1. Think about the various data that websites/applications collect. Write down how analyzing the collected data could help the website/application.
    - Using CareerFoundry as an example, one of the ways that the site collects data is by asking students to provide them with how long it took to complete the reading and task. This information/data is used to determine the average time it should take to complete, which is provided to students with the instructions for the task. This allows students to better plan their time and CareerFoundry to maintain their expectations of students.
2. Read the Django official documentation on QuerySet API. Note down the different ways in which you can evaluate a QuerySet.
    - Iteration: uses a for loop to retrieve object individually
    - Slicing: uses Python's array-slicing syntax to retrieve a list
    - Pickling/caching: essentially reading results from the database
    - repr(): evaluates a QuerySet, allows you to immediately see your results
    - len(): evaluates a QuerySet, returns the length of the result list
    - list(): evaluates a QuerySet, returns a list of objects
    - bool(): tests a QuerySet in a Boolean context, checks if an object exists and returns a True or False
3. Explain the ways in which DataFrame is better than QuerySet for data processing.
    - DataFrame is better than QuerySet for data processing because it has a wider range of operations, extensive built-in functions, and integration w/ data analysis libraries like pandas. It also performs better than QuerySet because DataFrame is optimized for performance on in-memory data processing.