Phase-2 end project

Railway Crossing Status

Writeup

- The Railway Crossing Management application aims to provide a prototype for effectively managing railway crossings and their statuses. This write-up provides an overview of the code, explaining its purpose, features, and functionality.
- The code demonstrates the implementation of a Railway Crossing Management application using Java. It consists of several classes that collectively create and manage railway crossings, their details, and their statuses.
- The RailwayCrossing class represents a single railway crossing and contains attributes such as name, address, landmark, train schedules, person in charge, and status. It provides getter and setter methods to access and modify these attributes.
- The RailwayCrossingApp class serves as the main application class. It maintains a list of railway crossings and provides methods to add, delete, and update their statuses. It utilizes an ArrayList to store the railway crossings.
- The Main class serves as the entry point for the application. It creates an instance of the RailwayCrossingApp and demonstrates the functionality of the application prototype.
- Application Functionality:
-
- Creating Railway Crossings:
- The code initializes two instances of RailwayCrossing named crossing1 and crossing2 with relevant details such as name, address, landmark, train schedules, person in charge, and status. The railway crossings are then added to the RailwayCrossingApp using the addRailwayCrossing() method.
- Displaying Railway Crossings:
- The code retrieves the list of railway crossings from the RailwayCrossingApp using the getRailwayCrossings() method. It iterates through the list and displays the details of each crossing, including its name, address, landmark, train schedules, person in charge, and status.
- Updating Railway Crossing Status:
- The code uses the updateRailwayCrossingStatus() method of the RailwayCrossingApp to update the status of crossing1 to "closed". It verifies the updated status by retrieving and displaying the modified status of crossing1.
- Deleting Railway Crossing:
- The code removes crossing2 from the RailwayCrossingApp using the deleteRailwayCrossing() method. It confirms the deletion by displaying a message indicating the successful removal of the railway crossing.
- The provided code showcases the prototype implementation of a Railway Crossing Management application. It allows the creation, display, updating, and deletion of railway crossings, along with their associated details and statuses. This write-up serves to provide an overview of the code's functionality and demonstrates its ability to manage railway crossings efficiently. Further development and refinement of the

application can be undertaken based on the requirements and feedback from stakeholders.