, Django is suitable. It's designed for rapid development and favors convention over configuration, which speeds up development.

- **You need to build a very basic application, which doesn't require any database access or file operations:**

Not ideal for Django. Django is a bit overkill for such simple applications; a lightweight framework or even static site generators might be more appropriate.

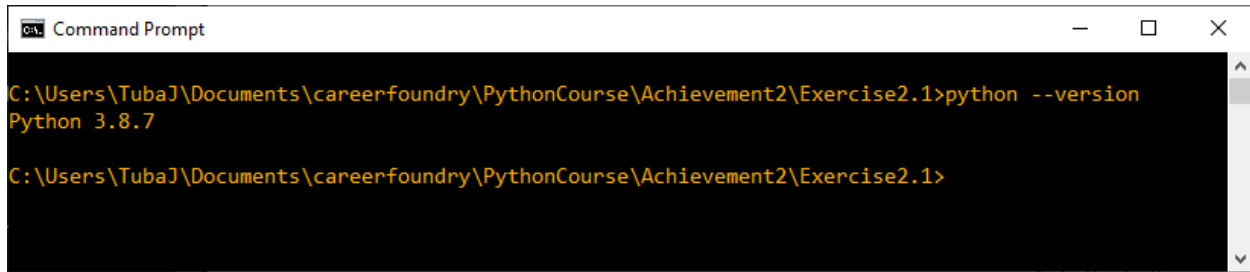
- **You want to build an application from scratch and want a lot of control over how it works:**

Yes, but with considerations. Django is very flexible and customizable, but it does have its own way of doing things. For absolute control, a micro-framework might be better.

- **You're about to start working on a big project and are afraid of getting stuck and needing additional support:**

Yes, Django is ideal. It has a strong community, extensive documentation, and many resources available for support.

## Download and Install Python:

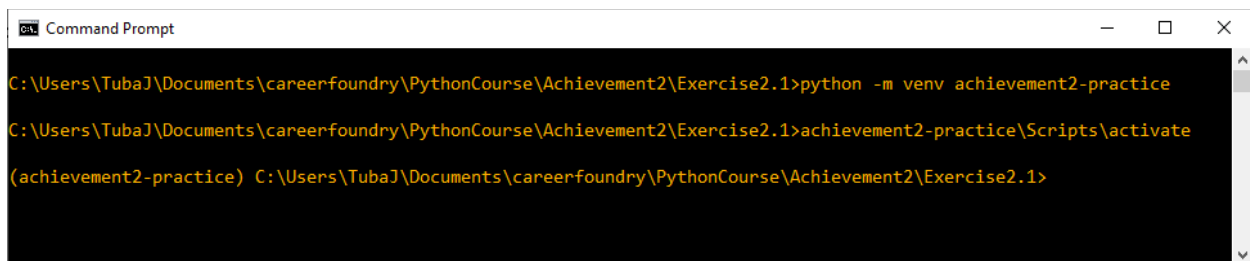


```
Command Prompt

C:\Users\TubaJ\Documents\careerfoundry\PythonCourse\Achievement2\Exercise2.1>python --version
Python 3.8.7

C:\Users\TubaJ\Documents\careerfoundry\PythonCourse\Achievement2\Exercise2.1>
```

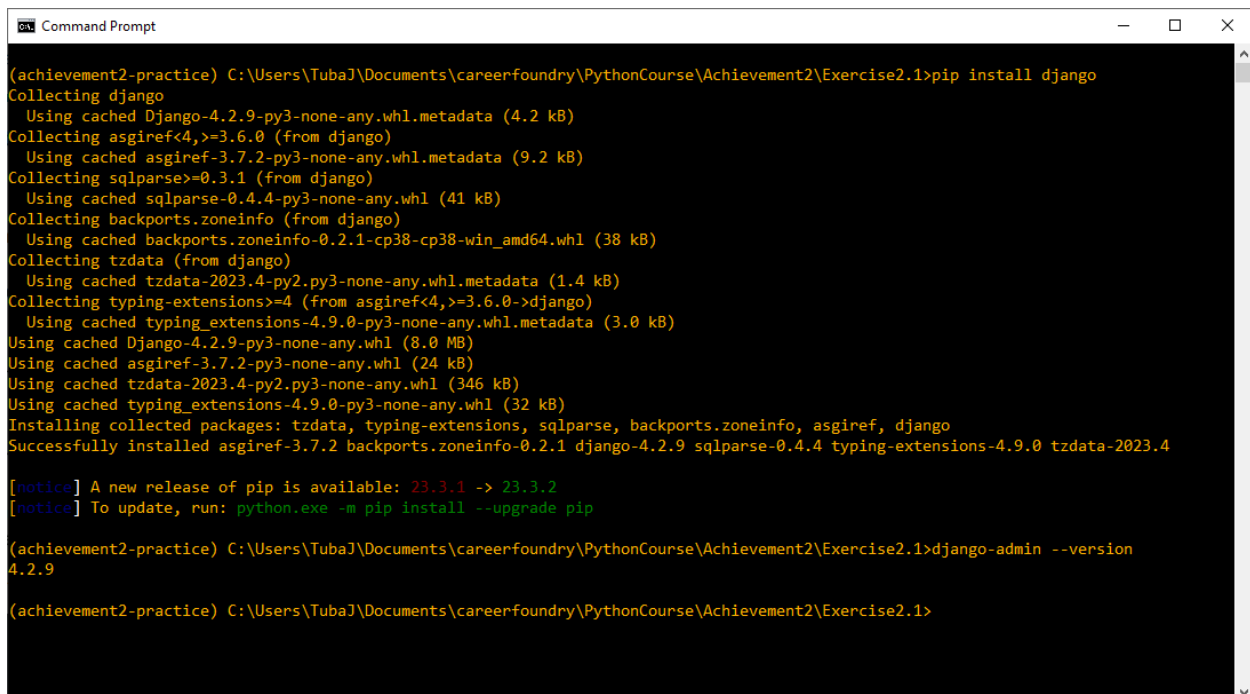
## Create new virtual environment and activate it:



```
Command Prompt

C:\Users\TubaJ\Documents\careerfoundry\PythonCourse\Achievement2\Exercise2.1>python -m venv achievement2-practice
C:\Users\TubaJ\Documents\careerfoundry\PythonCourse\Achievement2\Exercise2.1>achievement2-practice\Scripts\activate
(achievement2-practice) C:\Users\TubaJ\Documents\careerfoundry\PythonCourse\Achievement2\Exercise2.1>
```

## Install Django:



```
Command Prompt

(achievement2-practice) C:\Users\TubaJ\Documents\careerfoundry\PythonCourse\Achievement2\Exercise2.1>pip install django
Collecting django
  Using cached Django-4.2.9-py3-none-any.whl.metadata (4.2 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 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-2023.4-py2.py3-none-any.whl.metadata (1.4 kB)
Collecting typing_extensions>=4 (from asgiref<4,>=3.6.0->django)
  Using cached typing_extensions-4.9.0-py3-none-any.whl.metadata (3.0 kB)
Using cached Django-4.2.9-py3-none-any.whl (8.0 MB)
Using cached asgiref-3.7.2-py3-none-any.whl (24 kB)
Using cached tzdata-2023.4-py2.py3-none-any.whl (346 kB)
Using cached typing_extensions-4.9.0-py3-none-any.whl (32 kB)
Installing collected packages: tzdata, typing_extensions, sqlparse, backports.zoneinfo, asgiref, django
Successfully installed asgiref-3.7.2 backports.zoneinfo-0.2.1 django-4.2.9 sqlparse-0.4.4 typing_extensions-4.9.0 tzdata-2023.4

[notice] A new release of pip is available: 23.3.1 -> 23.3.2
[notice] To update, run: python.exe -m pip install --upgrade pip

(achievement2-practice) C:\Users\TubaJ\Documents\careerfoundry\PythonCourse\Achievement2\Exercise2.1>django-admin --version
4.2.9

(achievement2-practice) C:\Users\TubaJ\Documents\careerfoundry\PythonCourse\Achievement2\Exercise2.1>
```