Django (Python)

An “all-inclusive” framework with lots of built-in/out-of-the-box features e.g. authentication, authorization, sessions.

Follows a Model-Template-View architecture (similar to MVC from CS2800)

* Model: handles “business logic” e.g. managing access to database
* Template: presentation layer – how data is presented to user. These are HTML files with Django template language embedded in there to dynamically generate content.
* View: processes user requests. They retrieve/update data from the model and generate final output by rendering templates.

A computer code with text

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated

Examples of a Django templates.

Note:

{% tag %} – Django functionality – load external information, control flow etc.

{{ variable }}

A computer screen shot of a computer code

Description automatically generated

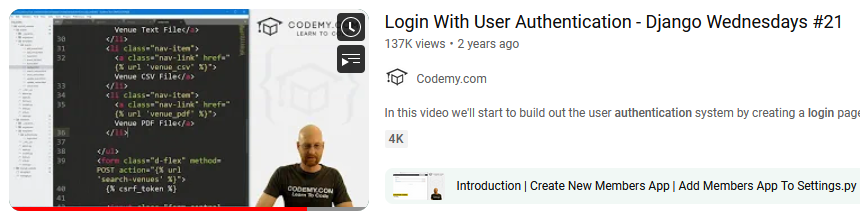
An example of a Django view – retrieves ‘username’ and ‘password’ POST parameters and uses built-in authentication method to authenticate them.

A screenshot of a video account

Description automatically generated

A screenshot of a computer

Description automatically generated



Lots of really comprehensive tutorials and documentation online.

A screenshot of a computer

Description automatically generated

Django has its own admin suite and database!

-? Django commonly used for larger, more complex projects.

-? Steeper learning curve than Flask

