

Amy Tiefermann Homework 2.1

2. Django is popular among web developers because it is a high-level Python web framework that promotes rapid development and clean, pragmatic design. It comes with built-in features like user authentication, admin panels, and database management, allowing developers to focus more on building their application's unique features rather than reinventing the wheel.

3. Five Companies That Use Django:

- Instagram: A global photo and video sharing social platform. Instagram uses Django to manage its huge user base and high-volume traffic due to its ability to scale and handle complex database operations.
- Spotify: A music streaming service. Spotify uses Django for data analysis and backend services thanks to Django's fast development cycle and Python's powerful data libraries.
- Pinterest: A visual discovery engine for finding ideas like recipes and home decor. Django helps Pinterest scale and manage their user interactions and content sharing.
- Mozilla: The organization behind the Firefox web browser. Mozilla uses Django for many internal tools and websites due to its open-source nature and flexibility.
- Disqus: A comment hosting service for blogs and online communities. Disqus was built entirely with Django and continues to use it for handling millions of users and posts.

4. Scenarios:

- You need to develop a web application with multiple users.
Use Django. Django has built-in user authentication, permissions, and admin tools that make it ideal for multi-user applications.
- You need fast deployment and the ability to make changes as you proceed.
Use Django. Django's rapid development philosophy and built-in tools make it easy to launch quickly and adjust as the project evolves.
- You need to build a very basic application, which doesn't require any database access or file operations.
Don't use Django. For very simple applications, Django might be overkill. A lightweight framework like Flask would be more appropriate.
- You want to build an application from scratch and want a lot of control over how it works.
Don't use Django. While Django is powerful, it follows a strict design pattern. Developers looking for full flexibility might prefer something more minimal like Flask or

FastAPI.

- 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, active community, excellent documentation, and tons of third-party packages — making it a great choice when support and resources are important.