SHAHEED ZULFIKAR ALI BHUTTO
INSTITUTE OF SCIENCE & TECHNOLOGY

SZABIST UNIVERSITY

Bachelor's of Computing Science Department

Research Report

Database Management System Lab

By

Syed Hassan Ali

Humza Hussain

Abdullah Shaikh

Mustafa Ahmed Zaidi

Umair Rasheed

SUBMITTED

TO

Miss.Sadia Aziz

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to all those who provided me the possibility to complete this report. A special thanks goes to my team mate, [Hamza Hussain, Mustafa Ahmed Zaidi, Umair Rasheed, Abdullah Shaikh], who help me to assemble the parts and gave suggestion about the project “FACTORY ROASTER”. Last but not least, many thanks go to the head of the project, [Hamza Hussain] whose have invested his full effort in guiding the team in achieving the goal. I have to appreciate the guidance given by other Teachers especially Ma’am Marjan Sikandar, as well as the seniors especially in our project presentation that has improved our presentation skills thanks to their comment and advices.

ABSTRACT

Project Details:

- The factory has various floors.
- Each floor runs a specific number of shifts per day (1 or 2 or 3) with their start and end time specified. Each floor also specifies whether there are any weekly holidays (weekends)
- The factory has many employees. Each employee has a designation/rank.
- Each floor has various stations and for each station, the number of employees of each rank required to run the station are specified.
- When a roster is created (for a week or fortnight or a month), employee assignments are made to different shifts and at different stations belonging to various floors.
- On the basis of assignments, the actual attendance and hours of each employee would be tracked by storing the sign in and sign out timings of employees.
- Employee leaves and also maintained in the system. In addition, public and company holidays are also maintained.

Expected Volume:

- 3 floors * 2 shifts * 10 stations per floor * avg. 5 employees per station * 250 days (365 excluding weekends &amp; public holidays) * 1 years of assignments = 75,000 assignments in the roster.
- Sign in and out times of each employee for each day

TABLE OF CONTENTS

1 INTRODUCTION

This intention behind creating this database was to ease storing and retrieving of data associated with employees working in a factory. We have employed many techniques to resolve and normalize our database design as much as it was possible, so we can have a functioning and robust database when once implemented.

1.1 MOTIVATION FOR THE PROJECT

The factor that motivated us a lot was fluent db design and use of tools such as Microsoft Visio to make a robust database. We wanted to experiment with different tools for creating data definition language and modelling relationships in our database. No opportunity was better than this project to get hands on experience on all of the various tools out there.

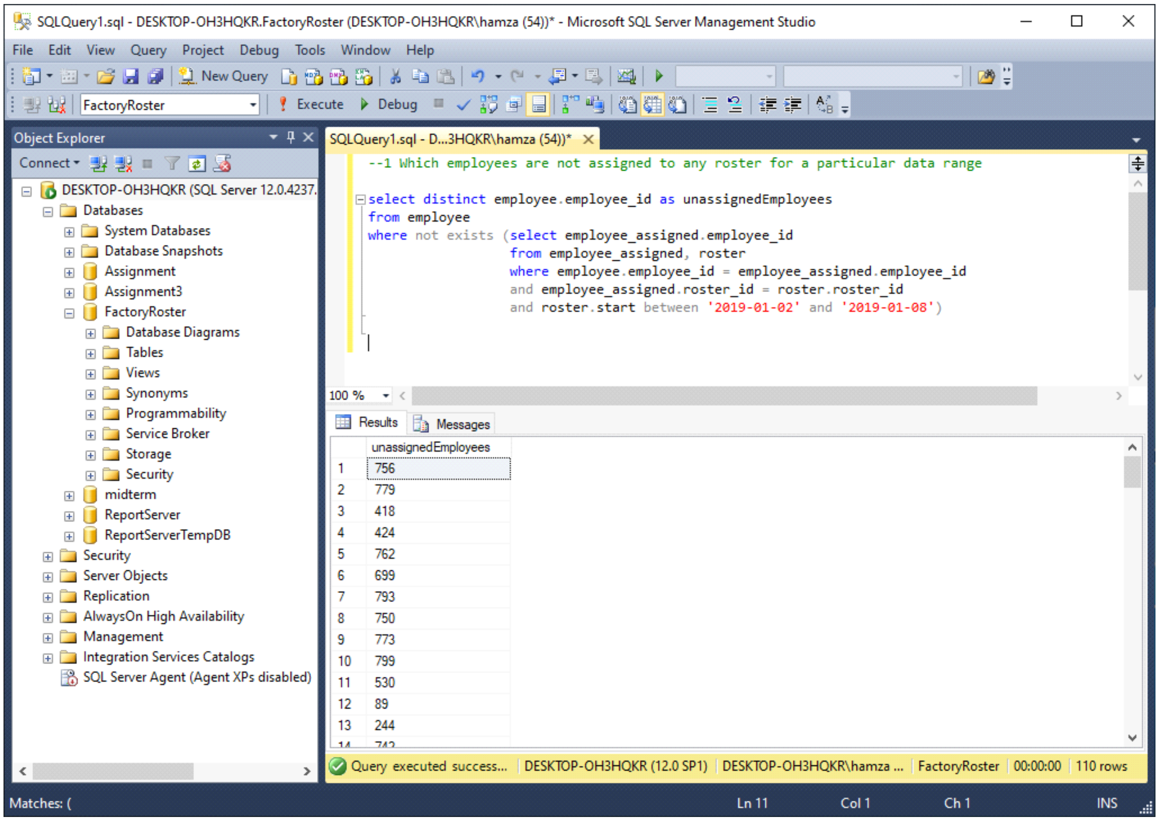
2 BACKGROUND

The factory has various floors. Each floor runs a specific number of shifts per day (1 or 2 or 3) with their start and end time specified. Each floor also specifies whether there are any weekly holidays (weekends) factory has many employees. Each employee has a designation/rank. Each floor has various stations and for each station, the number of employees of each rank required to run the station are specified. The employees are stationed such that their rank increases with the number of stations.

3 WORKING OF PROJECT

When a roster is created (for a week or fortnight or a month), employee assignments are made to different shifts and at different stations belonging to various floors. On the basis of assignments, the actual attendance and hours of each employee would be tracked by storing the sign in and sign out timings of employees. Employee leaves and also maintained in the system. In addition, public and company holidays are also maintained. Project can work efficiently with two shifts a day at most, one starting from 06:00:00 to 12:00:00 and second shift starting from 12:00:00 to 18:00:00. Project makes use of many data types especially those of date time and time.

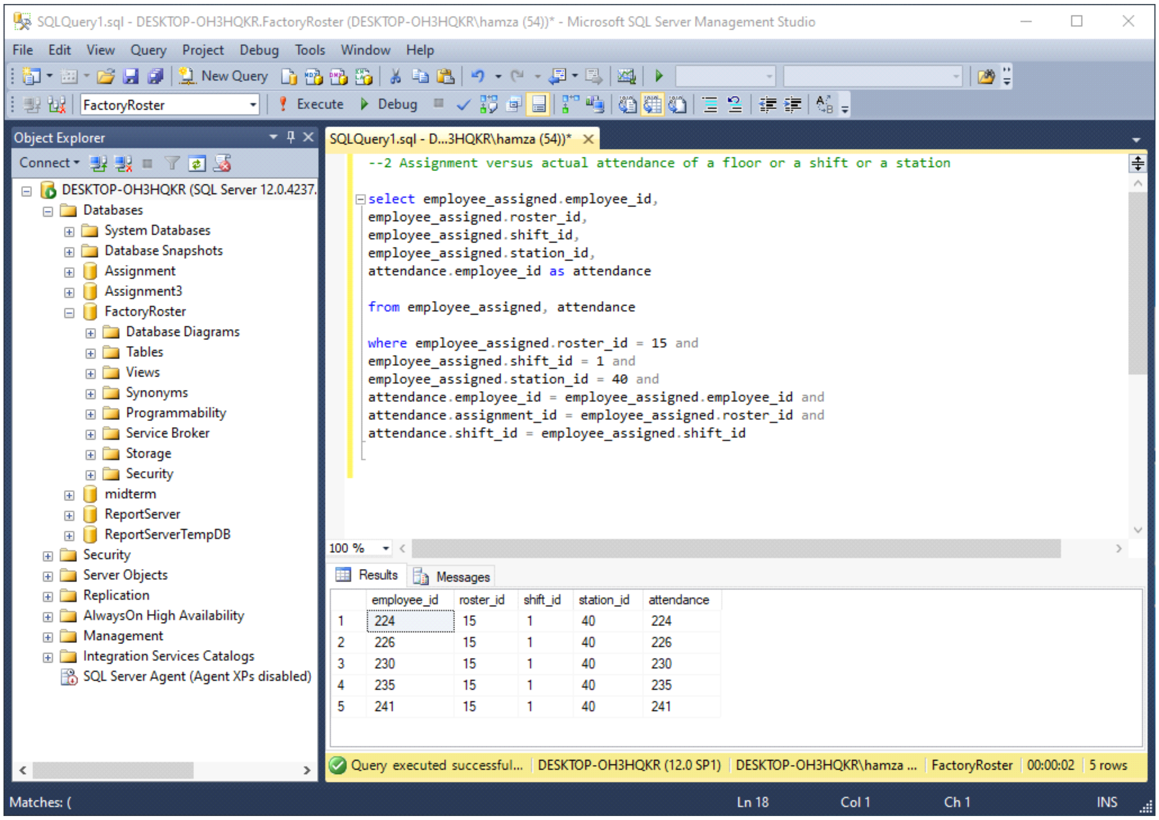
3.1 QUERIES



```sql
--1 Which employees are not assigned to any roster for a particular data range

select distinct employee.employee_id as unassignedEmployees
from employee
where not exists (select employee_assigned.employee_id
                  from employee_assigned, roster
                  where employee.employee_id = employee_assigned.employee_id
                  and employee_assigned.roster_id = roster.roster_id
                  and roster.start between '2019-01-02' and '2019-01-08')
```

	unassignedEmployees
1	756
2	779
3	418
4	424
5	762
6	699
7	793
8	750
9	773
10	799
11	530
12	89
13	244
14	742

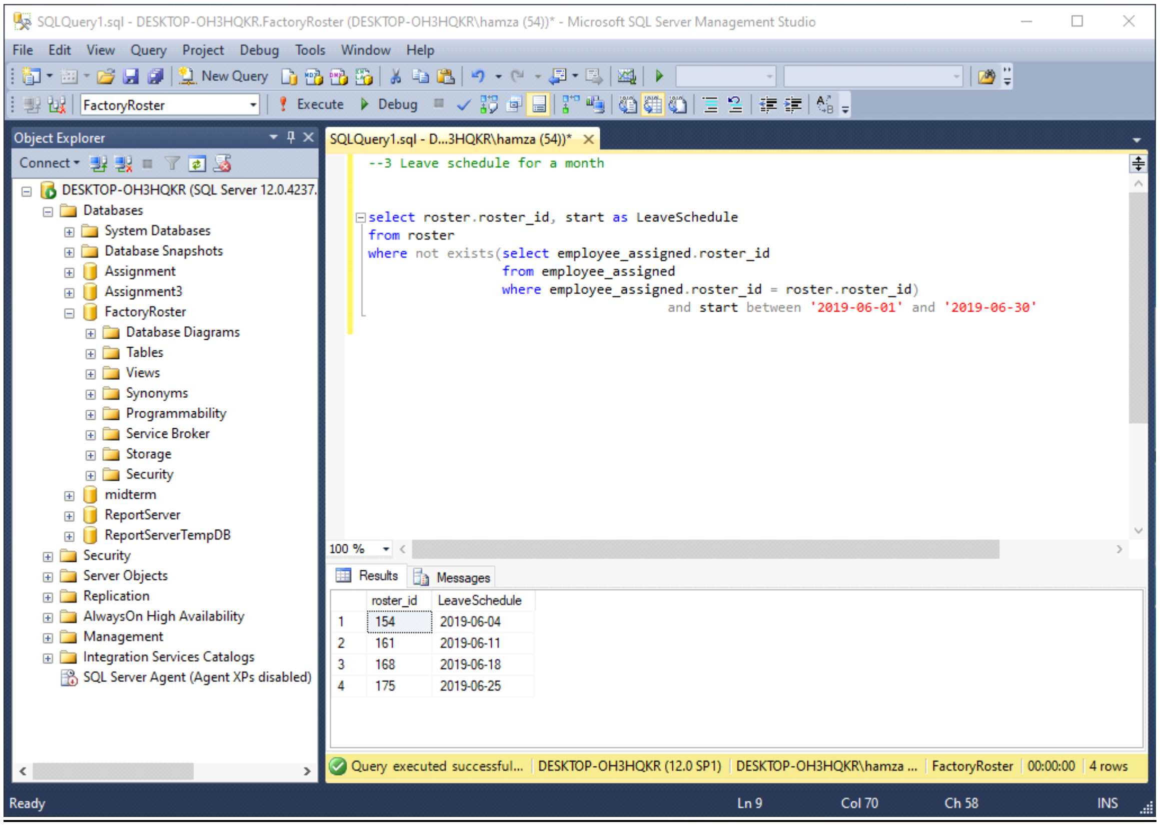
Query executed success... | DESKTOP-OH3HQKR (12.0 SP1) | DESKTOP-OH3HQKR\hamza ... | FactoryRoster | 00:00:00 | 110 rows



```sql
--2 Assignment versus actual attendance of a floor or a shift or a station

select employee_assigned.employee_id,
employee_assigned.roster_id,
employee_assigned.shift_id,
employee_assigned.station_id,
attendance.employee_id as attendance

from employee_assigned, attendance

where employee_assigned.roster_id = 15 and
employee_assigned.shift_id = 1 and
employee_assigned.station_id = 40 and
attendance.employee_id = employee_assigned.employee_id and
attendance.assignment_id = employee_assigned.roster_id and
attendance.shift_id = employee_assigned.shift_id
```

	employee_id	roster_id	shift_id	station_id	attendance
1	224	15	1	40	224
2	226	15	1	40	226
3	230	15	1	40	230
4	235	15	1	40	235
5	241	15	1	40	241

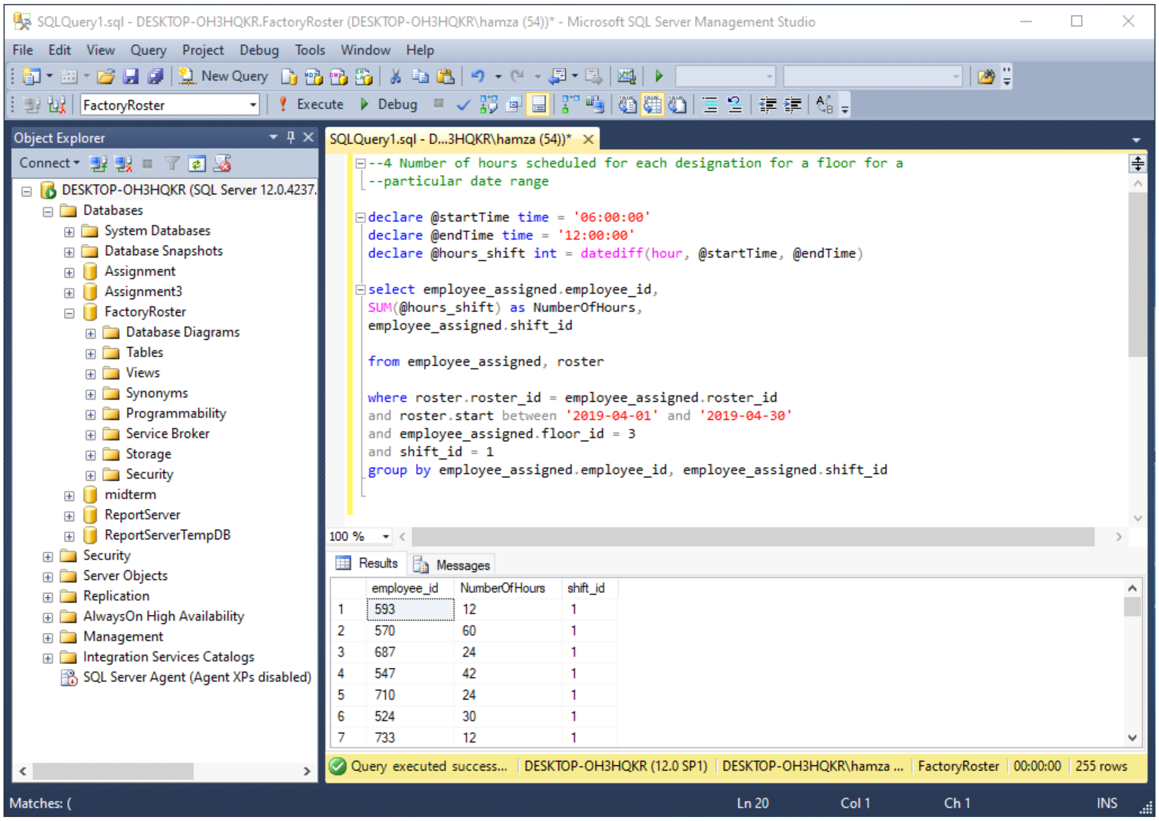
Query executed successful... | DESKTOP-OH3HQKR (12.0 SP1) | DESKTOP-OH3HQKR\hamza ... | FactoryRoster | 00:00:02 | 5 rows


SQLQuery1.sql - DESKTOP-OH3HQKR.FactoryRoster (DESKTOP-OH3HQKR\hamza (54))* - Microsoft SQL Server Management Studio
File Edit View Query Project Debug Tools Window Help
New Query
FactoryRoster
Execute
Debug
Object Explorer
Connect
DESKTOP-OH3HQKR (SQL Server 12.0.4237.
Databases
System Databases
Database Snapshots
Assignment
Assignment3
FactoryRoster
Database Diagrams
Tables
Views
Synonyms
Programmability
Service Broker
Storage
Security
midterm
ReportServer
ReportServerTempDB
Security
Server Objects
Replication
AlwaysOn High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
SQLQuery1.sql - D...3HQKR\hamza (54))*
100 %
Results
Messages
Ready
Ln 9
Col 70
Ch 58
INS

```sql
--3 Leave schedule for a month


select roster.roster_id, start as LeaveSchedule
from roster
where not exists(select employee_assigned.roster_id
                 from employee_assigned
                 where employee_assigned.roster_id = roster.roster_id)
                                      and start between '2019-06-01' and '2019-06-30'
```

	roster_id	LeaveSchedule
1	154	2019-06-04
2	161	2019-06-11
3	168	2019-06-18
4	175	2019-06-25

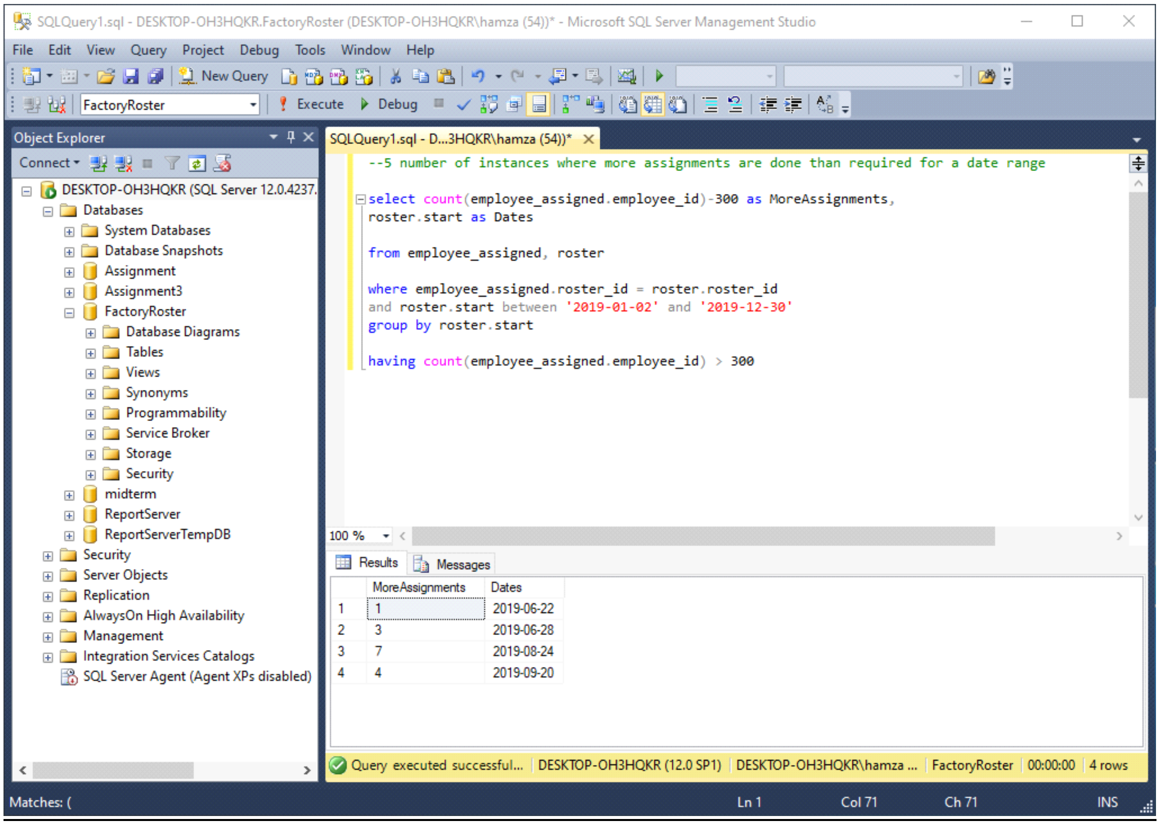
Query executed successful... | DESKTOP-OH3HQKR (12.0 SP1) | DESKTOP-OH3HQKR\hamza ... | FactoryRoster | 00:00:00 | 4 rows



```sql
--4 Number of hours scheduled for each designation for a floor for a
--particular date range

declare @startTime time = '06:00:00'
declare @endTime time = '12:00:00'
declare @hours_shift int = datediff(hour, @startTime, @endTime)

select employee_assigned.employee_id,
SUM(@hours_shift) as NumberOfHours,
employee_assigned.shift_id

from employee_assigned, roster

where roster.roster_id = employee_assigned.roster_id
and roster.start between '2019-04-01' and '2019-04-30'
and employee_assigned.floor_id = 3
and shift_id = 1
group by employee_assigned.employee_id, employee_assigned.shift_id
```

	employee_id	NumberOfHours	shift_id
1	593	12	1
2	570	60	1
3	687	24	1
4	547	42	1
5	710	24	1
6	524	30	1
7	733	12	1

Query executed success... | DESKTOP-OH3HQKR (12.0 SP1) | DESKTOP-OH3HQKR\hamza ... | FactoryRoster | 00:00:00 | 255 rows

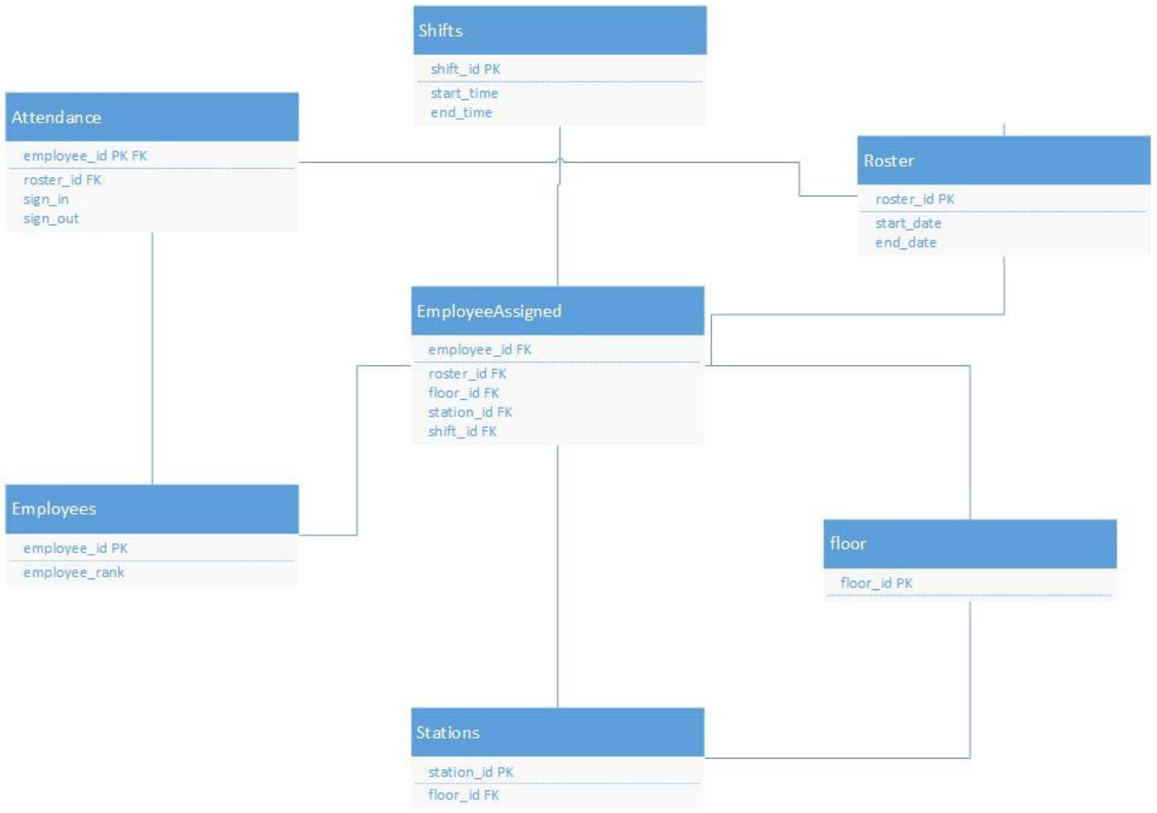


```sql
--5 number of instances where more assignments are done than required for a date range

select count(employee_assigned.employee_id)-300 as MoreAssignments,
roster.start as Dates

from employee_assigned, roster

where employee_assigned.roster_id = roster.roster_id
and roster.start between '2019-01-02' and '2019-12-30'
group by roster.start

having count(employee_assigned.employee_id) > 300
```

	MoreAssignments	Dates
1	1	2019-06-22
2	3	2019-06-28
3	7	2019-08-24
4	4	2019-09-20

Query executed successful... | DESKTOP-OH3HQKR (12.0 SP1) | DESKTOP-OH3HQKR\hamza ... | FactoryRoster | 00:00:00 | 4 rows

3.2 ERD


Shifts
shift_id PK
start_time
end_time
Attendance
employee_id PK FK
roster_id FK
sign_in
sign_out
Roster
roster_id PK
start_date
end_date
EmployeeAssigned
employee_id FK
roster_id FK
floor_id FK
station_id FK
shift_id FK
Employees
employee_id PK
employee_rank
floor
floor_id PK
Stations
station_id PK
floor_id FK

3 SCOPE OF PROJECT

This project can work very efficiently and will satisfy most of the industry standards. It can be tweaked a little to fit to any factory's specific requirements and will prove very optimal in organizing and maintaining factory's data.

4 CONCLUSION

I will conclude this project with the various tools, things, and techniques we learned along the way. Before starting out we didn't know much about transact-SQL, data base design tools and advanced normalization but after completing the project successfully we do have a good grasp and understanding of the fundamentals behind these things.

6 REFERENCES

https://www.geeksforgeeks.org/sql-join-set-1-inner-left-right-and-full-joins/

https://docs.microsoft.com/en-us/sql/t-sql/functions/date-and-time-data-types-and-functions-transact-sql?view=sql-server-ver15

https://www.mssqltips.com/sqlservertip/1145/date-and-time-conversions-using-sql-server/

https://www.w3schools.com/sql/