# My Music App - Django Basics Exam Prep I

*More exam preps at* [***SoftUni Judge System***](https://judge.softuni.org/Contests/#!/List/ByCategory/474/Django-Basics-Exams)

*The app allows a user to browse different albums, including the author, genre, and price. The user can create a catalog of albums. Album creators can also edit or delete their posts at any time.*

## Skeleton

All essential **HTML** pages, **images**, and **CSS** resources for the project have been provided to you.

## Structure

Establish a well-organized project structure by initiating **at least** **two** Django **apps** and configuring the **corresponding URLs**.

You have the flexibility to utilize **both** **Function-Based Views** and **Class-Based Views** as needed.

Adhere to best practices and uphold the separation of concerns principle for a robust and maintainable codebase.

## Database

You will need to create **two models**:

* **Profile**
  + **Username**
    - **Character** field, required.
    - It should have at least **2 characters** and **maximum -** **15 characters.**
    - The username can consist **only of letters, numbers, and underscore ("\_")**. Otherwise, raise a **ValidationError** with the message: **"Ensure this value contains only letters, numbers, and underscore."**
  + **Email**
    - **Email** field, required.
  + **Age**
    - **Integer** field, **optional**.
    - The age cannot **be below 0**.
* **Album**
  + **Album Name**
    - **Character** field, required.
    - All album **names** **must** be **unique**.
    - It should consist of a **maximum** **of 30 characters.**
  + **Artist**
    - **Character** field, required.
    - It should consist of a **maximum** **of 30 characters.**
  + **Genre**
    - **Character** field, required.
    - It should consist of a **maximum** **of 30 characters.**
    - The choices are **"Pop Music"**, **"Jazz Music"**, **"R&B Music"**, **"Rock Music"**, **"Country Music"**, **"Dance Music"**, **"Hip Hop Music"**, and **"Other"**.
  + **Description**
    - **Text** field, **optional**.
  + **Image URL**
    - **URL** field, required.
  + **Price**
    - **Float** field, required.
    - The price cannot **be below 0.0**.
  + **Owner**
    - A **foreign key** to the **Profile** model.
    - Establishes a **many-to-one** relationship with the **Profile** model, associating each album with a profile.
    - The **ON DELETE** constraint must be **configured** to an **appropriate value** in alignment with the specified **additional tasks**.
    - This field should remain **hidden** in forms.

**Note:** **Validations will be assessed solely on the user side, with no evaluation from the Admin site.**

## Routes

* <http://localhost:8000/> - Home page
* <http://localhost:8000/album>/add/ - Add album page
* [http://localhost:8000/album/<id>/details/](http://localhost:8000/album/%3cid%3e/details/)- Details album page
* [http://localhost:8000/album/<id>/edit/](http://localhost:8000/album/%3cid%3e/edit/) - Edit album page
* [http://localhost:8000/album/<id>/delete/](http://localhost:8000/album/%3cid%3e/delete/) - Delete album page
* <http://localhost:8000/profile/details/> - Profile details page
* <http://localhost:8000/profile/delete>/ - Delete profile page

## Pages

### Base Template

**Template file**: **"base.html"**

Create a **base template** to be **inherited** by all other templates in the project. This base template should encompass project **metadata** within the <head> tag, along with a consistent **navigation bar** and **footer** for every web page.

**Navigation bar**:

* The **"Home"** link leads to the Home page.
* The **"Create Album"** link leads to the Create Album page.
* The **"Profile"** link leads to the Profile page.

Note that the **"Create Album"** and **"Profile"** links in the navigation bar are **visible** exclusively when the **user** **has** a **profile**.

This template lacks a direct path and is **inaccessible** via **URL**.

### Home Page

**Template files**: **"home-no-profile.html"**; "**home-with-profile.html"**

If **no profile** has been **created**, the home page **displays** a **form** for profile **creation**. This form **includes**:

* A **"Username:"**
  + There must be a **placeholder** "**Username**".
* An **"Email:"**
  + There must be a **placeholder** "**Email**".
* An **"Age:"**
  + There must be a **placeholder** "**Age**".
* A button **"Enter"**.
  + Upon clicking, **successful** profile **creation** **redirects** to the **same** page, now displaying a home template withthe **user's profile**.
  + In case of any **errors**, the **form** will present relevant **validation** **messages**.

A screenshot of a computer

Description automatically generated

The **"Home with Profile"** page **displays** **all albums** created by the user. If **no albums exist**, only a paragraph is visible: **"No Albums in Catalog!"**

A black record with blue text

Description automatically generated

If a **profile** and **album**/**s** **exist**, the page should **include** the following **components**:

* A **navigation** bar, as displayed below.
* A headline titled **"All Albums"**
* A **card** for each **album**, featuring the album's **image**, **name**, **artist**, **genre**, and **price** (**formatted** to the **second decimal place**). Each card-box should have a **"Details"** **button**, leading to the **details page** for the selected album.

A screenshot of a music album

Description automatically generated

### Add Album Page

Template file: **"album-add.html"**

This page **displays** an **album** **creation form** containing:

* An **"Album Name:"**
  + There must be a placeholder "**Album Name**".
* An **"Artist:"**
  + There must be a placeholder "**Artist**".
* A **"Genre:"**
  + The user is **required** to **select** one option from the **provided** **choices** in the **drop-down** list.
* A **"Description:"**
  + There must be a placeholder "**Description**".
* An **"Image URL:"**
  + There must be a placeholder "**Image URL**".
* A **"Price:"**
  + There must be a placeholder "**Price**".
* A button **"Add New Album"**
  + Upon clicking, **successful** album **creation** **redirects** to the **home** page, **displaying** a template for a **home** page **with** the user's **profile**.
  + In case of any **errors**, the **form** will **present** the relevant **validation messages**.
* **Associate** the new **album** with the **user profile** you already have in the database.

A record album with a record

Description automatically generated with medium confidence

### Album Details Page

Template file: **"album-details.html"**

This page **showcases** the **details** of the **selected** **album**, including the following information:

* The **album's** **image**
* The **album's name**
* The **artist's name**
* The **genre** of the **album**
* The **price** of the album, **formatted** to the **second decimal place**
* The page will include the **description** **if provided**; otherwise, this line will be **omitted**
* An **"Edit"** **button** that leads to the **Edit album page**
* A **"Delete"** **button** that leads to the **Delete album page**

A screenshot of a music application

Description automatically generated

### Edit Album Page

Template file: **"album-edit.html"**

On the page, the **form** should be **pre-filled** with the data of the **selected** **album** for **editing**.

**Note** that the **price** should **have no limit** on the **number of decimal places** both when **displayed** and during **editing**.

When the **"Edit Album"** button is clicked:

* If the **album** is **successfully** **edited**, **redirection** to the **home** page occurs, **displaying** a template for a **home** page **with** the **user's profile**.
* Otherwise, the **form** will **present** relevant **validation** **errors**.

A record album with a record on the screen

Description automatically generated with medium confidence

### Delete Album Page

Template file: **"album-delete.html"**

On this page, the **form** should be **populated** with the **album's data**, and the **fields** should be **read-only**.

Clicking the **"Delete"** button will **remove** the **album** from the **database**, and you will be **redirected** to the **home** page, displaying a template for a **home page** **with** the **user's profile**.

The **deleted** **album** should **no longer** be **visible** in the app.

A record album with a record box

Description automatically generated with medium confidence

### Profile Details Page

Template file: **"profile-details.html"**

This page displays the **user's information**, including:

* The **default** **profile** **image (**as a **source link** in the **HTML code)**
* The **username**
* The **email**
* The **age** (if provided); **otherwise**, this line is **omitted**
* The **total number** of **albums** that the **user** currently has. Display **0** (**zero**) if the user has **no albums**
* A **"Delete"** button that leads to the **Delete profile page**

A screenshot of a computer

Description automatically generated

### Delete Profile Page

Template file: **"profile-delete.html"**

Deleting a profile should delete the profile **info** and **all associated albums.** After **deletion**, the user should be redirected to the **home page with no profile**.

A black record on a black background

Description automatically generated