

2) Why is Django so popular among web developers?

Django is a high-level Python web framework that enables developers to build web applications rapidly and with clean, pragmatic designs. It offers built-in features like an ORM, authentication, and admin panels, saving time and reducing redundancy. Django is scalable, secure, and easy to extend, making it suitable for projects of all sizes, from small apps to large, complex systems.

3) List five large companies that use Django and specify what they use it for.

Instagram is a social media platform for sharing photos and videos, and they use Django to handle millions of daily users, enabling scalable and fast development.

Spotify is a music streaming service, and they use Django for managing its backend services and APIs.

Pinterest is a social media platform for sharing images. They use Django to handle a large volume of user-generated content and traffic.

Mozilla is the creator of the Firefox browser and they use Django for parts of its website, and especially for the Mozilla Developer Network.

Dropbox is a cloud storage service, and they use Django to power their file handling and sharing service.

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.

Use Django. Django's authentication system is great for handling multiple users, managing permissions, and user sessions securely.

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

Use Django, because Django's robust admin interface and scalability allow for quick deployment and the ability to make changes easily without breaking the system.

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

Do not use Django. Django may be too heavy for a basic application without database needs.

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

Do not use Django. If you require complete control and flexibility, a lightweight framework would be better since Django has more built-in features that may limit customization.

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

Use Django. Django has a large community, extensive documentation, and many resources, which make it easier to find if you get stuck.