Team 4 – CS551Q Project Overview

This web application visualizes PM2.5 air pollution levels across different countries and years using data from an Excel dataset. Users can either:

- View pollution levels for a single country, or
- Compare levels between two countries through an interactive chart interface.

Team Members: Julius Sarah, Selu 'Ita, Minli Yu, Zizhuo Dou, Zhouming Liu

Architecture

- Framework: Django 4.x
- Frontend: HTML (Django Templates), Chart.js for visualizations
- Backend: Python-based Django views and models
- Database: SQLite (development), PostgreSQL (production)
- Deployment: Render cloud platform < https://team-4-deployment-1.onrender.com/>

Key Components

- Data Models:
  - *Country*: Stores country information
  - *PM25Record*: Yearly PM2.5 measurements linked to countries
- Views & Pages:
  - *Homepage*: Navigation hub for single country lookup or comparison
  - *Country Search*: Dropdown with autocomplete
  - *Comparison Tool*: Bar chart for visual comparison
- Data Processing:
  - Custom management command (parse_country) to import Excel data
  - Server-side data handling via Django views
  - Minimal JavaScript for rendering charts using Chart.js

Implementation Approach

Data Flow

1. Excel data parsed and loaded into the database
2. User selects a country/year from dropdowns
3. Django views retrieve and process data
4. Processed data passed to templates
5. Chart.js renders visualizations

User Experience

- Responsive design for multiple screen sizes
- Enhanced dropdowns with search for quick selection
- Clear, labelled PM2.5 charts and data display
- Seamless navigation between features

Deployment Strategy

Development

- Local development using SQLite
- Team collaboration via Codio
- Git for version control

Production

1. Database migration to PostgreSQL
2. Deployment to Render cloud platform < https://team-4-deployment-1.onrender.com/>
3. Environment variable setup for secure configs
4. Static files served via cloud storage
5. HTTPS enabled for security

Technical Challenges & Solutions

- Database Migration: Planned smooth transition from SQLite to PostgreSQL
- Visualization: Implemented Chart.js for dynamic comparisons
- Performance: Prioritized server-side logic to reduce client-side load
- Deployment: Shifted from Codio to a stable cloud-hosted platform

Future Enhancements

- Geographic map-based visualizations of PM2.5 levels
- Time-series trend analysis across multiple years

!!!!!Sign-in details to activate Render App=

user: seluvaiasariahita@gmail.com; temp-pswrd: secret1234!