Chinook Database Management Project Report

This project demonstrates a Python-based CRUD application using the Chinook Database. The system is designed to manage various entities such as Artists, Albums, Media Types, Genres, Tracks, Playlists, Customers, Employees, and Invoices. The project provides a command-line interface for interacting with the database, enabling efficient management of music-related data.

Objectives

- Implement CRUD operations (Create, Read, Update, Delete) for all major entities.
- Provide a modular design using separate view files for each entity.
- Ensure seamless interaction with the SQLite Chinook database.
- Create a dashboard to navigate between different management modules.
- Provide a simple and user-friendly CLI-based interface.

Modules Implemented

Module	Description
Artist Management	CRUD operations for artist details.
Album Management	CRUD operations for albums.
Track Management	CRUD operations for music tracks.
Playlist Management	Manage playlists and playlist tracks.
Media Type Management	CRUD for media types (e.g., MPEG, AAC).
Genre Management	CRUD for different music genres.
Customer Management	CRUD for customer information.
Employee Management	CRUD for employee records.
Invoice Management	CRUD for invoices and invoice items.
Dashboard	Central navigation menu to access all modules.

Technologies Used

- **Programming Language:** Python 3.13
- **Database:** SQLite (Chinook.db)
- **IDE:** Visual Studio Code
- **Libraries:** sqlite3, reportlab

Conclusion

This project successfully demonstrates a Python-based CLI application for managing the Chinook Database. It provides a scalable foundation for future improvements, such as integrating a GUI or a web-based front end. The modular structure ensures that new

entities can be added with minimal effort, making the system extendable and maintainable.