**Database Management System Case Studies**

# Case Study 1

**Hotel Management System**

**Aim:** XYZ is a multiservice luxury hotel which provides various services to its customers from picking up form location to logging facilities etc. Different types of customer come to hotel who requires different types of rooms and services. Also, hotel is having huge staff to manage and provide easy and fast services to customers. Hotel has several other positions for better management.

The aim of this case study is to design and develop a database for the hotel to maintain records of hotel staff and customers. Also keep record of availability of the rooms and maintenance of other services provided by hotel.

**Description(SRS)** In hospital, there are many departments like Orthopedic, Pathology, Emergency, Dental, Gynecology, Anesthetics, I.C.U., Blood Bank, Operation Theater, Laboratory, M.R.I., Neurology, Cardiology, Cancer Department, Corpse, etc. There is an OPD where patients come and get a card (that is, entry card of the patient) for check up from the concerned doctor. After making entry in the card, they go to the concerned doctor’s room and the doctor checks up their ailments. According to the ailments, the doctor either prescribes medicine or admits the patient in the concerned department. The patient may choose either private or general room according to his/her need. But before getting admission in the hospital, the patient has to fulfill certain formalities of the hospital like room charges, etc. After the treatment is completed, the doctor discharges the patient. Before discharging from the hospital, the patient again has to complete certain formalities of the hospital like balance charges, test charges, operation charges (if any), blood charges, doctors’ charges, etc.

Next we talk about the doctors of the hospital. There are two types of the doctors in the hospital, namely, *regular doctors* and *call on doctors*. Regular doctors are those doctors who come to the hospital daily. Calls on doctors are those doctors who are called by the hospital if the concerned doctor is not available.

## Table Description:

Following are the tables along with constraints used in *Hospital Management* database.

**DEPARTMENT:** This table consists of details about the various departments in the hospital. The information stored in this table includes department name, department location, and facilities available in that department.

***Constraint***: Department name will be unique for each department.

1. **ALL\_DOCTORS:** This table stores information about all the doctors working for the hospital and the departments they are associated with. Each doctor is given an identity number starting with DR or DC prefixes only.

***Constraint***: Identity number is unique for each doctor and the corresponding department should exist in DEPARTMENT table.

1. **DOC\_REG:** This table stores details of regular doctors working in the hospital. Doctors are referred to by their doctor number. This table also stores personal details of doctors like name, qualification, address, phone number, salary, date of joining, etc.

***Constraint***: Doctor’s number entered should contain DR only as a prefix and must exist in ALL\_DOCTORS table.

1. **DOC\_ON\_CALL:** This table stores details of doctors called by hospital when additional doctors are required. Doctors are referred to by their doctor number. Other personal details like name, qualification, fees per call, payment due, address, phone number, etc., are also stored.

***Constraint***: Doctor’s number entered should contain DC only as a prefix and must exist in ALL\_DOCTORS table.

1. **PAT\_ENTRY:** The record in this table is created when any patient arrives in the hospital for a check up. When patient arrives, a patient number is generated which acts as a primary key. Other details like name, age, sex, address, city, phone number, entry date, name of the doctor referred to, diagnosis, and department name are also stored. After storing the necessary details patient is sent to the doctor for check up.

***Constraint***: Patient number should begin with prefix PT. Sex should be *M* or *F* only. Doctor’s name and department referred must exist.

1. **PAT\_CHKUP:** This table stores the details about the patients who get treatment from the doctor referred to. Details like patient number from patient entry table, doctor number, date of check up, diagnosis, and treatment are stored. One more field status is used to indicate whether patient is admitted, referred for operation or is a regular patient to the hospital. If patient is admitted, further details are stored in PAT\_ADMITtable. If patient is referred for operation, the further details are stored in PAT\_OPR table and if patient is a regular patient to the hospital, the further details are stored in PAT\_REG table.

***Constraint***: Patient number should exist in PAT\_ENTRY table and it should be unique.

1. **PAT\_ADMIT:** When patient is admitted, his/her related details are stored in this table. Information stored includes patient number, advance payment, mode of payment, room number, department, date of admission, initial condition, diagnosis, treatment, number of the doctor under whom treatment is done, attendant name, etc.

***Constraint***: Patient number should exist in PAT\_ENTRY table. Department, doctor number, room number must be valid.

1. **PAT\_DIS:** An entry is made in this table whenever a patient gets discharged from the hospital. Each entry includes details like patient number, treatment given, treatment advice, payment made, mode of payment, date of discharge, etc.

***Constraint***: Patient number should exist in PAT\_ENTRY table.

1. **PAT\_REG:** Details of regular patients are stored in this table. Information stored includes date of visit, diagnosis, treatment, medicine recommended, status of treatment, etc.

***Constraint***: Patient number should exist in patient entry table. There can be multiple entries of one patient as patient might be visiting hospital repeatedly for check up and there will be entry for patient’s each visit.

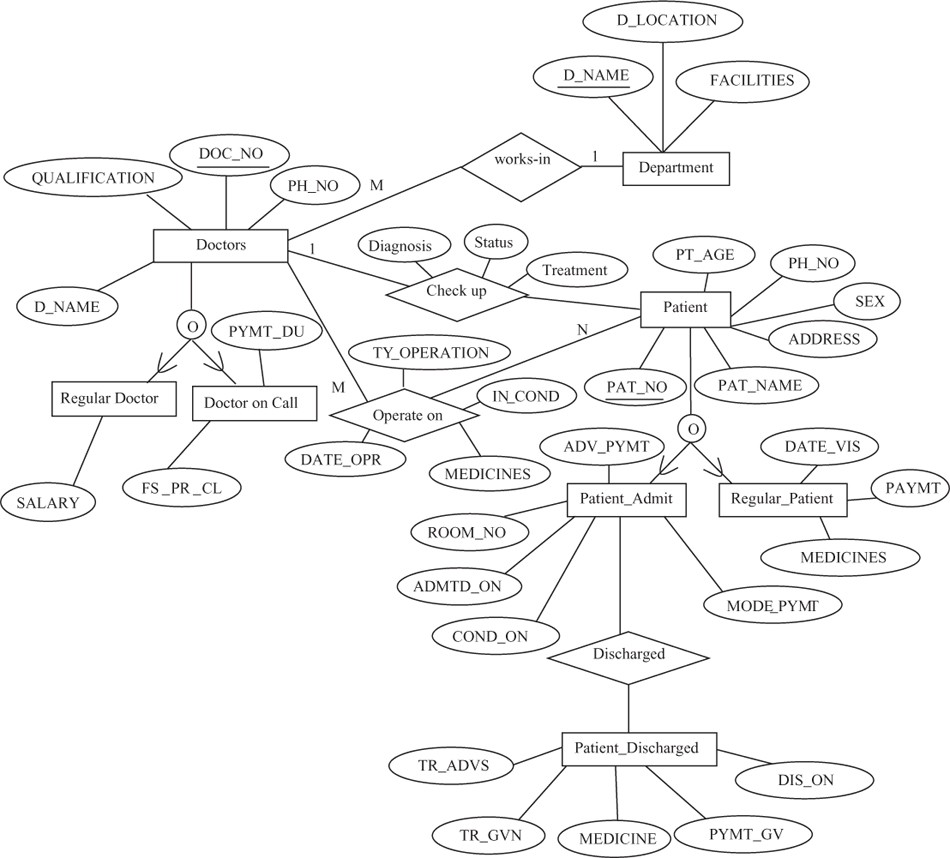
1. **PAT\_OPR:** If patient is operated in the hospital, his/her details are stored in this table. Information stored includes patient number, date of admission, date of operation, number of the doctor who conducted the operation, number of the operation theater in which operation was carried out, type of operation, patient’s condition before and after operation, treatment advice, etc.

***Constraint***: Patient number should exist in PAT\_ENTRY table. Department, doctor number should exist or should be valid.

1. **ROOM\_DETAILS:** It contains details of all rooms in the hospital. The details stored in this table include room number, room type (general or private), status (whether occupied or not), if occupied, then patient number, patient name, charges per day, etc.

***Constraint***: Room number should be unique. Room type can only be *G* or *P* and status can only be *Y* or *N*

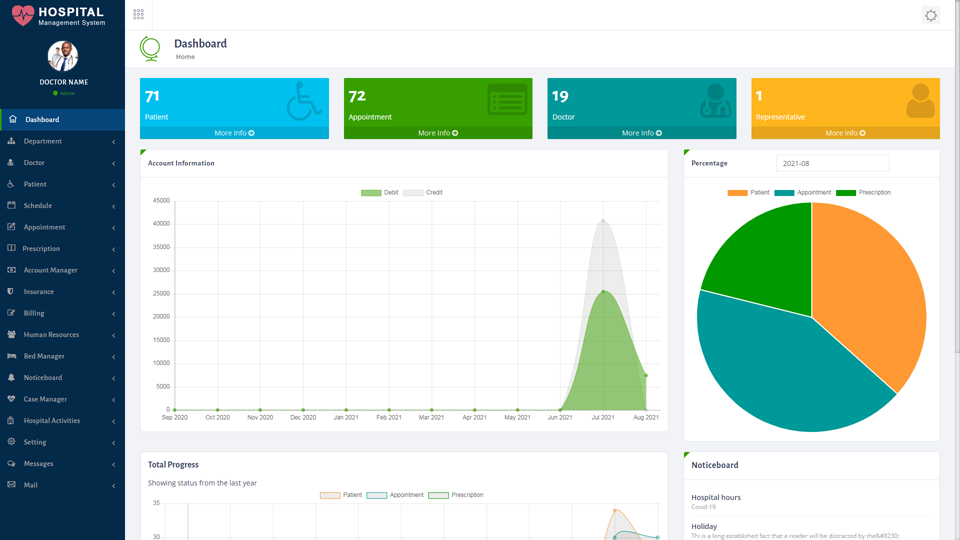
## E‐R Diagram(Conceptual Design using ER features)

****

**Relational Database Schema for Case Study**

The relational database schema for *Hospital Management* database is as follows:

1. DEPARTMENT (D\_NAME, D\_LOCATION, FACILITIES)
2. ALL\_DOCTORS (DOC\_NO, DEPARTMENT)
3. DOC\_REG(DOC\_NO, D\_NAME, QUALIFICATION, SALARY, EN\_TIME, EX\_TIME, ADDRESS, PH\_NO, DOJ)
4. DOC\_ON\_CALL (DOC\_NO, D\_NAME, QUALIFICATION, FS\_PR\_CL, PYMT\_DU, ADDRESS, PH\_NO)
5. PAT\_ENTRY (PAT\_NO, PAT\_NAME, CHKUP\_DT, PT\_AGE, SEX, RFRG\_CSTNT, DIAGNOSIS, RFD, ADDRESS, CITY, PH\_NO, DEPARTMENT)
6. PAT\_CHKUP (PAT\_NO, DOC\_NO, DIAGNOSIS, STATUS, TREATMENT)
7. PAT\_ADMIT (PAT\_NO, ADV\_PYMT, MODE\_PYMT, ROOM\_NO, DEPTNAME, ADMTD\_ON, COND\_ON, INVSTGTN\_DN, TRMT\_SDT, ATTDNT\_NM)
8. PAT\_DIS (PAT\_NO, TR\_ADVS, TR\_GVN, MEDICINES, PYMT\_GV, DIS\_ON)
9. PAT\_REG (PAT\_NO, DATE\_VIS, CONDITION, TREATMENT, MEDICINES, DOC\_NO, PAYMT)
10. PAT\_OPR (PAT\_NO, DATE\_OPR, IN\_COND, AFOP\_COND, TY\_OPERATION, MEDICINES, DOC\_NO, OPTH\_NO, OTHER\_SUG)
11. ROOM\_DETAILS (ROOM\_NO, TYPE, STATUS, RM\_DL\_CRG, OTHER\_CRG)



**Graphical User Interface, Source Code**

### Hospital Management System Test cases

* Check by entering the correct URL in the browser, and the application should be loading properly
* Check is there any user verification functionality present on the application.
* Check by entering valid credential like user name and password user should be able to log in
* Check by entering invalid credentials the user should not be login into the application, and an error message should be displayed
* Check if the hospital management system application has an option to add a new patient
* Check whether all the mandatory fields are present registration portal
* Check after adding a new patient, and after completion of the payment process, the patient cards should be printed
* Check whether the patient card has the details like assign doctor name comma department, present application number command date of join and also located bed details, etc
* Check after completion of patient checkup process the details should be updated in the patient database
* Check if the patient exists in the [database](https://www.oracle.com/in/database/what-is-database/), and he performs some checkup then the user should be able to search the details of the present in the database
* Check if the doctors are also able to update the passenger details after check
* Check the number of roles in the hospital management system like the patient, doctor, admin, accountant, etc
* Check that the authorized users can see the doctor details in the portal like the doctors’ timings and fees.
* Check if there is any functionality to add a new doctor in the hospital management system, for instance, we have added patient details in the database
* Check whether the admin users can delete doctor and patient information by the hospital management system portal
* Check whether an accountant user type can calculate the bills for patients by collecting data from different systems.
* Check after the formation of a bill that should be an option to print the bill or to generate a hard copy of the bill.
* Check the authorized users have the privilege to check the details report of the patients like day wise
* Check the admin has all the access

**Conclusion**

Implementation of hospital management system project helps **to store all the kinds of records**, provide coordination and user communication, implement policies, improve day- to-day operations, arrange the supply chain, manage financial and human resources, and market hospital services