📓 Project Report: Juls Walk-in Clinic Database System

To streamline operations and enhance patient care, I developed a **comprehensive and relational database system** for the **Juls Walk-in Clinic**, a healthcare facility designed to simulate real-world clinical workflows in a structured, scalable way.

🏢 Clinic Overview

The system caters to both **enrolled patients** and **walk-in visitors**. First-time patients are registered by the front-desk staff, and their records are securely stored. Enrolled patients can schedule appointments via phone or in-person, with real-time status updates such as **booked**, **cancelled**, **checked-in**, **no-show**, or **left untreated**—ensuring clarity and coordination between patients and clinic staff.

🔄 Patient Journey Workflow

Upon arrival, patients are checked in by a **secretary** and directed to the examination room by a **nurse**. Nurses record vitals including **blood pressure, weight, temperature, and symptoms**, which are immediately stored in the system for doctors to review.

Doctors conduct examinations, update diagnoses, prescribe medications, and request lab tests such as bloodwork or imaging. All this information is tracked and stored in real time, enabling accurate patient histories and better care decisions.

Lab test results, once available, are recorded by nurses into the system. If required, the system helps manage **follow-up appointments**, ensuring continuity of care.

👩‍⚕️ Employee & Shift Management

The clinic runs in **two daily shifts**, and the system manages shift schedules for non-doctor staff such as nurses, secretaries, and administrative employees. Salaries are calculated based on predefined hourly wages, and the database can generate **bi-weekly payroll summaries**, simplifying HR operations.

🔍 Key Capabilities & Features

- **Appointment tracking** with detailed status updates
- **Vitals and visit logs** per patient for health trend analysis
- **Lab test management** and result recording
- **Doctor-specific scheduling and patient assignment**

- **Employee records** with hourly wage-based salary calculations
- **Role-based structuring** for nurses, secretaries, and admin staff
- **Payment tracking** including amount, status, and method
- **Advanced SQL queries** for updates, data cleaning, views, procedures, and analytics
- **Auto-incremented primary keys** ensure unique, consistent entries
- **EER Diagram** visualizes all relationships between entities and the overall database flow

💡 Conclusion

This project allowed me to apply advanced **database design principles**, normalization techniques, and complex **SQL functionalities** like joins, views, subqueries, and procedures. It demonstrates a **real-world use case** in healthcare data systems and is designed to be **extensible**—with the potential for future integration of features like appointment reminders, digital prescriptions, and predictive analytics.

All project assets, including the schema, queries, and sample data, are available on **GitHub**: click *here*.