Kindly, check the finer version in WU QIXUAN codio.

GROUP GOLF: WU QIXUAN, PUTCHENKO KATERYNA, SHINDE SHIVANJALI, ANKOMAH CHARLES BOTA.

Render: https://groupproject-e6ub.onrender.com

1. Introduction

This dataset provides an in-depth look into the global CO2 emissions at the country level, allowing for a better understanding of how much each country contributes to the global cumulative human impact on climate. It contains information on total emissions as well as from coal, oil, gas, cement production and flaring, and other sources.

2. Built database.

Basically "emission app" has the record of emissions done by different countries with respect to oil, gas, cement, flaring etc. While creating data base we sorted the data in a proper way to make it easier further. The 'Primary' & 'Foreign' keys are two important elements. After combining tables & rows the data can be parsed and passed further.

3. Template

Templates is a basic HTML page with a navigation bar, a title, and a table. The navigation bar includes two links to create an account and log in. The table displays data related to emissions by country with various parameters such as serial number, total emissions, and emissions from coal, oil, gas, cement, flaring, and other sources. The HTML code also includes some CSS styling.

4. Connect Database and Templates.

In this application, we have analyzed a Django web application's views and URL configuration. The application consists of five views that render different HTML templates, and five URL patterns that map the requested URLs to the corresponding views.

5. Test

Testing is a crucial part of any software development process. Django comes with a built-in testing framework that makes it easy to write tests. For this application, we have written unit tests that test models.py, in particular testing both "country-data" and "value" classes. These tests ensure that the data is being saved and retrieved correctly from the database.

6. Render Deployment

Deploying a cloud to render in the cloud is a powerful way to leverage the scalability and flexibility of cloud computing for rendering. To deploy the code to the cloud need choose your cloud platform, in this instance render was chosen as the cloud platform for the deployment of our code. Render was connected to the GitHub account to source the codes. Render setup involves configuring the render settings, and output format. After deploying, codes were monitored to run efficiently and successfully rendered.

7. Conclusion

In summary, the application provides detailed information about global CO2 emissions at the country level. The database was built by sorting and combining the data using primary and foreign keys. The HTML template was created to display the data in a table with CSS styling. The Django web application was used to connect the database and templates through views and URL configuration. Unit tests were written to ensure the accuracy of data storage and retrieval. The cloud platform Render was chosen for deployment, and the code was monitored for efficient rendering.