

1. Suppose you're a web developer in a company and need to decide if you'll use vanilla (plain) Python for a project, or a framework like Django instead. What are the advantages and drawbacks of each?
  - In this scenario, whether I would recommend using plain Python or a framework like Django would depend on what the project entailed. If the project were smaller or required more flexibility and control over the development process, then I would recommend using plain Python for the project, even though it requires more effort to build functionalities from scratch. However, if the project were more large scale or needed more streamlined development processes, then I would recommend using a framework like Django instead, even though it makes it so you lose control over the internals of your system.
2. What is the most significant advantage of Model View Template (MVT) architecture over Model View Controller (MVC) architecture?
  - The most significant advantage of Model View Template architecture over Model View Controller architecture is that the Template component handles the controller output and determines how the output should be structured for the browser, while in MVC architecture you have to write the code yourself.
3. Write down 3 goals you have for yourself and your learning process during this Achievement. What do you want to learn about Django? What do you want to get out of this Achievement? Where or what do you see yourself working on after you complete this Achievement?
  - I want to gain an understanding of the MVT architecture of Django.
  - I want to learn how Django differs from JavaScript when creating full-stack applications.
  - I want to learn enough about Django to create my own applications.