Why is Django so popular amongst web developers?

Django provides many features out of the box making it suitable for large scale projects. Having this straight out of the box feature makes using Django desirable and the structure it provides the developers can give them an ease of mind when it comes down to the roadblocks of developing the fundamental ‘rules’ of their projects. The structure of Django creates an efficient development flow. There is a lesser hassle of setting up CDNs and encryption since it does most of the work for you. The framework promotes the use of original code (DRY), which reduces repetitive code for developers.

List of large companies that use Django

- **Spotify:** The music streaming service uses Django to manage their backend services and data processing
- **Pinterest:** Pinterest uses Django for their web application to handle a massive amount of user-generated content and interactions.
- **Instagram:** Instagram started using Django for its web application and continues to leverage its features to handle millions of users.
- **The New York Times:** The New York Times employs Django for some of its web applications and content management systems
- **Coursera:** An online education platform, Coursera, also uses Django to handle course management and user enrollment.

Scenarios for Django

You need to develop a web application with multiple users: I would use Django for this scenario because of its built-in encryption and security features that come along with the rest of its package.

You need fast deployment and the ability to make changes as you proceed.: I would only use Django in this scenario if the project is at a big scale. Django has the ability to expedite development processes but the structure of Django can interfere with the desire to make changes as you go.

You need to build a very basic application, which doesn’t require any database access or file operations: I would not use Django in this scenario. As it is most optimal for a large scale project that requires complex data structure. I would use a different framework like Flask instead since that framework is more small project friendly.

You want to build an application from scratch and want a lot of control over how it works: I would not use Django in this scenario, because of the 'Django way' it restricts you from controlling everything due its strict out of the box package. I would suggest using Pyramid because it provides more flexibility to the development process without compromising a lot of efficiency in your project.

You're about to start working on a big project and are afraid of getting stuck and needing additional support: I would suggest using Django in this scenario because Django has a large community where resources are highly accessible if you need support in the development process. Django is scalable, efficient, and is most suitable for a large scale project.