GlobalLandTemperaturesByCity Data Report

Render URL:https://assessment-odbh.onrender.com/

Git URL:https://github.com/lrh500/Assessment

Introduction

This report provides an overview of the code written for a climate change monitoring application. The application is built using the Django web framework and uses Plotly for data visualization.

Design(Code Structure):

The codebase is organized into several modules, including:

Models:

The climate_change.models module defines the data models used by the application. The country_id model represents a country and its unique identifier, while the Global_tem model represents the average temperature of a city in a particular country on a specific date. The Globaltem_change model represents the change in global temperature over time.

Views:

The climate_change.views module defines the views for the application. The index view renders the homepage of the application. The country_list view displays a list of all the countries in the application's database. The global_tem_list view displays a list of all the global temperature records in the application's database. The global_tem_change_view view displays the change in global temperature over time. The country_by_name view displays a list of cities in a particular country, sorted by their average temperature. It also displays a Plotly chart of the average temperature of each city over time. The check_by_date view displays a list of countries that have recorded the average temperature on a particular date. It also displays a Plotly chart of the average temperature of each country on that date.

Data Visualization:

The application uses Plotly to create interactive data visualizations. The country_by_name view creates a scatter plot of the average temperature of each city in a particular country over time. The check_by_date view creates a scatter plot of the average temperature of each country on a particular date.

Development:

This code uses Django as a web framework to store climate change data in a database via models.py. The data is then queried and visualized using Plotly, Matplotlib, and other libraries in the view functions. This website allows users to easily browse and compare climate trends in different countries or regions, and query climate data at a specific time or place. It promotes environmental awareness and encourages a response to the issue of climate change.

Conclusion:

This report has provided an overview of the code written for a climate change monitoring application. The application is built using the Django web framework and uses Plotly for data visualization. The code is organized into several modules, including models, views, and templates. The application loads data from CSV files into a database and provides interactive data visualizations for exploring the data.