

Databook Part 1 - HOTEL MANAGEMENT NARRATIVE

- Database Description

Hotel management contains plenty of data and information to manage the day to day activities efficiently. Databases play a crucial role in hotel management, such as managing operations like room bookings, check-in, and check-outs for the guests, handling room services, various payments, etc. Hence, there is a need for a well-curated database that can help streamline all the activities. The data must be stored, organized, and accessed for hotel staff to manage daily operations efficiently. The main components include reservations and bookings, guest profile, inventory management, staff management, and finance reports.

Reservations and Bookings - The database for a hotel needs to store information related to room availability, the guests bookings and all the details related to the reservations. This is necessary to allocate rooms, and track and monitor the bookings. Maintaining this, also allows guests to make their bookings via mobile apps like MakeMytrip, goibibo, etc.

Guest Profile - The management has to store the guest's profile that includes their personal information like name, address, history, identification etc. The housekeeping staff on other hand can use the database to collect and store guest preferences, which can make them feel valued. This leads to effective and accurate services to the guests.

Inventory Management - Databases are necessary for efficient inventory management. The food and beverage department needs to have a track about the inventory levels. The system can help to optimize the inventory levels, which will lead to minimum waste, reduced costs and avoid delays.

Staff Management - Internally, the staff management needs a database to manage the staff schedules, their attendance and their performances. They can maintain the number of leaves, and their payrolls accordingly.

Finance Reports - Databases are essential for accurate financial reporting. Say, the management team can make use of reports to track the revenue generated, other financial metrics like rates, and this can help them make informed business decisions.

In this project, we will be considering a mid-sized, multi floored hotel with rooms and suites. The leisure include, restaurant, bar, fitness center and swimming pool. The hotel management database will be accessible and used by the stakeholders, hotel manager, staff and management.

The users of our hotel management system-

- The Hotel Staff: The hotel staff includes receptionists, housekeeping staff, restaurant staff, and maintenance staff. These staff members will use the database system to manage various hotel operations, such as room bookings, guest check-ins, guest checkouts, room service requests, and billing.
- Guests: The guests are the primary users of the hotel management system. They will use the system to make room reservations, check-in, request room service, and check-out.
- Management: The hotel management will use the database system to generate reports, monitor hotel operations, and make data-driven decisions.

- Database Design - Our hotel management database system will consist of the following entities.

1. ROOM_TYPE - This table stores the types of room available, mainly deluxe and standard. The table includes name, description, maximum occupancy and rate.
2. ROOM - This table will store information about each room, including room number, room type, room price, room availability, room description.
3. GUEST - This table will store information about each guest, including name, identification, address, phone number.
4. RESERVATION - This table will store information about each room reservation, including the guest's ID, room number, check-in date, and check-out date.
5. RESTAURANT - This table assumes that there are more than 1 restaurant in the hotel. It stores restaurant id, restaurant name, location, phone number, hours of operation.
6. MENU - This will store the menu id, menu name, menu description, restaurant id
7. AMENITY - This stores the id for various amenities offered, amenity names and description.
8. HOTEL_SERVICE - This table will store information about each room service request, including the guest's name, room number, service type, and service status.
9. STAFF - This stores the staff id, staff name, their address, phone number, department id.
10. SHIFT - This includes, shift ID, staff ID, start time, end time.
11. MANAGER - This is the manager table; this is a child table from the parent table STAFF. This table contains all the personal information about the manager.
12. DEPARTEMENT - This table includes the various departments of the hotel, their IDs, the manager.
13. PAYMENT - This table stores details about each guest payment during the stay, payment method, etc.
14. INVOICE - The table stores invoice for each guest stay, this contains breakdown of charges and taxes. Invoice date, due date, amount are included.

- Data dictionary

Entity	Attribute name	Contents	Data Type	Format	Range	Required	PK/FK	Reference
ROOM TYPE	ROOMTYPE_ID	Room type id	INT(3)	999	100-999	Y	PK	
	ROOM_NAME	Name for the room	CHAR(50)	Xxxxx		Y		
	ROOM_DESC	Description of the rooms	VARCHAR(50)	Xxxxx		Y		
	ROOM_OCCUPANCY	Number of people allowed	INT(2)	99	0-10	Y		
	ROOMTYPE_RATE_PER_NIGHT	Cost per room type	DECIMAL(5,2)	999.99	100.00-999.99	Y		
ROOM	ROOM_ID	Room ID	INT(5)	9999	1000-9999	Y	PK	
	ROOM_NUM	Room number	INT(2)	99	10-99	Y		
	ROOMTYPE_ID	Room type id	INT(3)	999	100-999	Y	FK	ROOM TYPE
	ROOM_AVA	Availability of room	BOOL	0/1	0-1	Y		
	ROOM_FLOOR	Floor of room	INT(1)	9	0-9	Y		
	ROOM_ISSMOKING	If smoking is permitted or not	BOOL	0/1	0-1	Y		
ROOM_AMENITY	ROOM_ID	Room ID	INT(5)	99999	10000-99999	Y	PK, FK	ROOM
	AMENTITY_ID	Amenity id	INT(4)	9999	1000-9999	Y	PK, FK	AMENITY
GUEST	GUEST_ID	Guest id	INT(4)	9999	1000-9999	Y	PK	
	GUEST_LNAME	Last name	VARCHAR(20)	Xxxxx		Y		
	GUEST_FNAME	First name	VARCHAR(20)	Xxxxx		Y		
	GUEST_PID	Personal id like a passport	VARCHAR(20)	Xxxxx		Y		
	GUEST_CITY	City	VARCHAR(20)	Xxxxx		Y		
	GUEST_COUNTRY	Country	VARCHAR(20)	Xxxxx		Y		
	GUEST_PHONE_NO	Phone num	CHAR(12)	999-999-9999		Y		
	ROOM_ID	Room ID	INT(5)	99999	10000-99999	Y	FK	ROOM
	REST_ID	Restaurant id	INT(3)	999	100-999	Y	FK	
RESERVATION	RESERV_ID	Reservation id	INT(5)	99999	10000-9999	Y	PK	
	GUEST_ID	Guest id	INT(4)	9999	1000-9999	Y	FK	GUEST
	ROOM_ID	Room ID	INT(5)	99999	10000-99999	Y	FK	ROOM
	CHECK_IN_DATE	Checked in	DATE	yyyy-m m-dd		Y		
	CHECK_IN_OUT	Checked out	DATE	yyyy-m m-dd		Y		
	NO_OF_DAYS	Number of days stayed	INT(2)	99	1-99	Y		
	TOTAL_PRICE	Total final bill	DECIMAL(5,2)	999.99	0.01-999.9	Y		

RESTAURANT	REST_ID	Restaurant id	INT(3)	999	100-999	Y	PK	
	REST_NAME	Name	VARCHAR(30)	Xxxx		Y		
	REST_LOC	Location	VARCHAR(50)	Xxxx		Y		
	REST_PHONE_NO	Phone num	CHAR(12)	999-99 9-9999		Y		
	REST_EMAIL	Email id	VARCHAR(50)	Xxx@xx x		Y		
	REST_WEBSITE	Website link	VARCHAR(50)	Xxx.xxx		Y		
MENU	MENU_ID	Menu id	INT(5)	99999	10-99999	Y	PK	
	ITEM_NAME	Name	VARCHAR(30)	Xxxx		Y		
	ITEM_DESC	Description	VARCHAR(100)	Xxxx		Y		
	ITEM_AMT	Amount	DECIMAL(5,2)	999.99	100.00-999.99	Y		
	ITEM_PORTION	Quantity	VARCHAR(10)	Xxxx		Y		
	REST_ID	Restaurant id	INT(3)	999	100-999	Y	FK	RESTAURANT
AMENITY	AMENTITY_ID	Amenity id	INT(4)	9999	1000-9999	Y	PK	
	AMENTITY_NAME	Name	VARCHAR(30)	Xxxx		Y		
	AMENTITY_DESC	Description	VARCHAR(100)	Xxxx		Y		
	AMENITY_AMT	Amount	DECIMAL(5,2)	999.99	100.00-999.99	Y		
	AMENITY_STATUS	Status ID	BOOL					
HOTEL_SERVICE	SERVICE_ID	Service id	INT(3)	999	100-999	Y	PK	
	RESERV_ID	Reservation id	INT(5)	99999	10000-99999	Y	FK	RESERVATION
	SERVICE_TYPE	Type	VARCHAR(20)	Xxxx		Y		
	SERVICE_DESC	Description	VARCHAR(100)	Xxxx		Y		
	SERVICE_PRICE	Cost	DECIMAL(5,2)	999.99	100.00-999.99	Y		
DEPARTMENT	DEPT_ID	Department id	INT(3)	999	100-999	Y	PK	
	DEPT_PHONE	Phone number	CHAR(12)	999-99 9-9999		Y		
	DEPT_NAME	Department name	VARCHAR(20)	Xxxx		Y		
	DEPT_DESC	Dept description	VARCHAR(50)	Xxxx		Y		
	DEPT_LOC	Location	VARCHAR(50)	Xxxx		Y		
STAFF	STAFF_ID	Staff id	INT(3)	999	100-999	Y	PK	
	STAFF_LNAME	Last name	VARCHAR(20)	Xxxx		Y		
	STAFF_FNAME	First name	VARCHAR(20)	Xxxx		Y		
	STAFF_POSITION	Position	VARCHAR(20)	Xxxx		Y		
	STAFF_EMAIL_ADD	Email address	VARCHAR(50)	Xxx@xx x		Y		
	STAFF_PHONE_NO	Phone num	CHAR(12)	999-99 9-9999		Y		
	DEPT_ID	Department id	INT(3)	999	100-999	Y	FK	DEPARTMENT
	REST_ID	Restaurant id	INT(3)	999	100-999	Y	FK	RESTAURANT

STAFF_SHIFT	STAFF_ID	Staff id	INT(3)	999	100-999	Y	PK, FK	STAFF
	SHIFT_ID	Shift Id	INT(2)	99	10-99	Y	PK, FK	SHIFT
SHIFT	SHIFT_ID	Shift id	INT(2)	99	10-99	Y	PK	
	START_TIME	Shift start time	VARCHAR(10)	HH:MM:SS		Y		
	END_TIME	Shift end time	VARCHAR(10)	HH:MM:SS		Y		
	SHIFT_TYPE	Morning, noon, night	VARCHAR(10)	Xxxxx		Y		
	NO_OF_DAYS	Number of working days	INT(1)	0-9	9	Y		
MANAGER	STAFF_ID	Staff id	INT(3)	999	100-999	Y	PK, FK	
(This is a child table of staff)	TITLE	The title of the manager	VARCHAR(20)			Y		
	HIRE_DATE	Date of hire	DATE			Y		
	PERFORMANCE_RATING	Ratings received	FLOAT	9.9	0.0-9.9	Y		
	GENDER	Gender	VARCHAR(10)			Y		
INVOICE	INVOICE_ID	Invoice id	INT(5)	99999	10000-99999	Y	PK	
	INVOICE_DATE	Date of generation	DATE	yyyy-m m-dd		Y		
	TAX_AMT	Tax charges	DECIMAL(4,2)	99.99	10.00-99.99	Y		
	ADD_CHARGES	Any additional amt	DECIMAL(4,2)	99.99	10.00-99.99	Y		
	INVOICE_AMT	Final total	DECIMAL(5,2)	999.99	100.00-999.99	Y		
	PAY_DUE_DATE	The due date	DATE	yyyy-m m-dd		Y		
	PAY_STATUS	Paid/overdue	VARCHAR(10)	Xxxxx		Y		
	RESERV_ID	Reservation id	INT(5)	99999	10000-99999	Y	FK	RESERVATION
PAYMENT	PAYMENT_ID	Payment id	INT(2)	99	0-99	Y	PK	
	INVOICE_ID	Invoice id	INT(5)	99999	10000-99999	Y	FK	INVOICE
	PAYMENT_METHOD	cash/card/online	VARCHAR(10)	Xxxxx		Y		
	PAYEMENT_DATE	Date of payment	DATE	yyyy-m m-dd		Y		
	PAYEMENT_AMT	Total amt	DECIMAL(5,2)	999.99	100.00-999.99	Y		
	PAYEMENT_STATUSES	Success, refund, declined	VARCHAR(10)	Xxxxx		Y		
RESERV_HOTEL	RESERV_ID	Reservation id	INT(5)	99999	10000-99999	Y		RESERVATION
	SERVICE_ID	Service id	INT(3)	999	100-999	Y		HOTEL_SERVICE

- Business rules

1. ROOM_TYPE / ROOM

- A ROOM TYPE can have multiple rooms, but a room can belong to only one Room type.
- A room must belong to a valid ROOM TYPE.
- The ROOM RATE for a ROOM must be the same as the ROOM TYPE it belongs to.

2. ROOM / RESERVATION

- A ROOM can have zero or multiple Reservations, but a Reservation can only be for one ROOM.
- A RESERVATION can only be for an available ROOM.

3. ROOM / GUEST

- A ROOM can have multiple GUESTS, but zero or one GUEST can only stay in one ROOM at a time.
- A GUEST must be assigned to a valid ROOM

4. RESERVATION / INVOICE

- A RESERVATION can have one INVOICE and a INVOICE can only be for one RESERVATION
- A INVOICE must be made for a valid RESERVATION.

5. RESTAURANT / MENU

- A RESTAURANT can have one MENU, and a MENU can only belong to one RESTAURANT (Since all the restaurants are unique)
- A MENU must belong to a valid RESTAURANT

6. ROOM / AMENITY

- A ROOM can have zero or multiple AMENITY, and an AMENITY can be available in multiple ROOM
- We have an ROOM_AMENITY table that is a bridge between these two tables.
- The amenities are not included in the room rate and will be charged separately in the INVOICE.

7. STAFF / DEPARTMENT

- Each staff member can work in only one department but many staff members can work in one department.
- A department in the DEPARTMENT table must have at least one associated staff member in the STAFF table.

8. STAFF / SHIFT

- Staff members can have many shifts assigned to them and a shift can be assigned to multiple staff members.
- The STAFF_SHIFT is a bridge table between these two tables as they have many to many relations.

9. STAFF / MANAGER

- Each staff can have only one manager, and each manager can be a part of only a group of staff.

10. INVOICE / PAYMENT

- An INVOICE can have one payment, and each payment can be only for one invoice.
- The total amount of all PAYMENTS made for an INVOICE must be equal to the total amount on the INVOICE.

11. HOTEL_SERVICE / RESERVATION

- A RESERVATION can have zero or multiple HOTEL_SERVICE , and an RESERVATION can be available in multiple HOTEL_SERVICE
- We have a RESERV_HOTEL table that is a bridge between these two tables.

12. RESTAURANT / STAFF

- One RESTAURANT should have one or many staff.
- We need to have at least one staff member present.

13. RESTAURANT / GUEST

- One RESTAURANT must have guests.
- There can be 0 or many guests at a given time.

- Entity relationship model (ERM) components

ENTITY	RELATIONSHIP	CONNECTIVITY	ENTITY
ROOM_TYPE	..has..	1:M	ROOM
ROOM	..has...	0:M	RESERVATION
ROOM	...has....	M:N	AMENITY
ROOM	..can have...	0:M	GUEST
RESERVATIONmust have...	1:1	INVOICE
RESTAURANT	...must have...	1:1	MENU
DEPARTMENT	..has...	1:M	STAFF
STAFF	...has ...	M:N	SHIFT
STAFF	...may have...	1:M	MANAGER
INVOICE	...will have....	1:1	PAYMENT
RESERVATION	..may have....	M:N	HOTEL_SERVICE
RESTAURANT	...must have...	1:M	STAFF
RESTAURANT	...must have...	0:M	GUEST

Note -

- STAFF_SHIFT is a bridge table between the many to many relation, STAFF and SHIFT table.
- ROOM_AMENITY, is a bridge table between ROOM and AMENITY table.
- RESERV_HOTEL, is a bridge table between RESERVATION and HOTEL table.
- MANAGER Table is a child table of the parent table STAFF.

- WHY DO WE NEED THIS: INTENDED USE -

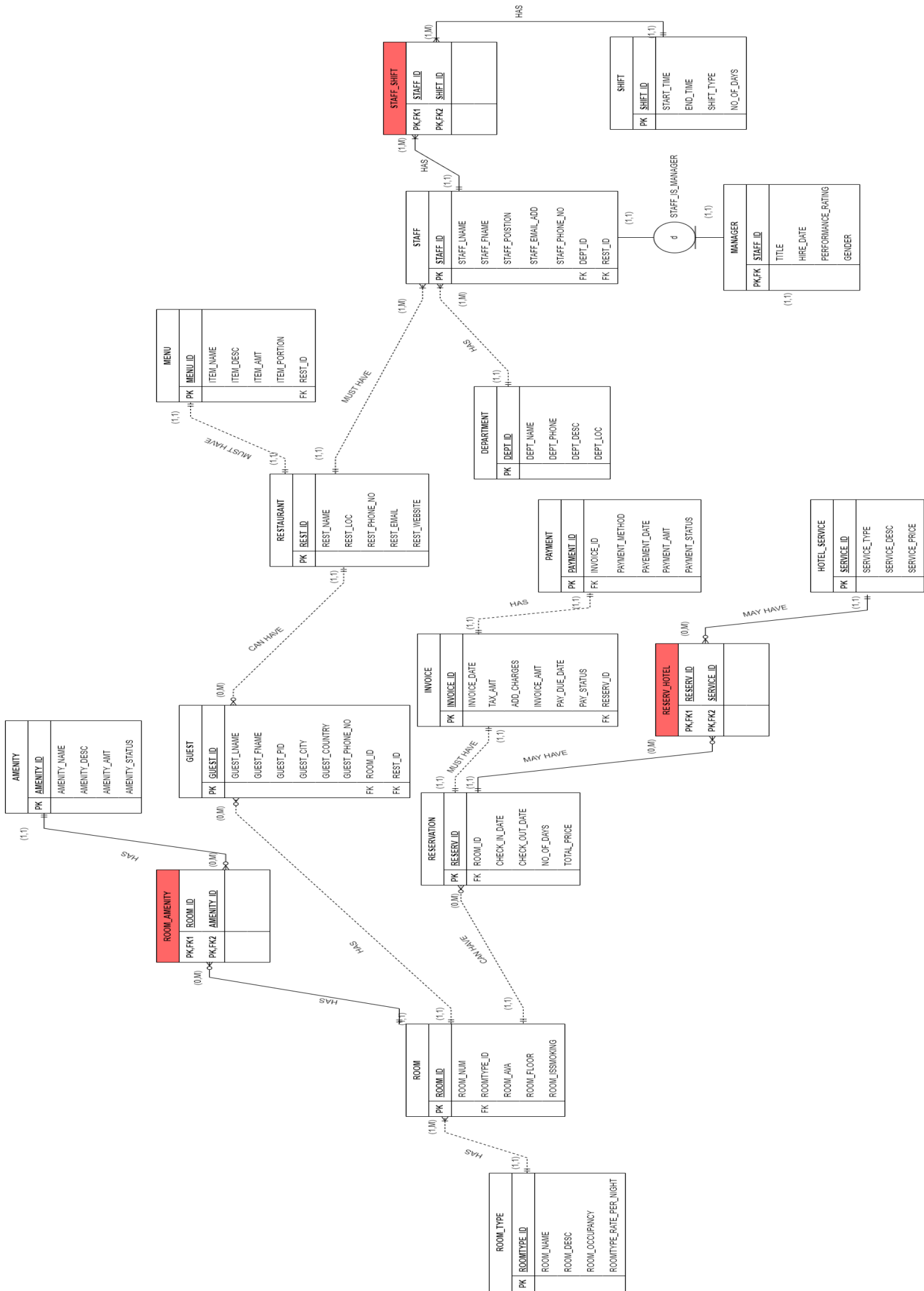
- Our Hotel Database is useful for multiple important factors, to begin with the most important being managing hotel operations. The hotel database will be used to manage various operations like reservations, room allotting, the staff shifts scheduling, maintenance, and many more.
- The database is very useful to store and track guest information. The name, address, payment information, etc helps the hotel management to take good care of the guests and provide them with great discounts in the future.
- One of the most beneficial advantages of this is analyzing the hotel's performance. The database will help us track the occupancy rate, the generated revenue, and various performance metrics that will lead management to make informed decisions. Reports related to revenue and guest stratification can thus be made. This will also help us make improvements and grow the business.

Data Book: Part 2, The Entity Relationship Diagram

Relational schemas

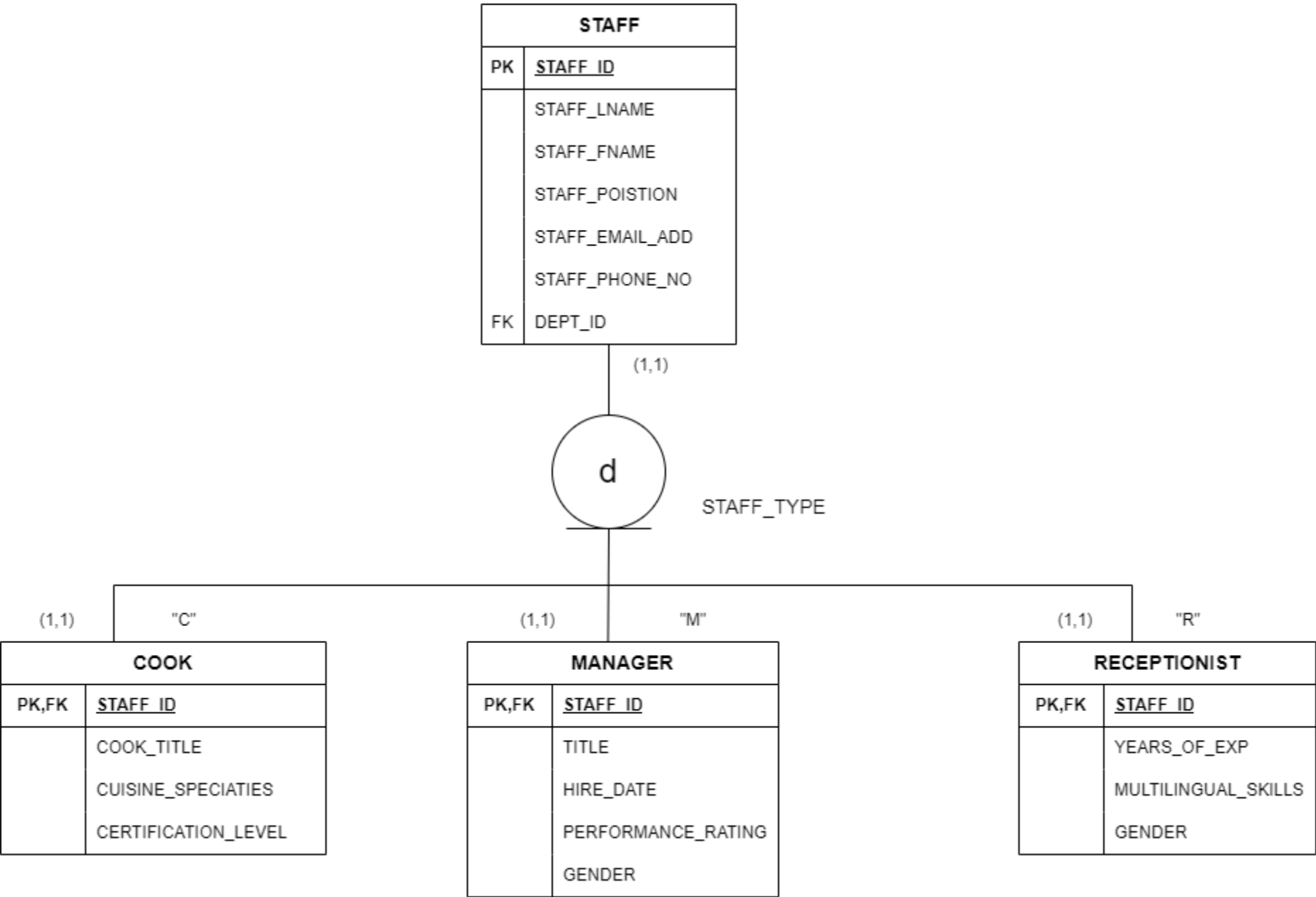
1. ROOM_TYPE(**ROOMTYPE_ID**, ROOM_NAME, ROOM_DESC, ROOM_OCCUPANCY, ROOMTYPE_RATE_PER_NIGHT)
2. ROOM(**ROOM_ID**, ROOM_NUM, ROOMTYPE_ID, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING)
3. RESERVATION(**RESERV_ID**, ROOM_ID, CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE)
4. RESTAURANT(**REST_ID**, REST_NAME, REST_LOC, REST_PHONE_NO, REST_EMAIL, REST_WEBISTE)
5. MENU(**MENU_ID**, ITEM_NAME, ITEM_DESC, ITEM_AMT, ITEM_PORTION, REST_ID)
6. GUEST(**GUEST_ID**, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO, ROOM_ID, REST_ID)
7. AMENITY(**AMENITY_ID**, AMENITY_NAME, AMENITY_DESC, AMENITY_AMT, AMENITY_STATUS)
8. HOTEL_SERVICE(**SERVICE_ID**, SERVICE_TYPE, SERVICE_DESC, SERVICE_PRICE)
9. DEPARTMENT(**DEPT_ID**, DEPT_PHONE, DEPT_NAME, DEPT_DESC, DEPT_LOC)
10. STAFF(**STAFF_ID**, STAFF_LNAME, STAFF_FNAME, STAFF_POSITION, STAFF_EMAIL_ADD, STAFF_PHONE_NO, DEPT_ID, REST_ID)
11. SHIFT(**SHIFT_ID**, START_TIME, END_TIME, SHIFT_TYPE, NO_OF_DAYS)
12. MANAGER(**STAFF_ID**, TITLE, HIRE_DATE, PERFORMANCE_RATING, GENDER)
13. INVOICE(**INVOICE_ID**, INVOICE_DATE, TAX_AMT, ADD_CHARGES, INVOICE_AMT, PAY_DUE_DATE, PAY_STATUS, RESERV_ID)
14. PAYMENT(**PAYMENT_ID**, INVOICE_ID, PAYMENT_METHOD, PAYMENT_DATE, PAYMENT_AMT, PAYMENT_STATUS)
15. ROOM_AMENITY(**ROOM_ID**, **AMENITY_ID**)
16. STAFF_SHIFT(**STAFF_ID**, **SHIFT_ID**)
17. RESERV_HOTEL(**RESERV_ID**, **SERVICE_ID**)

Using Crow's Foot notation, build a diagram of all tables in Microsoft Visio or draw.io.



Using Crow’s Foot notation, build a subtype/supertype diagram among one set of related tables in Microsoft Visio or draw.io.

Here, we take the example of STAFF (parent/supertype) with MANAGER, COOK, and RECEPTIONIST (child/subtype)



A documented walk-through of one normalized table

The denormalized table here is as follows -

GUEST RESERVATION TABLE

<u>RESERV_ID</u>	CHECK_IN_DATE	CHECK_OUT_DATE	NO_OF_DAYS	TOTAL_PRICE	<u>GUEST_ID</u>	GUEST_LNAME	GUEST_FNAME	GUEST_PID	GUEST_CITY	GUEST_COUNTRY	GUEST_PHONE_NO	ROOM_NUM	ROOM_AVA	ROOM_FLOOR	ROOM_ISSMOKING	ROOM_RATE
------------------	---------------	----------------	------------	-------------	-----------------	-------------	-------------	-----------	------------	---------------	----------------	----------	----------	------------	----------------	-----------

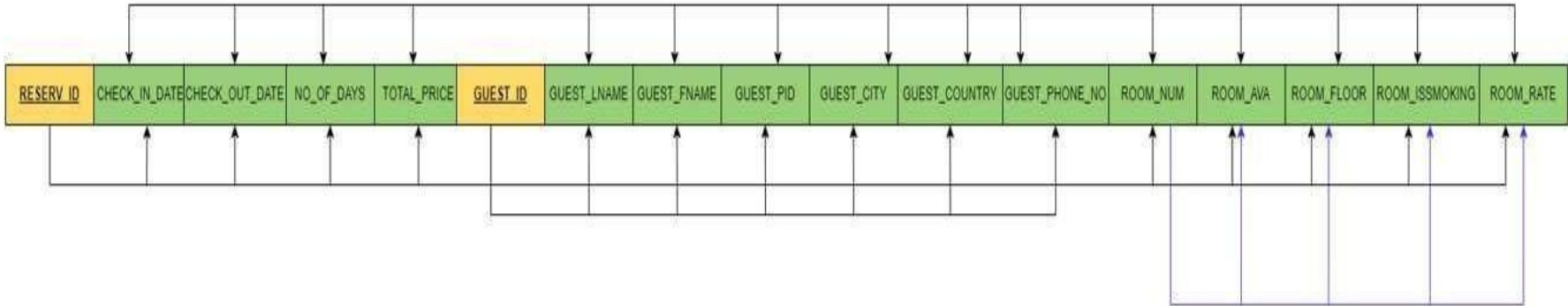
RESERVATION(RESERV_ID,CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE, GUEST_ID, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO, ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)

RESERVATION (RESERV_ID, CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE, GUEST_ID, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO, ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)

1NF

- In the following diagram, we identify the partial dependency and transitive dependencies.
- (RESERV_ID, CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE, ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)
- (GUEST_ID, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO)
- And a transitive dependency -
- (ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)

1NF

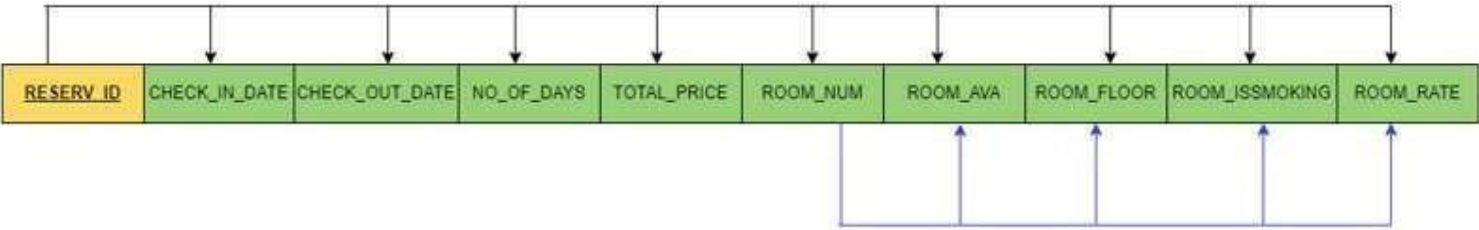


RESERVATION (RESERV_ID, CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE, GUEST_ID, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO, ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)

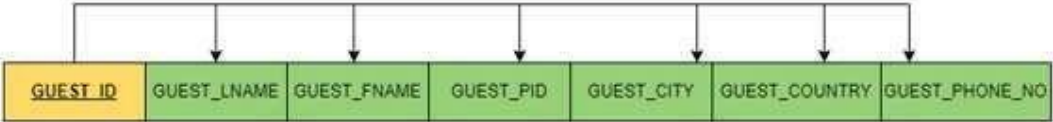
2NF

- In the 2NF form, we will be removing the partial dependency by creating a new table for GUEST.
- RESERVATION (RESERV_ID, CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE, ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)
- GUEST (GUEST_ID, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO

2NF



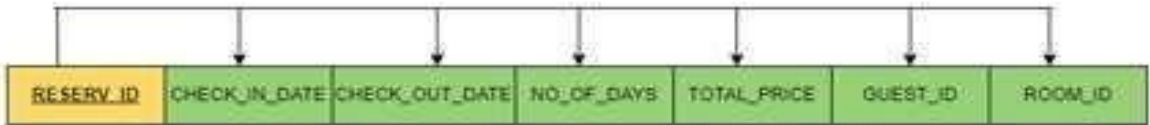
RESERVATION (RESERV_ID, CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE, ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)



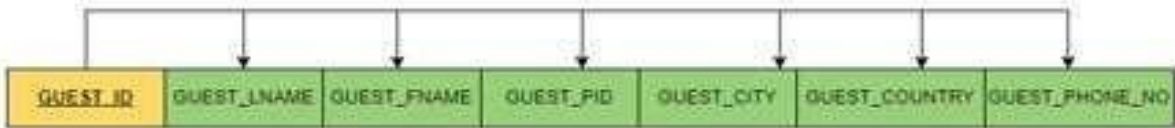
GUEST (GUEST_ID, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO

3NF

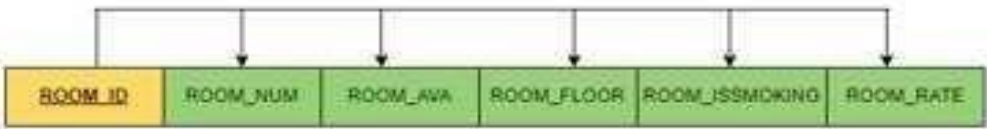
- In 3NF, we will be removing the transitive dependency, by creating a new table for ROOM with the primary key as ROOM_ID and GUEST_ID, and ROOM_ID as the foreign keys.
- RESERVATION (RESERV_ID, CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE, GUEST_ID, ROOM_ID)
- GUEST(GUEST_ID, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO
- ROOM (ROOM_ID, ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)



RESERVATION (RESERV_ID, CHECK_IN_DATE, CHECK_OUT_DATE, NO_OF_DAYS, TOTAL_PRICE, GUEST_ID, ROOM_ID)



GUEST(GUEST_ID, GUEST_LNAME, GUEST_FNAME, GUEST_PID, GUEST_CITY, GUEST_COUNTRY, GUEST_PHONE_NO



ROOM (ROOM_ID, ROOM_NUM, ROOM_AVA, ROOM_FLOOR, ROOM_ISSMOKING, ROOM_RATE)

DATABOOK PART 3 – SQL QUERIES

1. A query that pulls data from one table - The guests who are from the USA.

```
SELECT GUEST_LNAME, GUEST_FNAME
FROM HOTEL_MANAGEMENT.GUEST
WHERE GUEST_COUNTRY = 'USA';
```

	GUEST_LNAME	GUEST_FNAME	GUEST_COUNTRY
▶	SMITH	JOHN	USA
	BROWN	SARAH	USA
	STEVEN	IZZIE	USA

2. A query that pulls data from two tables - Find the room details for a particular guests

```
SELECT  R.ROOM_NUM,  R.ROOM_FLOOR,  G.GUEST_LNAME,  G.GUEST_FNAME,
G.GUEST_CITY
FROM HOTEL_MANAGEMENT.ROOM R
INNER JOIN HOTEL_MANAGEMENT.GUEST G ON R.ROOM_ID = G.ROOM_ID;
```

	ROOM_NUM	ROOM_FLOOR	GUEST_LNAME	GUEST_FNAME	GUEST_CITY
▶	11	2	SMITH	JOHN	NEW YORK
	12	4	BROWN	SARAH	LOS ANGELES
	21	7	LEE	DAVID	TORONTO
	22	10	JOHNSON	EMILY	LONDON
	32	8	KHAN	IRA	MUMBAI
	36	12	STEVEN	IZZIE	MEXICO

3. A query that also includes a subquery – The count of staff in a restaurant

```
SELECT S.STAFF_LNAME, S.STAFF_FNAME, R.REST_NAME,
      (SELECT COUNT(*)
        FROM HOTEL_MANAGEMENT.STAFF S2 WHERE S2.REST_ID = R.REST_ID) AS
NUM_STAFF
FROM HOTEL_MANAGEMENT.STAFF S
JOIN HOTEL_MANAGEMENT.RESTAURANT R ON S.REST_ID = R.REST_ID
WHERE S.STAFF_POSITION = 'CHEF';
```

	STAFF_LNAME	STAFF_FNAME	REST_NAME	NUM_STAFF
▶	LEE	SAM	CRAZY CAFE	1