

1. Do some research on Django views. In your own words, use an example to explain how Django views work.

A Django view is a piece of logic that handles requests from a user via the URL and returns an appropriate response (a webpage, an error code, etc). This can be represented as a function or a class method. Each view is typically associated with a specific template. The view acts like a tourguide might if you asked to be taken to a specific place. If the tourguide understood what you were asking, they would take you where you wished to go. If they did not understand, they would not take you anywhere.

2. Imagine you're working on a Django web development project, and you anticipate that you'll have to reuse lots of code in various parts of the project. In this scenario, will you use Django function-based views or class-based views, and why?

In this case it would be better to use class-based views even though they may be a little more difficult to write at first, because then reusing the code will be way easier afterward

3. Read Django's documentation on the [Django template language](#) and make some notes on its basics.

A template is a text file that holds variables that are replaced with values when the template is evaluated. A variable is placed inside `{{double curly braces}}` and is replaced with the specified value when the template evaluates it. There are also `{{variable|filters}}` available which will run the variable through the selected filter before displaying it in the webpage. Filters can be chained as needed and may accept arguments. A default value if there is no value read can also be placed via this filter syntax. `{%Tags%}` are more complex, and can perform logic in regards to the value being read into the template, like ``for loops`` or ``if statements``.