

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

Ans: Django is popular because it enables fast web development with its easy-to-use Python syntax. It includes many built-in features, reducing the need for extra tools, and emphasizes security, making it a favorite for creating efficient and secure web applications

2. After some research, list five large companies that use Django. Specify what the company's product or service is and what they use Django for.

Instagram - A social media application for sharing photos and videos. Django help Instagram to build tools that scale well.

Eventbrite - A website that provides event management and ticketing service. Initially, Eventbrite was not relying on Django to support his system. However, after rapid audience growth, the platform started facing scalability issues. The developer's team therefore decided to move on to Django to help to scale their website and better handle high traffic.

YouTube - A popular video-sharing platform relies on Django to swiftly implement new features and upgrades, enhancing its development process efficiency.

The Washington Post - A newspaper company that offers products both physically and digitally. The company uses Django mainly to handle the huge traffic their site receive every day with a very well, fast and efficient performance. Django is very useful to scale their website and handle large amount of data generated by their audience.

National Geographic - A television network and popular magazine series focused on delivering educational content in subject areas like science, culture, and history. Django helps with the development of data-driven / complex website.

3. 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 - Django is well-suited for developing web applications with multiple users due to its robust authentication and authorization features, which help in managing user access and roles effectively.

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

Yes - Django's MVT architecture ensures that development can be fast and easy

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

No - If you don't require database access or file management, using Django might be unnecessary, as it is specifically designed to optimize development tasks related to these functionalities. If your project doesn't involve databases or file operations, exploring alternative frameworks could be more efficient for your needs.

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

No - Django is known for its rigid structure, demanding adherence to a specific methodology. If you seek extensive control and flexibility in how you build your applications, Django may not align with your preferences.

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

Yes - Django is exceptionally well-suited for large projects, and its robust capabilities are complemented by a thriving community. The abundance of available resources and support makes it a reliable choice for handling extensive and complex web development endeavors.

#### 4. Screenshot of the terminal window for python version

```
(web-dev) C:\Users\kunzs\Envs\web-dev\Scripts>py --version
Python 3.8.7

(web-dev) C:\Users\kunzs\Envs\web-dev\Scripts>
```

#### 5. Screenshot of the activated environment

```
Microsoft Windows [Version 10.0.22621.2715]
(c) Microsoft Corporation. All rights reserved.

C:\Users\kunzs>mkvirtualenv achievement2-practice
created virtual environment CPython3.8.7.final.0-64 in 3554ms
creator CPython3Windows(dest=C:\Users\kunzs\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\kunzs\AppData\Local\pypa\virtualenv)
added seed packages: pip==23.3.1, setuptools==68.2.2, wheel==0.41.3
activators BashActivator,BatchActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator

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

#### 6. Screenshot of the terminal with the Django version

```
(achievement2-practice) C:\Users\kunzs>py -m pip install django
Collecting django
  Using cached Django-4.2.8-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-2023.3-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-4.2.8-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, backports.zoneinfo, asgiref, django
Successfully installed asgiref-3.7.2 backports.zoneinfo-0.2.1 django-4.2.8 sqlparse-0.4.4 typing-extensions-4.8.0 tzdata-2023.3

(achievement2-practice) C:\Users\kunzs>django-admin --version
4.2.8

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