



Übung 4 – Klassen-basierte Views & css

Lernziele: In dieser Übung lernen wir, wie man:

- ein Model gestaltet und
 - die Funktionalitäten zum List / Detail / Create / Delete per Klassen-basierte Views implementiert und
 - die Website mit css verschönert.
-

Aufgabe 1 – List / Detail / Create

Gestalten Sie ein **Model** zum Computer-Game. Inhaltlich gibt es mindestens:

- Name.
- Beschreibungs-Text.
- Genre – Benutzen Sie ein paar festgeschriebene Auswahlen, z.B. “first-person shooter”, “role playing”, “puzzle” usw. Hier können Sie gerne mit Ihrer Kreativität angeben :)
- Ersteller – Der Benutzer, der das Computer-Game eingetragen hat.
- Erstellungs-Zeitstempel. **Hinweis:** Benutzen Sie den Field-Type `DateTimeField`.

Implementieren Sie die folgenden Funktionalitäten per Klassen-basierte Views, die die Computer-Games verarbeiten:

- **List:** Eine Seite, die alle Computer-Games auflistet.
- **Detail:** Eine Seite, die die Details zu einem bestimmten Computer-Game auflistet.
- **Create:** Eine Seite, mit der man ein neues Computer-Game erstellt.

Hinweis: Zum Inspirieren wenden Sie sich gerne den Beispiel-Code in der Vorlesung an.

Aufgabe 2 – Delete

Implementieren Sie eine Seite zum **Delete** Funktionalität, d.h. Computer-Game löschen, auch per Klassen-basierte View:

- Auf der Detail-Seite sollte es einen Link geben. Wenn der Benutzer den Link klickt, wird eine neue Seite zum Zweck “bitte das Löschen bestätigen” gezeigt.
- Der Benutzer muss weiter einen Link oder eine Taste klicken, bevor das Computer-Game gelöscht wird.
- Nachdem das Computer-Game gelöscht ist, wird die List-Seite gezeigt. Selbstverständlich sollte jetzt das gelöschte Computer-Game weg sein.

Hinweise:

- Ein Klassen-basiertes View zum Delete haben Sie nicht in der Vorlesung gesehen, aber so was ist bei Django vorhanden. Schauen Sie in der Django-Dokumentation an:
<https://docs.djangoproject.com/en/3.0/ref/class-based-views/generic-editing/>
- Wie kann man einen Link zu dieser neuen Delete-Seite bauen? Schauen Sie die Links bei `book-list.html`.

Aufgabe 3 – css

Setzen Sie mindestens eine `css`-Datei ein. Verschönen Sie mindestens 3 Elemente in Ihrer Website, z.B. Text-Art und -größe, Text-Farbe, Hintergrund-Farbe, spacing, usw.

Abgabe

Laden Sie Ihr Django-Projekt als eine Zip-Datei hoch, aber ohne die Datenbank und ohne venv. Dazu müssen Sie auch Screenshots zum List / Detail / Create / Delete hochladen.