

## 2.1

2. Django is very popular amongst web developers because it provides a high-level, Python framework that simplifies creating large, scalable applications. It is extremely useful for applications that require a database.
3. 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.
  1. Instagram: A social media platform for photo and video sharing. Instagram uses Django to handle its massive data and web traffic, especially for managing its backend infrastructure, allowing for rapid scaling
  2. Spotify: Global music streaming service. Spotify uses Django for its web interface and backend services.
  3. Pinterest: another social media platform, for finding ideas related to various subjects. Pinterest uses Django for prototyping
  4. Dropbox: a site for sharing and storing files. Dropbox uses Django to sync accounts across different devices.
  5. Mozilla: the organization behind the browser Firefox. They use Django for several of their applications, for to its robust admin panel.
4. For each of the following scenarios, explain if you would use Django (and why or why not):
  - You need to develop a web application with multiple users.
    - ◆ Yes, since Django has built in authentication and session management.
  - You need fast deployment and the ability to make changes as you proceed.
    - ◆ Yes, Django has a lot of built in tools for rapid iteration and easy deployment.
  - You need to build a very basic application, which doesn't require any database access or file operations.
    - ◆ No, since it's a very basic application with no db access, Django is overkill for this app.
  - You want to build an application from scratch and want a lot of control over how it works.
    - ◆ No, Django comes with too much built in, there are other frameworks that allow for more customization
  - You're about to start working on a big project and are afraid of getting stuck and needing additional support.

- ◆ Yes, Django has a lot of prebuilt tools and documentation, so there would be a lot to go off.

## 5. Python Version:

```

nv_details
• (achievement2-practice) alexsweeney@Alexs-Mac-mini Python-Specialization % python -V
Python 3.8.7

```

## 6 & 7:

```

can't remove not found python
• alexsweeney@Alexs-Mac-mini Python-Specialization % mkvirtualenv achievement2-practice
created virtual environment CPython3.8.7.final.0-64 in 238ms
  creator CPython3Posix(dest=/Users/alexsweeney/.virtualenvs/achievement2-practice, clear=False, no_vcs
_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy, app_data_di
r=/Users/alexsweeney/Library/Application Support/virtualenv)
    added seed packages: pip==24.2, setuptools==74.1.3, wheel==0.44.0
  activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonAct
ivator
virtualenvwrapper.user_scripts creating /Users/alexsweeney/.virtualenvs/achievement2-practice/bin/prede
activate
virtualenvwrapper.user_scripts creating /Users/alexsweeney/.virtualenvs/achievement2-practice/bin/postd
eactivate
virtualenvwrapper.user_scripts creating /Users/alexsweeney/.virtualenvs/achievement2-practice/bin/preac
tivate
virtualenvwrapper.user_scripts creating /Users/alexsweeney/.virtualenvs/achievement2-practice/bin/posta
ctivate
virtualenvwrapper.user_scripts creating /Users/alexsweeney/.virtualenvs/achievement2-practice/bin/get_e
nv_details
• (achievement2-practice) alexsweeney@Alexs-Mac-mini Python-Specialization % python -V
Python 3.8.7

```

## 8.

```

sions 4.12.2
• (achievement2-practice) alexsweeney@Alexs-Mac-mini Python-Specialization % django-admin --version
4.2.16
• (achievement2-practice) alexsweeney@Alexs-Mac-mini Python-Specialization %

```