Code Files Explanation

Project: School Activity Booking System

Simple Map of Our Files

Think of our project folder like a house.
Here is what is in every room:

---

1. The Main Room: `app.py`

This is the Brain.
- It contains almost everything.
- It decides what happens when you click buttons.
- It talks to the database.
- It sends emails.
- **Size**: 1,200+ lines of code!

Sections of the Brain:
- **Lines 1-50**: Imports (Getting tools)
- **Lines 50-170**: Database Models (Defining what a "User" is)
- **Lines 170-200**: Security (The Bouncers)
- **Lines 200-400**: Email & PDF Logic (The Messengers)
- **Lines 400-1200**: Routes (The Pages - Login, Dashboard, Admin, etc.)

---

2. The Settings: `config.py`

This is the Control Panel.
- It holds the secrets.
- It tells the app: "Use this database", "Use this email".
- It keeps secrets safe (reads from `.env` file).

---

3. The Decoration: `static/` Folder

This is the Paint and Furniture.
- **`css/style.css`**: The Styling.
- "Make buttons blue"
- "Make cards see-through"
- "Make text big"
- **`uploads/`**: The Storage.
- Where user profile pictures are kept.

---

4. The Blueprints: `templates/` Folder

These are the HTML Skeletons.
- They are web pages with holes in them.
- Python fills the holes with real data.

Examples:

- `base.html`: The frame (Header, Footer, Menu). Every page uses this.
- `dashboard.html`: The parent's main screen.
- `login.html`: The login box.
- `invoice_template.html`: The design for the PDF.
---

5. The Memory: `booking_system_v2.db`

This is the Filing Cabinet.
- It is a SQLite database file.
- It stores:
- Every user
- Every booking
- Every activity
- If you delete this, everyone is forgotten!
---

6. The Builder: `populate_db.py`

This is the Setup Robot.
- You run this once.
- It builds the database.
- It creates fake users (Demo data) so the site isn't empty.
- It creates: 1 Admin, 1 Parent, 6 Tutors, 8 Activities.
---

7. The Shopping List: `requirements.txt`

This is the Ingredients List.
- Tells the computer what to download.
- "I need Flask"
- "I need ReportLab"
- "I need SQLAlchemy"
---

8. The Instructions: `Procfile`

This is the Note for the Server.
- Tells the internet server (Render) how to start.
- Says: "Please run `gunicorn app:app` to start the website."
---

9. Helper Scripts

These are the Utility Tools.
- **`populate_db.py`**: Creates sample data (1 admin, 1 parent, 6 tutors, 8 activities)
- **`convert_to_pdf.py`**: Converts markdown documentation to PDF format
- **`view_attendance.py`**: Quick script to view all attendance records in database
- **`create_sanchit_parent.py`**: Creates a specific parent account with child
- **`update_sarah_email.py`**: Updates tutor email in database
- **`upload_to_notion.py`**: Automates uploading project plan to Notion workspace
---

10. Templates

Complete List of HTML Pages:
Base:
- `base.html` - Main layout frame (header, footer, navigation)
Parent Portal:
- `index.html` - Homepage
- `login.html` - Parent login
- `register.html` - Parent registration
- `dashboard.html` - Parent dashboard
- `activities.html` - Browse activities
- `payment.html` - Payment page
- `invoice_template.html` - PDF invoice design
Admin Portal:
- `admin/login.html` - Admin login
- `admin/dashboard.html` - Admin overview with statistics
- `admin/activities.html` - Manage activities (CRUD)
- `admin/tutors.html` - Manage tutors (CRUD)
- `admin/bookings.html` - View all bookings
Tutor Portal:
- `tutor/login.html` - Tutor login
- `tutor/dashboard.html` - Tutor overview
- `tutor/attendance.html` - Mark attendance
- `tutor/attendance_history.html` - View past attendance records
Shared:
- `error.html` - Custom 404/500 error pages
- `admissions.html` - School admissions information

Summary

File	Analogy	Purpose
`app.py`	The Brain	Logic & Rules (31 routes)
`config.py`	Control Panel	Settings
`templates/`	Skeletons	Page Layouts (18 templates)
`static/`	Paint	Colors & Styles
`database.db`	Memory	Storing Data
`populate.py`	Robot	Creating Sample Data
`Helper Scripts`	Tools	Maintenance & Automation

Total Project Size:
- **1,300+ lines** of Python code
- **18 HTML templates**
- **1 CSS file** with 500+ lines
- **7 Database models**
- **31 Routes** (endpoints)