James foday **CF**

Q&A
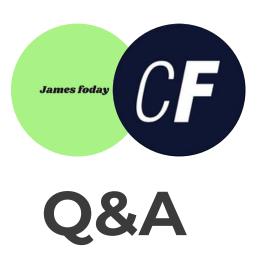
1. Why is Django so popular among web developers?

Django is popular because it is a high-level Python web framework that encourages rapid development and clean, pragmatic design. It comes with many built-in features like an admin panel, authentication, and ORM, which help developers build secure, scalable, and maintainable web applications quickly. Additionally, Django's adherence to the "batteries-included" philosophy makes it easy for developers to focus on writing their application logic instead of reinventing common web functionalities.

2. Five Large Companies Using Django and Their Use Cases

Company	Product/Service	Use of Django
Instagram	Social media platform	Web backend, API services
Pinterest	Visual discovery and bookmarking	Scalable backend services
Disqus	Blog comment hosting service	Web application framework and APIs
Mozilla	Web browser and internet services	Backend services and web apps
Spotify	Music streaming service	Backend data processing and web services

3. Should you use Django in these scenarios?

3. Should you use Django in these scenarios?

Scenario	Use Django?	Why or Why Not?
Develop a web application with multiple users	Yes	Django's built-in user authentication and session management make it ideal for multi-user apps.
Need fast deployment and ability to make changes on the fly	Yes	Django's modular design and "batteries-included" features speed up development and iteration.
Build a basic app with no database or file operations	No	Django is heavyweight for simple static apps without backend requirements; a lightweight framework is better.
Build an app from scratch wanting full control over internals	No	Django enforces a certain structure ("the Django way"), which may limit fine-grained control.
Starting a big project and want lots of community support	Yes	Django has a large, active community and extensive documentation, ideal for complex projects.

New virtual Environment Install

```
jamesfoday@jamess-imac python % python3 --version

Python 3.13.5
jamesfoday@jamess-imac python % python3 -m venv achievement2-practice
```

Activate virtual Envireonment and install Django

```
jamesfoday@jamess-imac python % source achievement2-practice/bin/activate
(achievement2-practice) jamesfoday@jamess-imac python % pip install django

Collecting django
  Using cached django-5.2.4-py3-none-any.whl.metadata (4.1 kB)
Collecting asgiref>=3.8.1 (from django)
  Using cached asgiref-3.9.1-py3-none-any.whl.metadata (9.3 kB)
Collecting sqlparse>=0.3.1 (from django)
  Using cached sqlparse-0.5.3-py3-none-any.whl.metadata (3.9 kB)
Using cached django-5.2.4-py3-none-any.whl (8.3 MB)
Using cached asgiref-3.9.1-py3-none-any.whl (23 kB)
Using cached sqlparse-0.5.3-py3-none-any.whl (44 kB)
Installing collected packages: sqlparse, asgiref, django
Successfully installed asgiref-3.9.1 django-5.2.4 sqlparse-0.5.3

[notice] A new release of pip is available: 25.1.1 -> 25.2
[notice] To update, run: pip install --upgrade pip
(achievement2-practice) jamesfoday@jamess-imac python % django-admin --version

5.2.4
(achievement2-practice) jamesfoday@jamess-imac python % 
```