CINS 465: Assignment 5

This is an individual assignment. **Do your own work. Do not copy / paste anything from any outside source** You may brainstorm with or get help on technical problems from others, but all submitted work must be your own.

Objectives

- Migrate your multi-page chess website so that all pages are hosted / generated by the Django framework.

Instructions

- Starting with your solution for Assignment 3 (a multi-page chess site with a board allowing piece movement, a page explaining the history of chess, a page explaining the chess pieces and how they move, and an "About" page explaining how you implemented the site):
    1. Within a conda environment containing Django, create a new Django project **named "assignment5**" and a new Django application within the project **named "app1",** using the techniques described on week 6's Tuesday lecture (my notes are posted under Course Content > Lectures > Week 6 > Lecture 11).
    2. Migrate your existing chess site content into your Django application and project, as we did for my Suduku example site during week 6's Thursday lecture (my notes are posed under Course Content > Lectures > Week 6 > Lecture 12).

Requirements

- You must have one Django project and at least one Django application within that project.
- You must copy each of your existing chess html pages into an application directory, as shown in Lecture 12.
- You must have separate html pages for:
    - The chessboard (make this the home page, navigated via the first item on your nav bar).
    - The history page.
    - The piece description page.
    - An about page, with your name and a brief description of how your site functions (mention the various components you used and how they work together). You should have enough detail here to explain the major components of your web service and how they work. Just a few paragraphs should be adequate.
- You must use Django template tags to generate your chessboard from data passed in from your view function, similar to the way I generated my Sudoku board, as shown in Lecture 12. For now you may just define a static dictionary containing the starting piece locations in separate rows. Here is how I defined the data for my sudoku board:

```
page_data = {
    "rows": [
      {"r1c1": 6, "r1c2": 7, "r1c3": 0, "r1c4": 0, "r1c5": 4, "r1c6": 8, "r1c7": 0, "r1c8": 1, "r1c9": 0},
      {"r2c1": 3, "r2c2": 5, "r2c3": 0, "r2c4": 0, "r2c5": 0, "r2c6": 1, "r2c7": 0, "r2c8": 4, "r2c9": 7},
      {"r3c1": 0, "r3c2": 1, "r3c3": 0, "r3c4": 7, "r3c5": 2, "r3c6": 0, "r3c7": 6, "r3c8": 8, "r3c9": 0},
      {"r4c1": 8, "r4c2": 0, "r4c3": 3, "r4c4": 0, "r4c5": 0, "r4c6": 0, "r4c7": 1, "r4c8": 6, "r4c9": 9},
      {"r5c1": 0, "r5c2": 0, "r5c3": 7, "r5c4": 9, "r5c5": 1, "r5c6": 0, "r5c7": 8, "r5c8": 3, "r5c9": 0},
      {"r6c1": 0, "r6c2": 9, "r6c3": 6, "r6c4": 4, "r6c5": 8, "r6c6": 3, "r6c7": 0, "r6c8": 0, "r6c9": 0},
      {"r7c1": 0, "r7c2": 0, "r7c3": 9, "r7c4": 1, "r7c5": 0, "r7c6": 4, "r7c7": 3, "r7c8": 0, "r7c9": 8},
      {"r8c1": 4, "r8c2": 8, "r8c3": 1, "r8c4": 0, "r8c5": 0, "r8c6": 0, "r8c7": 7, "r8c8": 0, "r8c9": 6},
      {"r9c1": 7, "r9c2": 0, "r9c3": 0, "r9c4": 8, "r9c5": 6, "r9c6": 2, "r9c7": 0, "r9c8": 0, "r9c9": 1}
    ]
  }
```

- That is a dictionary containing a "rows" array, each element of which is a dictionary containing the sudoku cell values within that row.
- You would modify the dictionary above such that each name/value pair is a chessboard location and a piece character, e.g. for the first row: {"a1": "♖", …}. Represent the starting position of the board. Note: The white queen should be on a light square and the black queen should be on a dark square.
- Note: Going forward we will use the standard chess convention of column letter followed by row number to designate each location. The rows are numbered starting from the bottom of the board, so row 1 is the first row of white pieces at the bottom of the board.

- You must share your navigation html on each of your site pages using a Django include template tag, as mentioned in Lecture 12. You may view my provided Sudoku solution for an example (the tgz file posted under the Week 6 lecture folder).
- You must move any custom CSS, javascript, fonts and images into a static content area, which sub-directories for each type of content as shown in Lecture 12.
- You must use the appropriate template tags within your site page html templates to import your static content.
- Some UI enhancements for the chessboard page:
    - Center the board horizontally and size it so fits on the page without having to scroll (on a standard laptop screen).
    - Align your movement form controls with the chessboard and size any input controls appropriately.
    - Make sure there are instructions of some kind on the chessboard page that indicate the expected input for anything the user must type.
    - If you use drop-down controls for the chess movement, separate the “from” and “to” selectors so the user doesn’t have more than 8 choices.
    - Have a separate button control to apply the piece movement rather than automatically applying it on selection or input into the movement input controls.
- **You must submit a single zip or tar file** containing your entire Django project such that when I open it I can navigate into your assignment5 project directory and start your web server using manage.py.