

Write two to three sentences on why Django is so popular among web developers.

Django is popular because it comes with a lot of built-in features that make web development easier. Instead of having to build everything from scratch, developers can use Django's ready-to-go tools for things like databases, user authentication, and security. This saves a lot of time and lets developers focus more on making the app work the way they want.

Another big reason Django is used so much is that it helps with security. It automatically protects against common security issues, so developers don't have to worry as much about hackers. Django is also great for projects that need to grow over time because it's built to handle a lot of data and users without slowing down.

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.

Many large companies use Django because it helps them handle a lot of data and users efficiently while still allowing them to update their sites quickly.

Here are some examples:

1. **Instagram** – Uses Django to handle millions of users posting, liking, and commenting on photos every second. Django makes it easier to manage all that data.
2. **Spotify** – Uses Django for its backend, helping it organize all the music, playlists, and user accounts quickly and smoothly.
3. **Pinterest** – Uses Django to keep track of the billions of images users upload and share. It helps make sure everything loads fast.
4. **Mozilla (Firefox)** – Uses Django for different parts of its website, including managing updates and user support.
5. **National Geographic** – Uses Django to organize all the articles, photos, and videos on its website so people can easily find and view them.

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

1. You need to develop a web application with multiple users.
Yes, I would use Django for this because it has a built-in way to handle user accounts. This makes it easy for people to sign up, login, and have individualized user based access.
2. You need fast deployment and the ability to make changes as you proceed.

Yes, Django is great for that. It makes it easy to build and update websites quickly. Its built-in tools take care of security and databases so nothing needs to be set up by hand. It also allows for making changes to the data without having to start over.

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

No, Django would not be a good fit for this. It would be too much for a simple app that doesn't need to store data.

Maybe Flask would be a better fit for this project.

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

Probably not the best fit for Django due to the structure required by its framework to build the website. If full control is needed, you would not get that with Django.

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

Yes, Django could work for this project. It has a large user community and plenty of documentation available online for reference.

It is also open source software and you can find tutorials and resources for troubleshooting issues easily.

Image of my successful Django installation:

```
(web-dev) amysikora@Amys-MacBook-Air ~ % django-admin --version
4.2.19
(web-dev) amysikora@Amys-MacBook-Air ~ %
```