

1. In your own words, use an example to explain how Django views work.
  - Django views can be represented by Python functions or methods of a Python class that accept a request and return a response. To set up the view of an app, you need to define a function to receive the users' request and return the `HttpResponse` in the `views.py` file of the app directory. Once you connect the view to the `urls.py` file of the app and the main project app, you should be able to run your server and visit your URL to view the `HttpResponse`.
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 scenario, I would use Django class-based views because they are the better option when creating views that need to be reused in various parts of the project.
3. Read Django's documentation on the Django template language and make some notes on its basics.
  - A template is a text file that can generate any text-based format like HTML, XML, CSV, etc. Templates contain variables and tags. Variables get replaced with values when the template is evaluated and are encased in double curly brackets like this: `{{variable}}`. Tags control the logic of the template and look like this: `{% tag %}`.