Discussion

Atharv Pandit, apandit3

Harshwardhan Joshi, hjoshi2

Muhammad Ali Qureshi, mquresh

Sogol Mansouri, smansou2

<u>Extended FD List</u>:

1) **Manufacturer, ServiceId->Duration**
2) **Manufacturer, ServiceCenter, ServiceId->.Cost**
3) **CarId->Manufacturer, Year, Mileage**
4) **CarId->Last Schedule**
5) **CustumerId, Service_CenterId->Standing, Status**
6) **Invoice_Id->CustumerId,Service_CenterId,Paid**
7) **Service_CenterId->Hourly Rate**

All the tables are in BCNF.

Each of the functional dependencies X->Y have X as a Primary Key.

Table Name	NF	Additional Constraint
Valid-Login Validation	BCNF	
Service_Centre-Service Centre Entity	BCNF	Check if Hourly rate is between min and max wage-Captured using check constraint
Manufacturer	BCNF	
Car	BCNF	
Cust_Has_Car-All cars that each customer possess	BCNF	
Services- All types of services	BCNF	
Car_Needs_Service- Used to capture the duration of	BCNF	

Table Name	NF	Additional Constraint
Service		
Car_Has_Cost_Of_Service-Used to capture cost of a service	BCNF	
Employee	BCNF	
Contract Employees	BCNF	
Manager	BCNF	
Receptionist	BCNF	
Hourly Employees	BCNF	
Mechanics	BCNF	
Time_Slot-All timeslots in the month	BCNF	
Mech_time_off-records when mechanic has time off	BCNF	Check if at a particular slot less than 3 mechanics are there in that center if time off is given.
Repairs	BCNF	
Engine	BCNF	
Exhaust	BCNF	
Electrical	BCNF	
Transmission	BCNF	
Tire	BCNF	
Heat_and_AC	BCNF	
Maintenance	BCNF	
Invoice- Generates Invoice	BCNF	
Bookings- Timeslots when a mechanic is booked	BCNF	Check if Mechanic is booked for more than 50 hrs. There should be no double booking-captured by primary key

Table Name	NF	Additional Constraint
Invoice_has_service-All services covered in an invoice	**BCNF**	
Requests		

Additional Constraints-

- Each Employee is part of a single service center-Using a foreign key which will not be null in the employees table.
- Each Manager manages employees of that service center-Since the employees table has a foreign key on the service center, an employee in the manager table for that service center will be the manager for all the employees for that service center.
- Employees are segregated into specialized roles-captured using foreign keys for separate tables. Eg- Manager with a foreign key with parent as employee table.
- A center might operate 5 days a week or also on Saturdays-Boolean flag in the service center table that indicates whether it is open on saturdays.
- Duration of each service is the same for a model-A table that stores the duration of a service for a model. Implemented using Foreign Keys.
- Cost of each service is the same for a model at a service center-A table that stores the cost of a service at a service center for a model. Implemented using Foreign Keys.
- A service may be maintenance as well as repair- Captured by adding it to both the tables. Also have a foreign key where the parent is the service table.
- Customer is unique to a specific center- Hence it is modeled as a weak entity. It will be identified using a customerid and the service center.
- There should not be any double booking for a mechanic-captured using primary key

Advance SQL Features-

Triggers:

- Triggers to check the customer's status-
    - This value is to be updated to 0 when a customer has no cars. This trigger is fired whenever there is a delete on the car table. It will first delete the record of the customer and car mapping table and then update the status of that customer to 0 if there are 0 cars for that customer.
    - Another trigger is used to update the customer's status when a new car is added. If the status was previously 0, it is updated to 1.
- Trigger to add manufacturer.
    - Whenever a customer adds a car, an entry is made in the manufacturer's table for that manufacturer if it does not exist.
- Trigger to update standing:

    - Trigger is fired when there is an addition or update to invoice and it changes the customer's standing, i.e whether the customer has unpaid invoices.

Constraints implemented in code:

- Giving a mechanic time-off only when there are at least 3 mechanics in the store.
    - We first iterate through each time slot in the range provided by the mechanic. We check the count of mechanics available at the store at that time.
    - If the count is <=3 at any of these time slots, the time-off will not be approved.
- Check if mechanic is not allocated more than 50 hrs in the week.
- Swapping Time Slots- Check if the time slot is allocated to the mechanic for which it is requested.
- Selecting the next maintenance service.
- Checking if mechanic is not getting double booked.