1. Django's popularity can be credited to its fast development, fast processing, and scalability. MVT architecture ensures easy and fast development while also handling network transmission and content delivery at a high speed. Finally, again credited to MVT, Django is based on loosely coupled architecture that makes it simple to add new hardware and components for additional scaling.
2. Instagram, Pinterest, Spotify, Dropbox, and YouTube all use Django architecture.
   a. Instagram: Social media. Data storage and user interaction.
   b. Pinterest: Social media. Data storage and user interaction.
   c. Spotify: Music streaming. Fast backend, machine learning options.
   d. Dropbox: Cloud-based storage. Fast development.
   e. YouTube: Video streaming, social media. Fast processing/performance and scalability.
3. Explain if you would use Django (and why/why not):
   a. You need to develop a web application with multiple users.
      i. I would use Django as it provides advantages in performance and scaling.
   b. You need fast deployment and the ability to make changes as you proceed.
      i. I would use Django as its key benefits are fast deployment which include easy development.
   c. You need to build a very basic application, which doesn't require any database access or file operations.
      i. Django isn't necessary here as those are the operations for which Django offers advantages. I would not use Django in this instance.
   d. You want to build an application from scratch and want a lot of control over how it works.
      i. I would not use Django as there is a "Django Way" that comes with any structured framework that may not allow for flexibility in functions.
   e. You're about to start working on a big project and are afraid of getting stuck and needing additional support.
      i. I would use Django as it is open source and, thus, features a vast & active development community with plenty of documentation.


```
macbookpro:~ tylerkunce$ python3 --version
Python 3.8.7
```

```
macbookpro:~ tylerkunce$ mkvirtualenv achievement2-practice
created virtual environment CPython3.8.7.final.0-64 in 521ms
  creator CPython3Posix(dest=/Users/tylerkunce/.virtualenvs/achievement2-practic
e, clear=False, no_vcs_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle
, via=copy, app_data_dir=/Users/tylerkunce/Library/Application Support/virtualen
v)
    added seed packages: pip==24.3.1, setuptools==75.2.0, wheel==0.44.0
  activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerS
hellActivator,PythonActivator
virtualenvwrapper.user_scripts creating /Users/tylerkunce/.virtualenvs/achieveme
nt2-practice/bin/predeactivate
virtualenvwrapper.user_scripts creating /Users/tylerkunce/.virtualenvs/achieveme
nt2-practice/bin/postdeactivate
virtualenvwrapper.user_scripts creating /Users/tylerkunce/.virtualenvs/achieveme
nt2-practice/bin/preactivate
virtualenvwrapper.user_scripts creating /Users/tylerkunce/.virtualenvs/achieveme
nt2-practice/bin/postactivate
virtualenvwrapper.user_scripts creating /Users/tylerkunce/.virtualenvs/achieveme
nt2-practice/bin/get_env_details
(achievement2-practice) macbookpro:~ tylerkunce$
```

🗂 tylerkunce — -bash — 80×24

```
(achievement2-practice) macbookpro:~ tylerkunce$ pip install django
Collecting django
  Using cached Django-4.2.16-py3-none-any.whl.metadata (4.1 kB)
Collecting asgiref<4,>=3.6.0 (from django)
  Using cached asgiref-3.8.1-py3-none-any.whl.metadata (9.3 kB)
Collecting sqlparse>=0.3.1 (from django)
  Using cached sqlparse-0.5.1-py3-none-any.whl.metadata (3.9 kB)
Collecting backports.zoneinfo (from django)
  Using cached backports.zoneinfo-0.2.1-cp38-cp38-macosx_10_14_x86_64.whl.metada
ta (4.7 kB)
Collecting typing-extensions>=4 (from asgiref<4,>=3.6.0->django)
  Using cached typing_extensions-4.12.2-py3-none-any.whl.metadata (3.0 kB)
Using cached Django-4.2.16-py3-none-any.whl (8.0 MB)
Using cached asgiref-3.8.1-py3-none-any.whl (23 kB)
Using cached sqlparse-0.5.1-py3-none-any.whl (44 kB)
Using cached backports.zoneinfo-0.2.1-cp38-cp38-macosx_10_14_x86_64.whl (35 kB)
Using cached typing_extensions-4.12.2-py3-none-any.whl (37 kB)
Installing collected packages: typing-extensions, sqlparse, backports.zoneinfo,
asgiref, django
Successfully installed asgiref-3.8.1 backports.zoneinfo-0.2.1 django-4.2.16 sqlp
arse-0.5.1 typing-extensions-4.12.2
(achievement2-practice) macbookpro:~ tylerkunce$ django-admin --version
4.2.16
(achievement2-practice) macbookpro:~ tylerkunce$
```