Docker

Docker essentially acts like a virtual environment that we can set up easily.

WHAT DO WE NEED

Installation of Docker

A file called Dockerfile

A .dockerignore file (optional)

The Dockerfile

This is the first line of your Dockerfile.

It all starts with a base image. This can be a minimal version of ubuntu, for instance. We build our image on top of this.

We use python 3.10.7
This lets use pip to install more packages.

FROM

```
FROM python:3.10.7
```

RUN

This lets us run commands while building the image.

These are run only while building the image, NOT while running the container.

```
RUN pip install --no-cache mlflow
RUN pip install --no-cache numpy
RUN pip install --no-cache torch torchvision
```

ADD

```
ADD app.py .
ADD fashion_mnist.py .
RUN mkdir -p FashionMNIST
ADD FashionMNIST ./FashionMNIST
```

This command lets us add files and folders from our filesystem.

We use to add code and also our dataset (though that can be downloaded on the fly if we wish).

This is the command that is run when we run our application!
It is NOT run when building the image.

This is placed at the end of our Dockerfile. We can only use one of these statements.

CMD

```
CMD ["python", "./app.py"]
```

Flask

APP ROUTE

```
@app.route('/', methods=['GET', 'POST'])
def home():
```

This lets us use a directory structure on our website.

A link on the website corresponds to a python function.

Here we have the home page.

We can also specify methods to be used by the function
GET means it can render a template, i.e, accept GET requests from the user
POST means it can get information from a user, i.e, accept POST requests.

REQUESTS

Get information from a HTML form (more on that later)

request.files will get a file
request.form will get the value of a field (or a button) in a form.

A POST request is needed, so a form does need to be filled

TEMPLATES

We can return render_template to render a HTML file using jinja2 (a library used by Flask)

So, we can use HTML for the UI, but can change it with relatively simple code (since jinja2 lets you use if statements in HTML that can optionally take arguments from render_template)