

## Task 2.1

### Python version

```
(achievement2-practice) richardlindsell@iMac Achievement_2 % python --version
Python 3.13.5
(achievement2-practice) richardlindsell@iMac Achievement_2 %
```

### Achievement2-practice env

```
richardlindsell@iMac Achievement_2 % mkvirtualenv achievement2-practice
created virtual environment CPython3.13.5.final.0-64 in 356ms
creator CPython3macOsBrew(dest=/Users/richardlindsell/.virtualenvs/achievement2-practice, clear=False, no_vcs_ignore=False, global=False)
seeder FromAppData(download=False, pip=bundle, via=copy, app_data_dir=/Users/richardlindsell/Library/Application Support/virtualenv)
added seed packages: pip==25.1.1
activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator
virtualenvwrapper.user_scripts creating /Users/richardlindsell/.virtualenvs/achievement2-practice/bin/predeactivate
virtualenvwrapper.user_scripts creating /Users/richardlindsell/.virtualenvs/achievement2-practice/bin/postdeactivate
virtualenvwrapper.user_scripts creating /Users/richardlindsell/.virtualenvs/achievement2-practice/bin/preactivate
virtualenvwrapper.user_scripts creating /Users/richardlindsell/.virtualenvs/achievement2-practice/bin/postactivate
virtualenvwrapper.user_scripts creating /Users/richardlindsell/.virtualenvs/achievement2-practice/bin/get_env_details
(achievement2-practice) richardlindsell@iMac Achievement_2 %
```

### Django Install and version

```
(web-dev) richardlindsell@iMac Achievement_2 % pip install django
Collecting django
  Downloading django-5.2.5-py3-none-any.whl.metadata (4.1 kB)
Collecting asgiref<=3.8.1 (from django)
  Downloading asgiref-3.9.1-py3-none-any.whl.metadata (9.3 kB)
Collecting sqlparse<=0.3.1 (from django)
  Downloading sqlparse-0.5.3-py3-none-any.whl.metadata (3.9 kB)
Downloading django-5.2.5-py3-none-any.whl (8.3 MB)
8.3/8.3 MB 12.7 MB/s eta 0:00:00
Downloading asgiref-3.9.1-py3-none-any.whl (23 kB)
Downloading sqlparse-0.5.3-py3-none-any.whl (44 kB)
Installing collected packages: sqlparse, asgiref, django
Successfully installed asgiref-3.9.1 django-5.2.5 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
(web-dev) richardlindsell@iMac Achievement_2 % django-admin --version
5.2.5
```

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

In my opinion Django first and foremost benefits from being built on Python with its concise and highly readable syntax, which makes it easier for developers to write and maintain applications. It takes care of much of the hassle of web development meaning developers can focus on writing the app. For example it comes with many tools that solve common problems, like auth cookies and templates, plus it adapts well to high traffic. All of this increases the speed of development.

2. 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.

**INSTAGRAM** - The ubiquitous photo and video sharing app now owned by meta. Their traffic is over 500 million people per day. Django helped scale the application making it possible to process massive amounts of data and user interactions each second. Plus its robust security features suit its needs during rapid growth.

**PINTEREST** - Similar to instagram, users share, search for and save images that match their interests. With nearly 500 million users each month, Django allows Pinterest to perform and scale as per their increasing number of users. Django's ORM (Object-Relational Mapper) simplifies database interactions, making it easier to manage and retrieve millions of pins and user data.

**SPOTIFY** - Spotify is a worldwide music library servicing a vast array of devices, with over 400 million users each month. Spotify does not use Django as its main backend, but uses it extensively for other services and data analysis, mainly because of speed of development, scalability, Integration with Machine Learning and AI and its active community and ecosystem.

**INDEED** - Indeed is an employment website available in 28 languages in more than 60 countries. It contains around 175 million resumes and 10 new job offers are added each second. Along with the scalability, security and powerful ORM allowing developers to interact with different databases, Indeed benefits from Django's code organisation. It's unified structure for applications makes the codebase more maintainable and easier for teams to work on.

**DROPBOX** - Dropbox is a cloud storage service with about 700 million registered users storing documents, videos and graphics. Dropbox utilises Django's built-in file storage APIs that can be configured to various backends including Dropbox's own storage system. The Django framework is highly extensible, allowing developers to create custom storage backends and integrate with other services.

3. 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 I would use Django, because it is designed to handle a large number of users. It can be scaled horizontally by adding more servers to handle increased traffic.

- You need fast deployment and the ability to make changes as you proceed.

Yes, Django is suitable for fast deployment. It comes with a comprehensive set of built in features such as ORM, authentication, admin interface and caching so you reduce the need for external libraries. Its architecture, based on Model-View-Controller (MVC) pattern gives a clear

separation between data, logic, and presentation allowing for easier scaling and independent development and deployment of different components.

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

Django would not be the best choice for this kind of application. It comes with a significant amount of built in functionality and structure that would not be used. It is unnecessarily complex for a simple project.

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

Because Django is a "Batteries Included" system with a lot of built in features, you won't have as much control over your system. If you want control over fine details Django would not be a first choice.

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

Because Django is open source and has a huge community and excellent documentation, it would be a good choice if you need additional support.