

Lab 5

The terminal window shows the following command sequence:

```
PS E:\book_store> python -m venv venv
PS E:\book_store> E:\book_store\venv\Scripts\activate
```

Below the command history, the code editor displays the `models.py` file content:

```
book_store > books > models.py > Book
1 from django.db import models
2 from django.urls import reverse
3
4 class Author(models.Model):
5     name = models.CharField(max_length=100)
6     bio = models.TextField(blank=True)
7     birth_date = models.DateField(null=True, blank=True)
8     def __str__(self):
9         return self.name
10    def get_absolute_url(self):
11        return reverse('author_detail', kwargs={'pk':self.pk})
12
13 class Book(models.Model):
14     title = models.CharField(max_length=200)
15     author = models.ForeignKey(Author, on_delete=models.CASCADE, related_name='books')
16     description = models.TextField(blank=True)
17     published_date = models.DateField()
18     price = models.DecimalField(max_digits=6, decimal_places=2) # <-- ແກ້ວຂອງນີ້
19     isbn = models.CharField(max_length=20, blank=True)
20     def __str__(self):
21         return self.title
22     def get_absolute_url(self):
23         return reverse('book_detail', kwargs={'pk':self.pk})
```

ໄຟຣ් model

The terminal window shows the `urls.py` file content:

```
book_store > books > urls.py > ...
1 from django.urls import path
2 from .views import (
3     AuthorListView, AuthorDetailView, AuthorCreateView, AuthorUpdateView, AuthorDeleteView,
4     BookListView, BookDetailView, BookCreateView, BookUpdateView, BookDeleteView,
5     HomeView
6 )
7
8 urlpatterns = [
9     path('', HomeView.as_view(), name='home'), # ພັນ Home
10    path('authors/', AuthorListView.as_view(), name='author_list'),
11    path('authors/<int:pk>/', AuthorDetailView.as_view(), name='author_detail'),
12    path('authors/create/', AuthorCreateView.as_view(), name='author_create'),
13    path('authors/<int:pk>/update/', AuthorUpdateView.as_view(), name='author_update'),
14    path('authors/<int:pk>/delete/', AuthorDeleteView.as_view(), name='author_delete'),
15
16    path('books/', BookListView.as_view(), name='book_list'),
17    path('books/<int:pk>/', BookDetailView.as_view(), name='book_detail'),
18    path('books/create/', BookCreateView.as_view(), name='book_create'),
19    path('books/<int:pk>/update/', BookUpdateView.as_view(), name='book_update'),
20    path('books/<int:pk>/delete/', BookDeleteView.as_view(), name='book_delete'),
21 ]
22 ]
```

ໄຟຣ් urls

```
from django.views.generic import ListView, DetailView, CreateView,
UpdateView, DeleteView, TemplateView
from django.urls import reverse_lazy
from .models import Author, Book
from django.db.models import Q
# -----
# Author Views
# -----
class AuthorListView(ListView):
    model = Author
    template_name = 'books/author_list.html'
    context_object_name = 'authors'

class AuthorDetailView(DetailView):
    model = Author
    template_name = 'books/author_detail.html'
    context_object_name = 'author'

class AuthorCreateView(CreateView):
    model = Author
    template_name = 'books/author_form.html'
    fields = ['name', 'bio', 'birth_date']
    success_url = reverse_lazy('author_list')

class AuthorUpdateView(UpdateView):
    model = Author
    template_name = 'books/author_form.html'
    fields = ['name', 'bio', 'birth_date']
    success_url = reverse_lazy('author_list')

class AuthorDeleteView(DeleteView):
    model = Author
    template_name = 'books/author_confirm_delete.html'
    success_url = reverse_lazy('author_list')

# -----
# Book Views
# -----
class BookListView(ListView):
    model = Book
```

```
template_name = 'books/book_list.html'
context_object_name = 'books'

def get_queryset(self):
    queryset = super().get_queryset()
    query = self.request.GET.get('q')
    if query:
        queryset = queryset.filter(
            Q(title__icontains=query) |
            Q(author__name__icontains=query)
        )
    return queryset

class BookDetailView(DetailView):
    model = Book
    template_name = 'books/book_detail.html'
    context_object_name = 'book'

class BookCreateView(CreateView):
    model = Book
    template_name = 'books/book_form.html'
    fields = ['title', 'author',
              'published_date', 'price', 'isbn', 'description']
    success_url = reverse_lazy('book_list')

class BookUpdateView(UpdateView):
    model = Book
    template_name = 'books/book_form.html'
    fields = ['title', 'author', 'published_date', 'isbn']
    success_url = reverse_lazy('book_list')

class BookDeleteView(DeleteView):
    model = Book
    template_name = 'books/book_confirm_delete.html'
    success_url = reverse_lazy('book_list')

class HomeView(TemplateView):
    template_name = 'books/home.html'
```

```
def get_context_data(self, **kwargs):
    context = super().get_context_data(**kwargs)
    context['books'] = Book.objects.all()[:4]
    return context
```

ไฟล์ view

```
book_store > books > admin.py > ...
1  from django.contrib import admin
2  from .models import Author, Book
3
4  @admin.register(Author)
5  class AuthorAdmin(admin.ModelAdmin):
6      list_display= ['name','birth_date']
7      search_fields = ['name']
8
9  @admin.register(Book)
10 class BookAdmin(admin.ModelAdmin):
11     list_display = ['title','author','published_date','price']
12     list_filter = ['author','published_date']
13     search_fields = ['title','author__name']
14 # Register your models here.
15
```

Admin

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>{% block title %}Book Store{% endblock %}</title>
<!-- Bootstrap 5 CSS -->
<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet">
<!-- Google Fonts -->
<link
href="https://fonts.googleapis.com/css2?family=Roboto:wght@400;500;700&family=Poppins:wght@500;700&display=swap" rel="stylesheet">
<style>
    body {
```

```
font-family: 'Poppins', sans-serif;
background: linear-gradient(to right, #f8f9fa, #e9ecef);
min-height: 100vh;
display: flex;
flex-direction: column;
}

/* Navbar */
.navbar {
    background: linear-gradient(90deg, #212529, #343a40);
    box-shadow: 0 3px 10px rgba(0,0,0,0.3);
}

.navbar-brand {
    font-weight: 700;
    font-size: 1.6rem;
    letter-spacing: 1px;
    color: #ffc107 !important;
}

.nav-link {
    font-weight: 500;
    margin-right: 1rem;
    color: #f8f9fa !important;
    transition: all 0.3s ease;
}

.nav-link:hover {
    color: #ffc107 !important;
    transform: translateY(-2px);
}

/* Container */
.container {
    flex: 1;
    padding-top: 2rem;
    padding-bottom: 2rem;
}

/* Card Style */
```

```
.card {  
    border: none;  
    border-radius: 15px;  
    box-shadow: 0 6px 18px rgba(0,0,0,0.1);  
    transition: all 0.3s ease-in-out;  
}  
  
.card:hover {  
    transform: translateY(-6px);  
    box-shadow: 0 10px 25px rgba(0,0,0,0.2);  
}  
  
.card-title {  
    font-weight: 600;  
    color: #212529;  
}  
  
/* Footer */  
footer {  
    background: #212529;  
    color: #dee2e6;  
    text-align: center;  
    padding: 15px 0;  
    margin-top: auto;  
}  
  
footer a {  
    color: #ffc107;  
    text-decoration: none;  
    font-weight: 500;  
}  
footer a:hover {  
    text-decoration: underline;  
}  
</style>  
</head>  
<body>  
<!-- Navbar -->  
<nav class="navbar navbar-expand-lg">  
    <div class="container">
```

```

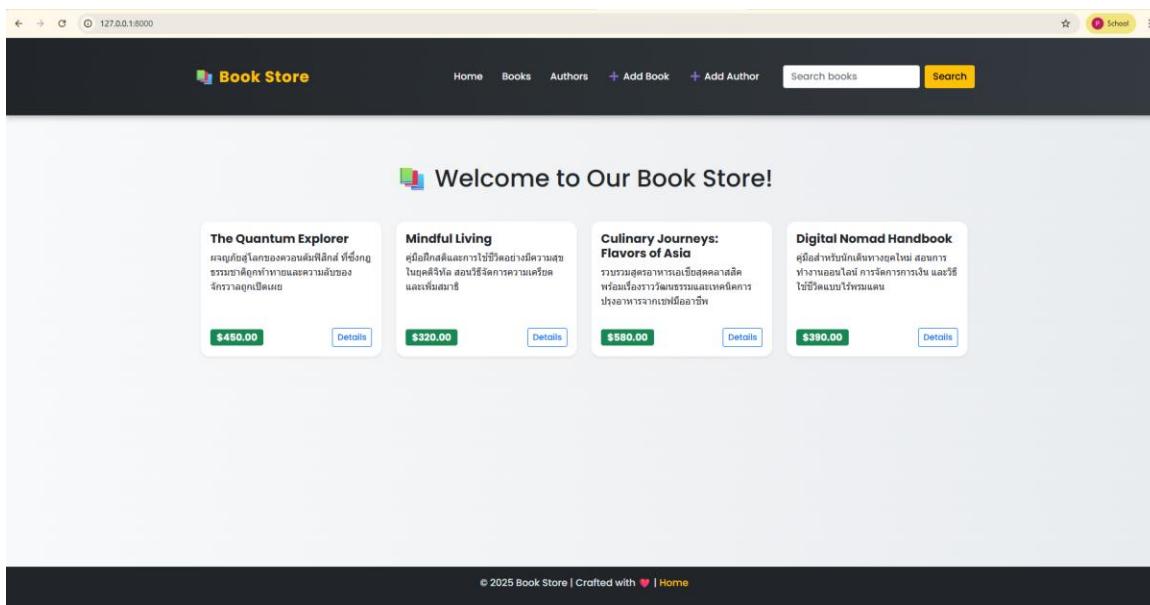
<a class="navbar-brand" href="{% url 'home' %}">📖 Book Store</a>
<button class="navbar-toggler text-light" type="button" data-bs-
toggle="collapse" data-bs-target="#navbarNav"
       aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle
navigation">
    <span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarNav">
    <div class="navbar-nav ms-auto">
        <a class="nav-link" href="{% url 'home' %}">Home</a>
        <a class="nav-link" href="{% url 'book_list' %}">Books</a>
        <a class="nav-link" href="{% url 'author_list' %}">Authors</a>
        <a class="nav-link" href="{% url 'book_create' %}">➕ Add Book</a>
        <a class="nav-link" href="{% url 'author_create' %}">➕ Add
Author</a>
    </div>
    <form class="d-flex ms-3" method="GET" action="{% url 'book_list'
%}">
        <input class="form-control me-2" type="search" placeholder="Search
books" aria-label="Search" name="q" value="{{ request.GET.q }}">
        <button class="btn btn-warning" type="submit">Search</button>
    </form>
</div>
</div>
</div>
</nav>

<!-- Main Content --&gt;
&lt;div class="container"&gt;
    {% block content %}
    {% endblock %}
&lt;/div&gt;

<!-- Footer --&gt;
&lt;footer&gt;
    &lt;p&gt;© 2025 Book Store | Crafted with ❤ | &lt;a href="{% url 'home'
%}"&gt;Home&lt;/a&gt;&lt;/p&gt;
&lt;/footer&gt;
</pre>

```

base.html



Home

The screenshot shows the "Books" page of the application. The header is identical to the home page. The main content displays a list of books in a grid format:

- The Quantum Explorer** by phu - \$450.00
- Mindful Living** by J.K. Rowling - \$320.00
- Culinary Journeys: Flavors of Asia** by George Orwell - \$580.00
- Digital Nomad Handbook** by phu - \$390.00
- Legends of the Forgotten Realm** by phu - \$520.00

Each book card includes "View Details", "Edit", and "Delete" buttons. At the bottom, there is a footer with the text "© 2025 Book Store | Crafted with ❤️ | Home".

Book

The screenshot shows the 'Authors' page of a web application. At the top, there is a navigation bar with links for Home, Books, Authors, Add Book, Add Author, and a search bar. Below the navigation bar, the title 'Authors' is displayed. A button labeled 'Add New Author' is visible. The main content area lists three authors: J.K. Rowling, George Orwell, and phu. Each author entry includes a 'View Details' link, an 'Edit' button, and a 'Delete' button. To the right of each entry, it indicates the number of books associated with that author: '1 book(s)' for J.K. Rowling, '1 book(s)' for George Orwell, and '3 book(s)' for phu.

Author

The screenshot shows the 'Add Book' page of the web application. At the top, there is a navigation bar with links for Home, Books, Authors, Add Book, Add Author, and a search bar. Below the navigation bar, the title 'Add Book' is displayed. The form fields for adding a book include: Title (input field), Author (dropdown menu), Published date (input field), Price (input field), ISBN (input field), and a large Description area (text area). Below the description area is a file upload input field. At the bottom of the form are 'Save' and 'Cancel' buttons.

Add book

127.0.0.1:8000/authors/create/

Book Store

Home Books Authors + Add Book + Add Author Search books Search

Add Author

Name:

Bio:

Birth date:

[Save](#) [Cancel](#)

© 2025 Book Store | Crafted with ❤️ | Home

Add Author

127.0.0.1:8000/books/?q=phu

Book Store

Home Books Authors + Add Book + Add Author phu Search

Books

[Add New Book](#)

The Quantum Explorer
by phu
หนังสือที่โลกของความคิดเห็นที่มีอยู่ที่สั่นสะเทือนมากที่สุด
ที่ทำให้เราต้องหันหน้าไปทางใหม่ สมองการทำงานของเราในปัจจุบัน

\$450.00

[View Details](#) [Edit](#) [Delete](#)

Digital Nomad Handbook
by phu
คู่มือสำหรับนักเดินทางดิจิตอลใหม่ สอนการพากษางานออนไลน์ การจัดการงานอิเล็กทรอนิกส์ และวิธีใช้ชีวิตแบบไม่พึ่งบ้านและเมือง

\$390.00

[View Details](#) [Edit](#) [Delete](#)

Legends of the Forgotten Realm
by phu
นิยายแฟนตาซีที่ท้าทายไปอีกขั้น โลกที่หายสาบสูญ การผจญภัย และการต่อสู้เพื่อกอบกู้อาณาจักรที่ถูกลืม

\$520.00

[View Details](#) [Edit](#) [Delete](#)

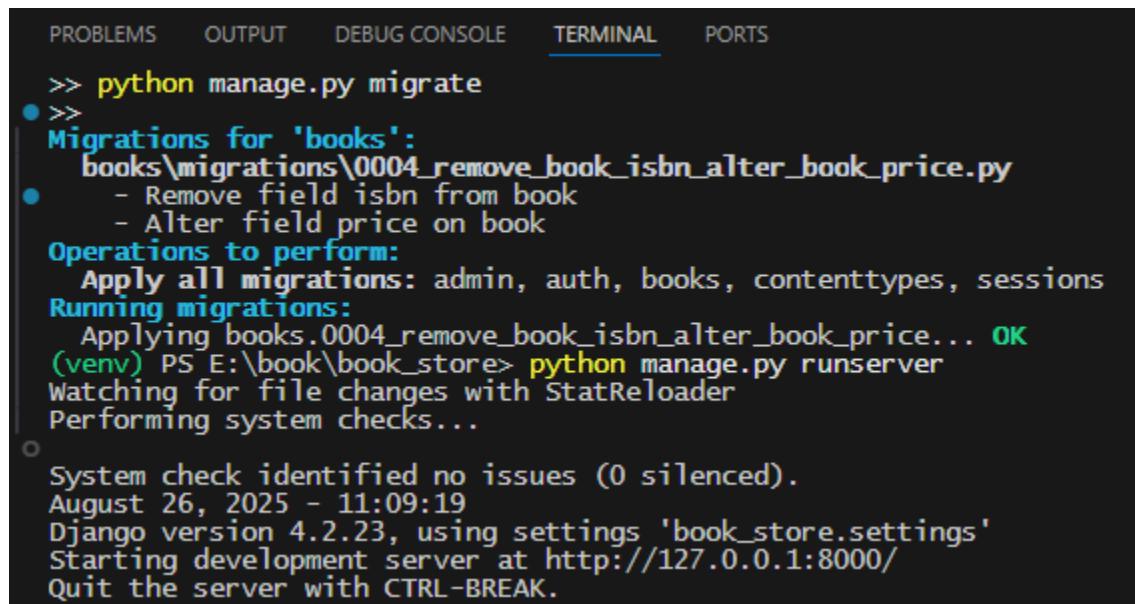
© 2025 Book Store | Crafted with ❤️ | Home

Search

```
class BookListView(ListView):
    model = Book
    template_name = 'books/book_list.html'
    context_object_name = 'books'

    def get_queryset(self):
        queryset = super().get_queryset()
        query = self.request.GET.get('q')
        if query:
            queryset = queryset.filter(
                Q(title__icontains=query) | Q(author__name__icontains=query)
            )
        return queryset
```

Code Search



The screenshot shows a terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
>> python manage.py migrate
>>
Migrations for 'books':
  books\migrations\0004_remove_book_isbn_alter_book_price.py
  ● - Remove field isbn from book
    - Alter field price on book
Operations to perform:
  Apply all migrations: admin, auth, books, contenttypes, sessions
Running migrations:
  Applying books.0004_remove_book_isbn_alter_book_price... OK
(venv) PS E:\book\book_store> python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...
System check identified no issues (0 silenced).
August 26, 2025 - 11:09:19
Django version 4.2.23, using settings 'book_store.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

migrate