
Aula 03

A Plataforma Django

➔ O modelo de dados e o Django ORM

1. Implemente o modelo de dados presente nos slides e proceda para que este fique acessível pelo Django Admin Site.
2. Usando o Django Admin Site, preencha a BD com alguns dados à sua escolha (Author, Publisher e Book).
3. Implemente páginas web em Django, à custa de *views* e *templates*, que permitam as seguintes funcionalidades:
 - a. uma página com a lista dos títulos de todos os livros;
 - b. na página anterior, o título de cada livro deve ser um link para outra página com os detalhes desse livro;
 - c. uma página com a lista dos nomes de todos os autores;
 - d. na página anterior, o nome de cada autor deve ser um link para outra página com os detalhes desse autor;
 - e. uma página com a lista dos nomes de todas as editoras;
 - f. na página anterior, o nome de cada editora deve ser um link para outra página com os detalhes dessa editora
 - g. na página com a lista de autores, à frente do nome de cada autor, deve encontrar-se um link “Livros” para uma página com a lista de livros desse autor;
 - h. na página com a lista de editoras, à frente do nome de cada editora, deve encontrar-se um link “Autores” para uma página com a lista de autores dessa editora.

4. Reimplemente o exercício da aula anterior (eurocv) utilizando agora a base de dados.

Para este efeito, deve:

- a. Criar um modelo de dados para armazenar os dados.
- b. Através da interface do Django Admin Site, proceder à inserção de dados na BD.
- c. Alterar a “view” que mostra o eurocv, por forma a que esta proceda às pesquisas necessárias na BD e passe os dados à “template”.
- d. Fazer as eventuais alterações necessárias à “template” para receber os dados.

Lesson 03

Django Framework

➔ The Data Model and the Django ORM

1. Develop the data model, presented on the slides, and make it accessible through Django Admin Site.
2. Using Django Admin Site, fill the DB with some data, on your choice, (authors, publishers, and books).
3. Develop some web pages in Django, with views and templates, which allow the following features:
 - a. a page with the list of titles of all books;
 - b. on the previous page, the title of each book must be a link to another page with the details of that book;
 - c. a page with the list of names of all authors;
 - d. on the previous page, the name of each author must be a link to another page with the details of that author;
 - e. a page with the list of names of all publishers;
 - f. on the previous page, the name of each publisher must be a link to another page with the details of that publisher;
 - g. on the page with the list of authors, in front of the name of each author, there must be a link “Books” to a page with the list of books from that author;
 - h. on the page with the list of publishers, in front of the name of each publisher, there must be a link “Authors” to a page with the list of authors from that publisher.
4. Using the exercise from past lesson, about eurocv, develop it again but with the database to store the cv information. For that, you must:
 - a. Create a data model to store the data.
 - b. Through Django Admin Site interface, insert the data in the DB.
 - c. Modify the “view” that shows the eurocv, to make queries to the DB and pass the

data to the “template”.

- d. Make any necessary modifications to the “template”, to receive the data.