Django Training - 3. Forms, Templates, and Views

Materials

1. Handling HTTP Requests
2. Forms
3. Templates
4. Model Inheritance
5. django-model-utils

Task

1. Instead of having an explicit `created_at` field in the `Album` model, inherit from `TimeStampedModel`
    - tip: using your IDE or djang-model-util GitHub, read through the source code of `TimeStampedModel` to see how it works
    - tip: get comfortable to reading open source code of libraries that you use
2. Create a form that allows a user to create an artist (it should be available at http://localhost:8000/artists/create)
3. Create a form that allows a user to create an album (it should be available at https://localhost:8000/albums/create)
    - (bonus) can you use a user friendly date/time input widget for the release datetime field instead of a plain text input field?
4. For both forms, when the validation fails, the user should see errors displayed in **red** text on top of the form letting the user know what the error is.
    - For example, if I attempt to create an artist with a stage name that already exists (violating the unique constraint), I should see an error telling me so.
5. Create a template view that lists all the albums grouped by each artist (it should be available at https://localhost:8000/artists/)
    - Don't worry about the page's design as long as the data is accurate
    - The page should look something like this:
    - Artist:
        - id: 1
        - Stage name: Drake
        - Social: https://....
        - Albums:
            - id: 1
            - Name: My recent album
            - Creation Time: ...
            - Release datetime: ...
            - Cost: ...
            - ---
            - id: 2
            - Name: Good album
            - Creation Time: ...
            - Release datetime: ...
            - Cost: ...
        - ---
        - id: 2
        - Stage name: Amazing Artist
        - Social: https://....
        - Albums:
            - id: 1
            - Name: My recent album
            - Creation Time: ...
            - Release datetime: ...
            - Cost: ...
            - ---
            - id: 2
            - Name: Good album
            - Creation Time: ...
            - Release datetime: ...
            - Cost: ...
    - Fetch the queryset above in an optimized manner (hint: how can you fetch both albums and artists in one database seek?)

Guidelines

1. Use this gitignore template
2. List your environment variables (if any) in a `.env.example` file
3. Don't forget to run `manage.py makemigrations` and `python manage.py migrate`
4. You are allowed to refer to any articles, documentation source, questions on StackOverFlow
5. You are not allowed to consult other people through any medium of communication
6. Your grade will be affected if your code isn't clean, you should maintain code cleanliness as much as you can