

Write two to three sentences on why Django is so popular among web developers. Django is popular because it's a toolkit that makes building websites quicker and less repetitive. It's like having a helpful assistant that guides you in making websites and saves you time. Plus, a lot of people use it and share their knowledge, so it's like being part of a supportive community.

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. Certainly!

Reddit:

- Use of Django: Reddit utilizes Django for various components of their backend, including handling user authentication, content management, and interactions within the platform.

The Washington Post:

- Use of Django: The Washington Post's website was built using Django, leveraging its features for content management, user authentication, and delivering news to readers.

Eventbrite:

- Use of Django: Django is used to power Eventbrite's backend, supporting functionalities related to event creation, ticketing, payment processing, and user management.

Mozilla:

- Use of Django: Django is used in various projects within Mozilla for web development, allowing for efficient and secure handling of features related to their products and services.

NASA - Data Science Division:

- Use of Django: Django is used in various data-driven applications within NASA's Data Science Division, providing a robust framework for processing and presenting scientific data.

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

- a. **You need to develop a web application with multiple users.** Django is great for handling multiple users and building robust web applications with user authentication, permissions, and other essential features.
- b. **You need fast deployment and the ability to make changes as you proceed.** Django's robust set of built-in features, extensive documentation, and the availability of numerous third-party packages make it suitable for rapid development and easy adaptability to changes.
- c. **You need to build a very basic application, which doesn't require any database access or file operations.** Consider something else: Django might be overkill for a basic application with no database or file operations. Lightweight frameworks like Flask or even plain HTML/CSS/JavaScript may be more appropriate and simpler for this purpose.
- d. **You want to build an application from scratch and want a lot of control over how it works.** Consider something else: If you desire extensive control over every aspect of your application, a more lightweight framework like Flask or a microframework might be a better fit. Django, while powerful, can come with some overhead and abstractions that limit low-level control.
- e. **You're about to start working on a big project and are afraid of getting stuck and needing additional support.** Django has a strong community and a plethora of resources, making it a good choice when you

anticipate needing support during a big project. The community can provide help, and there are many tutorials, forums, and documentation available.

Download and install Python (if you haven't done so already).

- f. Run the appropriate command to check the Python version.
- g. Take a screenshot of the terminal window with the command and version and paste it into your answers document.

```
C:\Users\caleb>python --version
Python 3.8.7
```

If you haven't done so already, set up and create a virtual environment and name it `achievement2-practice`. Then:

- h. Activate the virtual environment.
- i. Take a screenshot of the activated environment and paste it into your answers document.

```
C:\Users\caleb>C:\Users\caleb\achievement2-practice\Scripts\activate
(achievement2-practice) C:\Users\caleb>|
```

Install Django and verify the installation by checking the version. Then:

j. Take a screenshot of the terminal with the command and version.

k. Paste it into your answers document.

```
PS C:\Users\caleb\OneDrive\Documents\CF Projects\cf-python-acheivement-2> C:\Users\caleb\achievement2-pra
>>
PS C:\Users\caleb\OneDrive\Documents\CF Projects\cf-python-acheivement-2> py -m pip install Django
Collecting Django
  Obtaining dependency information for Django from https://files.pythonhosted.org/packages/b9/45/707dfc56
  Downloading Django-4.2.6-py3-none-any.whl.metadata (4.1 kB)
Collecting asgiref<4,>=3.6.0 (from Django)
  Obtaining dependency information for asgiref<4,>=3.6.0 from https://files.pythonhosted.org/packages/9b/
  7.2-py3-none-any.whl.metadata
  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)
Requirement already satisfied: typing-extensions>=4 in c:\users\caleb\appdata\local\programs\python\pytho
Downloading Django-4.2.6-py3-none-any.whl (8.0 MB)
  8.0/8.0 MB 7.9 MB/s eta 0:00:00
Using cached asgiref-3.7.2-py3-none-any.whl (24 kB)
Installing collected packages: tzdata, sqlparse, backports.zoneinfo, asgiref, Django
Successfully installed Django-4.2.6 asgiref-3.7.2 backports.zoneinfo-0.2.1 sqlparse-0.4.4 tzdata-2023.3
PS C:\Users\caleb\OneDrive\Documents\CF Projects\cf-python-acheivement-2> django-admin --version
4.2.6
```

