Notes:

Static template layout

templates/

├── base.html # base layout

├── books/

│ └── book\_list.html # where your book cards live

static/

└── your app css/js…

base :

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>{% block title %}Books{% endblock %}</title>

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<script src="https://cdn.tailwindcss.com"></script>

<style>

@keyframes fadeInUp {

0% { opacity: 0; transform: translateY(20px); }

100% { opacity: 1; transform: translateY(0); }

}

.fade-in-up { animation: fadeInUp 0.5s ease-out forwards; }

.bookmark-icon {

transition: transform 0.2s ease;

}

.bookmarked svg {

fill: #facc15;

stroke: #f59e0b;

}

</style>

{% block head %}{% endblock %}

</head>

<body class="bg-gray-100 min-h-screen">

<div class="max-w-6xl mx-auto p-6">

{% block content %}{% endblock %}

</div>

<script>

function toggleExpand(card) {

const content = card.querySelector('.content');

const icon = card.querySelector('svg:not(.bookmark-icon)');

content.classList.toggle('hidden');

icon.classList.toggle('rotate-180');

}

function toggleBookmark(button, bookId) {

let favorites = JSON.parse(localStorage.getItem('favorites')) || {};

const isBookmarked = favorites[bookId];

if (isBookmarked) {

delete favorites[bookId];

button.classList.remove('bookmarked');

} else {

favorites[bookId] = true;

button.classList.add('bookmarked');

}

localStorage.setItem('favorites', JSON.stringify(favorites));

}

document.addEventListener('DOMContentLoaded', () => {

let favorites = JSON.parse(localStorage.getItem('favorites')) || {};

document.querySelectorAll('.bookmark-button').forEach(button => {

const bookId = button.getAttribute('data-book-id');

if (favorites[bookId]) {

button.classList.add('bookmarked');

}

});

});

</script>

{% block scripts %}{% endblock %}

</body>

</html>

Book list:

{% extends "base.html" %}

{% block title %}Book List{% endblock %}

{% block content %}

<!-- Search bar -->

<div class="flex justify-between items-center mb-6">

<div class="flex items-center gap-2 bg-purple-200 rounded-full px-4 py-2 w-2/3">

<button>

<svg class="w-5 h-5 text-gray-700" fill="none" stroke="currentColor" viewBox="0 0 24 24">

<path stroke-linecap="round" stroke-linejoin="round" stroke-width="2"

d="M4 6h16M4 12h16M4 18h16" />

</svg>

</button>

<input type="text" placeholder="Search books"

class="bg-transparent focus:outline-none w-full text-sm text-gray-700" />

<svg class="w-5 h-5 text-gray-700" fill="none" stroke="currentColor" viewBox="0 0 24 24">

<path stroke-linecap="round" stroke-linejoin="round" stroke-width="2"

d="M21 21l-4.35-4.35M17 11a6 6 0 11-12 0 6 6 0 0112 0z" />

</svg>

</div>

</div>

<!-- Book cards -->

<div class="grid sm:grid-cols-2 lg:grid-cols-3 gap-6">

{% for book in books %}

<div class="bg-white rounded-3xl p-4 shadow-md fade-in-up relative" style="animation-delay: {{ forloop.counter0|add:"0.1"|floatformat:1 }}s;" onclick="toggleExpand(this)">

<!-- Bookmark Button -->

<button class="absolute top-3 right-3 bg-white rounded-full p-1 shadow bookmark-button" data-book-id="{{ book.id }}"

onclick="event.stopPropagation(); toggleBookmark(this, '{{ book.id }}')">

<svg class="w-5 h-5 text-yellow-500 bookmark-icon" fill="none" stroke="currentColor" stroke-width="2"

viewBox="0 0 24 24">

<path stroke-linecap="round" stroke-linejoin="round" d="M5 3v18l7-5 7 5V3z" />

</svg>

</button>

<img src="{{ book.cover\_url }}" alt="{{ book.title }}" class="rounded-xl w-full h-64 object-cover mb-4">

<div class="flex justify-between items-center">

<h2 class="text-lg font-semibold text-purple-800">{{ book.title }}</h2>

<svg class="w-5 h-5 text-purple-800 transition-transform" fill="none" stroke="currentColor"

viewBox="0 0 24 24">

<path stroke-linecap="round" stroke-linejoin="round" stroke-width="2" d="M19 9l-7 7-7-7" />

</svg>

</div>

<div class="content hidden mt-2 text-sm text-gray-700">

{{ book.description }}

</div>

</div>

{% endfor %}

</div>

{% endblock %}

Views:

from django.shortcuts import render

def book\_list(request):

books = [

{

"id": "1984",

"title": "1984 by George Orwell",

"cover\_url": "https://covers.openlibrary.org/b/id/8228691-L.jpg",

"description": "A dystopian novel set in a totalitarian society ruled by Big Brother."

},

{

"id": "mockingbird",

"title": "To Kill a Mockingbird",

"cover\_url": "https://covers.openlibrary.org/b/id/8225261-L.jpg",

"description": "A profound story of racial injustice and childhood innocence."

},

{

"id": "gatsby",

"title": "The Great Gatsby",

"cover\_url": "https://covers.openlibrary.org/b/id/10252554-L.jpg",

"description": "A jazz-age novel about wealth, illusion, and the American dream."

},

# Add more books here...

]

return render(request, "books/book\_list.html", {"books": books})

URLS:

# urls.py

from django.urls import path

from . import views

urlpatterns = [

path('books/', views.book\_list, name='book\_list'),

]