**CHAPTER-1**

**COMPANY PROFILE**

**Company Name: EZ Trainings and Technologies Pvt. Ltd.**

**Introduction:**

EZ Trainings and Technologies Pvt. Ltd. is a dynamic and innovative organization dedicated to providing comprehensive training solutions and expert development services. Established with a vision to bridge the gap between academic learning and industry requirements, we specialize in college trainings for students, focusing on preparing them for successful placements. Additionally, we excel in undertaking development projects, leveraging cutting-edge technologies to bring ideas to life.

**Mission:**

Our mission is to empower the next generation of professionals by imparting relevant skills and knowledge through specialized training programs. We strive to be a catalyst in the career growth of students and contribute to the technological advancement of businesses through our development projects.

**Services:**

**College Trainings:**

• Tailored training programs designed to enhance the employability of students.

• Industry-aligned curriculum covering technical and soft skills.

• Placement assistance and career guidance.

**Development Projects:**

• End-to-end development services, from ideation to execution.

• Expertise in diverse technologies and frameworks.

• Custom solutions to meet specific business needs.

**Locations:** Hyderabad | Delhi NCR

At EZ Trainings and Technologies Pvt. Ltd., we believe in transforming potential into excellence.

**Abstract:**

The Box Office Report Program is a data analysis tool designed to provide insights into the earnings of films across different theaters .This program utilizes object-oriented programming principles and abstract classes to organize and analyze box office data efficiently.

The core components of the program include the Box\_office class, which serves as the abstract base class defining methods for creating, reading, updating, and deleting box office data entries. The Box\_office\_analysis class extends the Box\_office class, implementing methods for generating reports on box office earnings and comparing earnings across theaters.

Key features of the program include:

Data Management: The program allows for the creation, retrieval, updating, and deletion of box office data entries, providing flexibility in managing box office records.

Earnings Analysis: Users can generate reports on the total earnings and average earnings per theater for a specific film, as well as compare earnings across multiple theaters.

User Interaction: The program interacts with users through a command-line interface, enabling them to input commands for data analysis and manipulation.

The Box Office Report Program provides a user-friendly and efficient solution for analyzing box office data, offering valuable insights for film distributors, theater owners, and industry analysts.

**Introduction:**

The Box Office Report program is a sophisticated tool designed to facilitate the analysis of box office earnings for films across various theaters. In today's dynamic entertainment industry, understanding the performance of movies in theaters is essential for filmmakers, distributors, and theater owners alike. The Box Office Report program aims to provide comprehensive insights into box office data, empowering stakeholders to make informed decisions and optimize their strategies.

**Implementation Details:**

The program is implemented using object-oriented programming principles, with classes such as Box\_office and Box\_office\_analysis forming the core components. The Box\_office class serves as an abstract base class defining methods for data manipulation, while the Box\_office\_analysis class extends this functionality to provide detailed earnings reports and comparisons.

**Significance:**

The Box Office Report program holds significant importance in the entertainment industry by offering actionable insights into box office performance. Filmmakers can use the data to evaluate the success of their films and tailor marketing strategies accordingly. Distributors can optimize film distribution plans based on theater-specific earnings data, while theater owners can identify opportunities for revenue growth and operational improvements.

In summary, the Box Office Report program represents a valuable asset for industry professionals seeking to understand and leverage box office trends effectively. Its intuitive interface, robust data management capabilities, and comprehensive analytical features make it an indispensable tool in the realm of film distribution and exhibition.

**MODULE DESCRIPTION:**

The Main.py module serves as the entry point for the Box Office Report program. It orchestrates interactions with the Box\_office class and provides a command-line interface for users to perform various operations related to box office data analysis and management.

**Classes:**

1**. BoxOfficeAnalyzer:**

This class encapsulates the functionality of analyzing box office data, including reporting earnings, comparing earnings across theaters, creating, updating, and deleting entries.

Methods:

* report\_box\_office\_earnings(film\_id): Generates a detailed report on the earnings of a specific film.
* compare\_earnings\_across\_theaters(comparison\_data): Compares earnings across theaters based on the provided list of theater IDs.
* create\_item(film\_id, theater\_id, earnings): Creates a new entry with the given film ID, theater ID, and earnings.
* update\_item(film\_id, theater\_id, new\_earnings): Updates the earnings of an existing entry with the given film ID and theater ID.
* delete\_item(film\_id, theater\_id): Deletes an entry with the given film ID and theater ID.
* show\_box\_office(): Displays the current box office data.

**Execution:**

The module executes the following steps:

1. Creates instances of the Box\_office class to manage box office data.

2. Populates the box office data with sample entries using the create\_items method.

3. Initializes the BoxOfficeAnalyzer object with the box office data.

4. Enters a loop to continuously prompt the user for input.

5. Based on the user input, performs the requested operation:

* Analyze box office earnings (analysis)
* Create a new entry (create)
* Update an existing entry (update)
* Delete an entry (delete)
* Exit the program (exit)

**Error Handling:**

* Exception handling is implemented to gracefully handle errors during data manipulation and analysis operations.
* If an error occurs, an appropriate error message is displayed to the user.

**User Interaction:**

* The program provides a user-friendly command-line interface, enabling users to interact with the system easily.
* Users can input commands and parameters to perform specific actions, such as analyzing earnings or manipulating box office data.

This module serves as a comprehensive tool for managing and analyzing box office data, catering to the needs of industry professionals involved in film distribution and exhibition.

**Algorithm for Box Office Report Program:**

1. Import Modules:

* Import the required modules, including box\_office for box office data management.

2. Define BoxOfficeAnalyzer Class:

* Define a class named BoxOfficeAnalyzer to encapsulate box office data analysis and management methods.

3. Initialize Method:

* Define the \_\_init\_\_ method to initialize the Box office Analyzer object with box office data.

4. Report Box Office Earnings Method:

* Define the report\_box\_office\_earnings method to generate a detailed report on the earnings of a specific film.
* Retrieve film entries from the box office data based on the provided film ID.
* Calculate total earnings, average earnings per theater, and identify the theater with the highest earnings.
* Print the report to the console.

5. Compare Earnings Across Theaters Method:

* Define the compare\_earnings\_across\_theaters method to compare earnings across theaters based on the provided list of theater IDs.
* Retrieve theater entries from the box office data for each theater ID.
* Calculate total earnings for each theater and store the results in a dictionary.
* Print the comparison results to the console.

6. Create Item Method:

* Define the create\_item method to create a new box office entry with the provided film ID, theatre ID, and earnings.
* Call the create\_items method of the box office data object to add the new entry.
* Print a success message to the console.

7. Update Item Method:

* Define the update item method to update the earnings of an existing box office entry with the provided film ID and theatre ID.
* Convert the new earnings to a float type.
* Call the update items method of the box office data object to modify the earnings.
* Print a success message to the console.

8. Delete Item Method:

* Define the delete item method to delete a box office entry with the provided film ID and theatre ID.
* Call the delete items method of the box office data object to remove the entry.
* Print a success message to the console.

9. Show Box Office Method:

* Define the show box office method to display the current box office data.
* Retrieve all entries from the box office data.
* Print each entry's film ID, theatre ID, and earnings to the console.

10. Main Execution Block:

* Create an instance of the Box office class to manage box office data.
* Populate the box office data with sample entries using the create items method.
* Initialize a Box office Analyzer object with the box office data.
* Enter a loop to continuously prompt the user for input.
* Based on the user input, perform the requested operation: analysis, create, update, delete, or exit.
* Handle exceptions and display appropriate error messages if any operation fails.
* Print a farewell message when the user chooses to exit the program.

11. End of Algorithm

* This algorithm outlines the steps involved in the Box Office Report program, from initializing the class and methods to interacting with users and performing various operations on box office data.

**OUTPUTS :**

OUTPUT 1:

Welcome to Box Office Report!

How may I help you?

Enter your request ('analysis(1)', 'create(2)', 'update(3)', 'delete(4)', 'exit(5)'): 1

What analysis would you like to perform?

Enter 'film' for film earnings or 'total' for total earnings: film

Enter the film ID: film1

Report for Film ID: film1

Total Earnings: 11000

Average Earnings per Theater: 1833.3333333333333

Theater with Highest Earnings: theater5

OUTPUT 2:

Welcome to Box Office Report!

How may I help you?

Enter your request ('analysis(1)', 'create(2)', 'update(3)', 'delete(4)', 'exit(5)'): 2

Enter the film ID: film1

Enter the theater ID: theater1

Enter the earnings: 200

Item created successfully.

The current data of box-office is:

Film ID: film1

Theater ID: theater1

Earnings: 1000

Film ID: film2

Theater ID: theater1

Earnings: 1500

Film ID: film1

Theater ID: theater1

Earnings: 1000

Film ID: film2

Theater ID: theater1

Earnings: 1500

Film ID: film3

Theater ID: theater1

Earnings: 1200

Film ID: film4

Theater ID: theater1

Earnings: 2550

Film ID: film5

Theater ID: theater1

Earnings: 1800

Film ID: film1

Theater ID: theater2

Earnings: 2500

Film ID: film2

Theater ID: theater2

Earnings: 1800

Film ID: film3

Theater ID: theater2

Earnings: 1800

Film ID: film4

Theater ID: theater2

Earnings: 1880

Film ID: film5

Theater ID: theater2

Earnings: 1800

Film ID: film1

Theater ID: theater3

Earnings: 1500

Film ID: film2

Theater ID: theater3

Earnings: 1800

Film ID: film3

Theater ID: theater3

Earnings: 1800

Film ID: film4

Theater ID: theater3

Earnings: 1600

Film ID: film5

Theater ID: theater3

Earnings: 1807

Film ID: film1

Theater ID: theater4

Earnings: 2200

Film ID: film2

Theater ID: theater4

Earnings: 1800

Film ID: film3

Theater ID: theater4

Earnings: 1800

Film ID: film4

Theater ID: theater4

Earnings: 1850

Film ID: film5

Theater ID: theater4

Earnings: 1800

Film ID: film1

Theater ID: theater5

Earnings: 2800

Film ID: film2

Theater ID: theater5

Earnings: 1700

Film ID: film3

Theater ID: theater5

Earnings: 2800

Film ID: film4

Theater ID: theater5

Earnings: 2800

Film ID: film5

Theater ID: theater5

Earnings: 3500

Film ID: film1

Theater ID: theater1

Earnings: 200.0

OUTPUT 3:

How may I help you?

Enter your request ('analysis(1)', 'create(2)', 'update(3)', 'delete(4)', 'exit(5)'): 3

Enter the film ID: film2

Enter the theater ID: theater2

Enter the new earnings: 20

Item updated successfully.

The current data of box-office is:

Film ID: film1

Theater ID: theater1

Earnings: 1000

Film ID: film2

Theater ID: theater1

Earnings: 1500

Film ID: film1

Theater ID: theater1

Earnings: 1000

Film ID: film2

Theater ID: theater1

Earnings: 1500

Film ID: film3

Theater ID: theater1

Earnings: 1200

Film ID: film4

Theater ID: theater1

Earnings: 2550

Film ID: film5

Theater ID: theater1

Earnings: 1800

Film ID: film1

Theater ID: theater2

Earnings: 2500

Film ID: film2

Theater ID: theater2

Earnings: 20.0

Film ID: film3

Theater ID: theater2

Earnings: 1800

Film ID: film4

Theater ID: theater2

Earnings: 1880

Film ID: film5

Theater ID: theater2

Earnings: 1800

Film ID: film1

Theater ID: theater3

Earnings: 1500

Film ID: film2

Theater ID: theater3

Earnings: 1800

Film ID: film3

Theater ID: theater3

Earnings: 1800

Film ID: film4

Theater ID: theater3

Earnings: 1600

Film ID: film5

Theater ID: theater3

Earnings: 1807

Film ID: film1

Theater ID: theater4

Earnings: 2200

Film ID: film2

Theater ID: theater4

Earnings: 1800

Film ID: film3

Theater ID: theater4

Earnings: 1800

Film ID: film4

Theater ID: theater4

Earnings: 1850

Film ID: film5

Theater ID: theater4

Earnings: 1800

Film ID: film1

Theater ID: theater5

Earnings: 2800

Film ID: film2

Theater ID: theater5

Earnings: 1700

Film ID: film3

Theater ID: theater5

Earnings: 2800

Film ID: film4

Theater ID: theater5

Earnings: 2800

Film ID: film5

Theater ID: theater5

Earnings: 3500

Film ID: film1

Theater ID: theater1

Earnings: 200.0

OUTPUT 4:

How may I help you?

Enter your request ('analysis(1)', 'create(2)', 'update(3)', 'delete(4)', 'exit(5)'): 4

Enter the film ID: film1

Enter the theater ID: theater1

Item deleted successfully.

The current data of box-office is:

Film ID: film2

Theater ID: theater1

Earnings: 1500

Film ID: film2

Theater ID: theater1

Earnings: 1500

Film ID: film3

Theater ID: theater1

Earnings: 1200

Film ID: film4

Theater ID: theater1

Earnings: 2550

Film ID: film5

Theater ID: theater1

Earnings: 1800

Film ID: film1

Theater ID: theater2

Earnings: 2500

Film ID: film2

Theater ID: theater2

Earnings: 20.0

Film ID: film3

Theater ID: theater2

Earnings: 1800

Film ID: film4

Theater ID: theater2

Earnings: 1880

Film ID: film5

Theater ID: theater2

Earnings: 1800

Film ID: film1

Theater ID: theater3

Earnings: 1500

Film ID: film2

Theater ID: theater3

Earnings: 1800

Film ID: film3

Theater ID: theater3

Earnings: 1800

Film ID: film4

Theater ID: theater3

Earnings: 1600

Film ID: film5

Theater ID: theater3

Earnings: 1807

Film ID: film1

Theater ID: theater4

Earnings: 2200

Film ID: film2

Theater ID: theater4

Earnings: 1800

Film ID: film3

Theater ID: theater4

Earnings: 1800

Film ID: film4

Theater ID: theater4

Earnings: 1850

Film ID: film5

Theater ID: theater4

Earnings: 1800

Film ID: film1

Theater ID: theater5

Earnings: 2800

Film ID: film2

Theater ID: theater5

Earnings: 1700

Film ID: film3

Theater ID: theater5

Earnings: 2800

Film ID: film4

Theater ID: theater5

Earnings: 2800

Film ID: film5

Theater ID: theater5

Earnings: 3500

OUTPUT 5:

How may I help you?

Enter your request ('analysis(1)', 'create(2)', 'update(3)', 'delete(4)', 'exit(5)'): 5

Exiting...

**CONCLUSION:**

In conclusion, the Box Office Management System offers a comprehensive solution for managing and analyzing box office data, empowering users to make informed decisions and optimize their box office operations effectively. Its user-friendly interface, robust functionality, and focus on efficiency make it a valuable tool for anyone involved in the entertainment industry.

**REFERENCE:**

* <https://chat.openai.com/>
* google