**FIX ‘EM EQUIPMENT MAINTENANCE GROUP**

**Aadith Narayan Ravishankar**

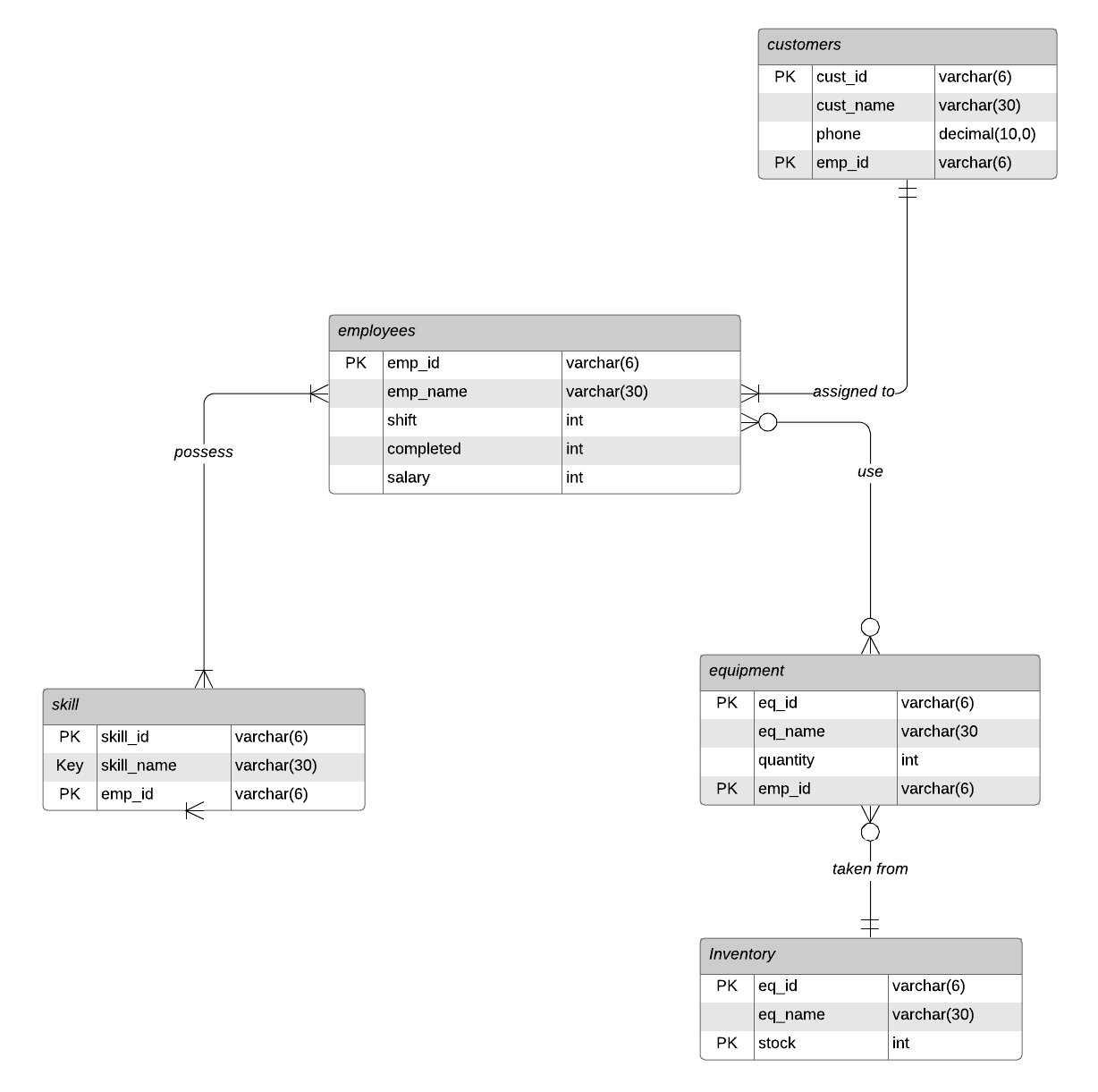
**AXR180085**

**Executive summary**

The database objective is to display the details of employees especially their skills, scheduling of employees to customers and also to manage the equipment from the inventory. Each employee would have a detailed description of their skills and their scheduling would be respect to their shifts. The salary of the employee is calculated by a base salary with a bonus related to the number of jobs completed. The details of the customers are collected and the administrator assigns employees to customers based on the employee skill and shift. Before commencing the job, the employee can get equipment from the inventory. The equipment can only be received if the item is on stock. The administrator records this equipment transaction and makes sure that the inventory has the necessary items which are required for employees.

**Business Rules**

1. Each employee will have one unique emp\_id and no two employees will possess the same emp\_id
2. An employee should have minimum one skill and can have more than one skill
3. Each skill will have a unique skill\_id and no two skills will possess the same skill\_id
4. Employees can be assigned to either the 1st or 2nd shift only.
5. Salary is calculated with a base value of 8000$ with 500$ for each job completed(8000+500\*completed)
6. Each customer can be assigned to minimum one employee or more than one employee
7. Each employee can be assigned to only one customer
8. Customers phone no should be of 10 digits
9. Each customer will have a unique cust\_id assigned and no two customers will be assigned the same cust\_id
10. Each employee can handle zero or more equipment.
11. Each equipment will have a unique EQ\_id and no two equipment will possess the same EQ\_id
12. Employees can take the needed quantity of equipment required provided that it is in stock. That is, the quantity should always be less than or equal to the stock.
13. An employee can take multiple equipment from the inventory provided it is in stock
14. The company has 25 technical employees and some have multiple skills
15. The company has 18 tools and equipment at their disposal to use it in the field
16. An employee cannot partake in both the shifts
17. Each equipment taken must be returned by the employee to the inventory to replenish the stock
18. The schedule administrator and the technical manager are the only users to have full access to the database
19. Employees must work for one customer only unless there is a change in the schedule. Completion of job is only with respect to the assigned customer
20. Two customers are allowed to have the same phone number provided that they are related to each other



**ERD Components**

Entities

1. employees
2. skill
3. customers
4. equipment
5. inventory

Attributes of Entities

1. employees- emp\_id,emp\_name,shift,completed,salary
2. skill- skill\_id,skill\_name,emp\_id
3. customers- cust\_id,cust\_name,phone,emp\_id
4. equipment- eq\_id,eq\_name,quantity,emp\_id
5. inventory- eq\_id,eq\_name,stock

Relationships

1. Employees assigned to customers
2. Employees possess skills
3. Employees use equipment
4. Equipments taken from inventory

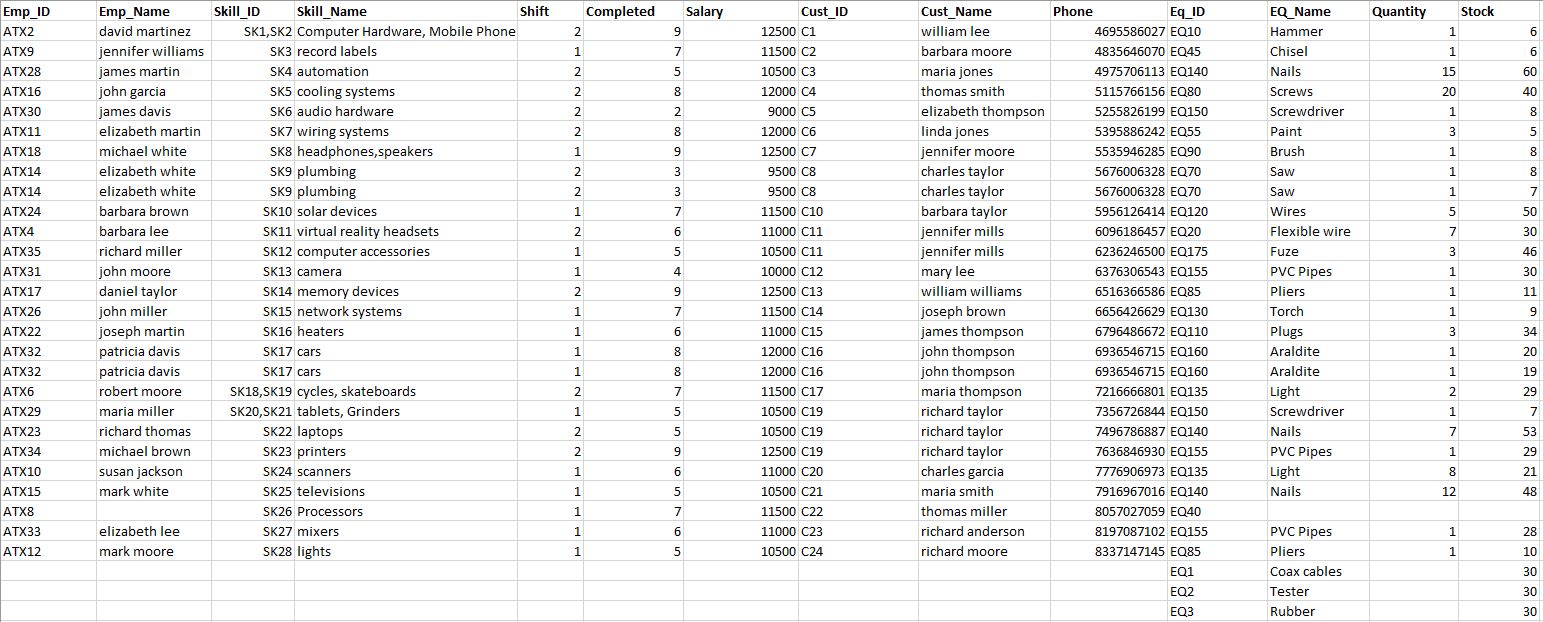
Cardinality of Relationships

1. Cardinality of Employees is 1
2. Cardinality of skills is 2
3. Cardinality of customers is 3
4. Cardinality of equipment is 3
5. Cardinality of inventory is 3

Key description

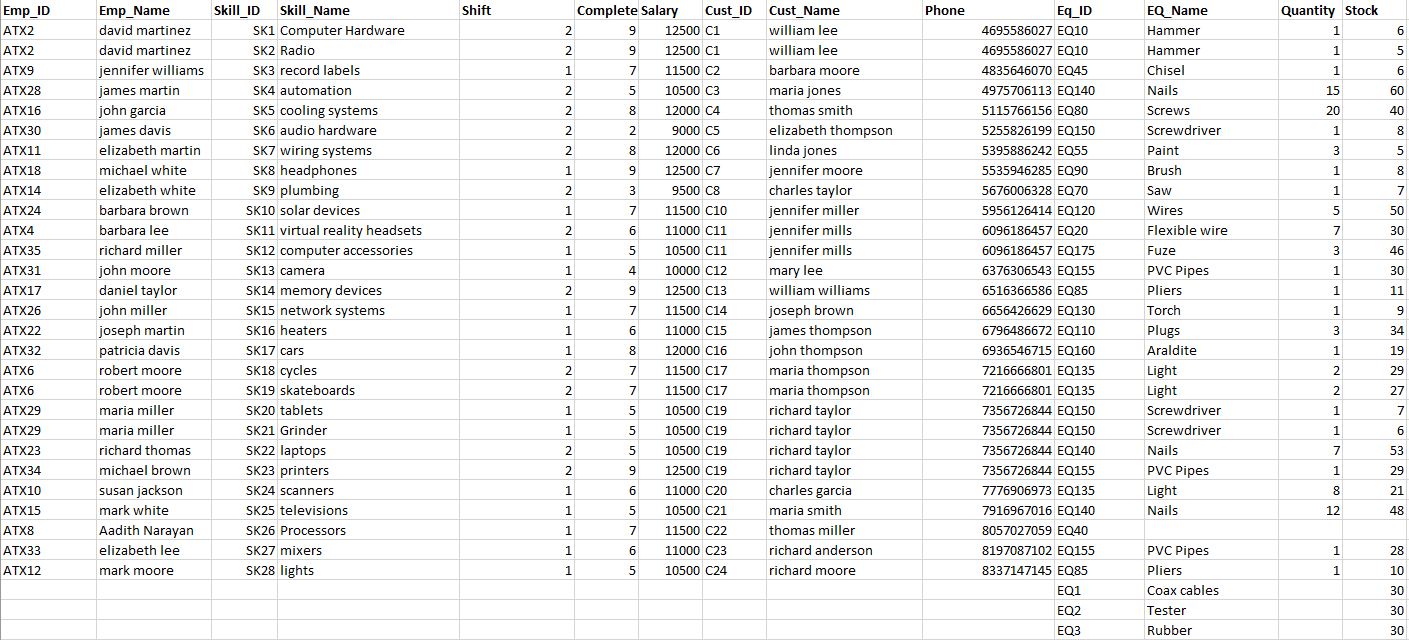
1. Employees- PK (emp\_id)
2. Skill- PK(skill\_id,emp\_id), FK(emp\_id)
3. Customers- PK(cust\_id, emp\_id), FK(emp\_id)
4. Equipment- PK(eq\_id, emp\_id), FK(emp\_id)
5. Inventory- PK(eq\_id,stock), FK(eq\_id)

**UNF**



The data is created according to the business rules. The data contains redundant data which shows a need for normalization.

**1NF**



The UNF contains duplicate rows, multiple data values which have to be normalized. All the duplicate rows are removed from the dataset. The multiple values are broken down and inserted into new rows to make the values singular and atomic.

**2NF**

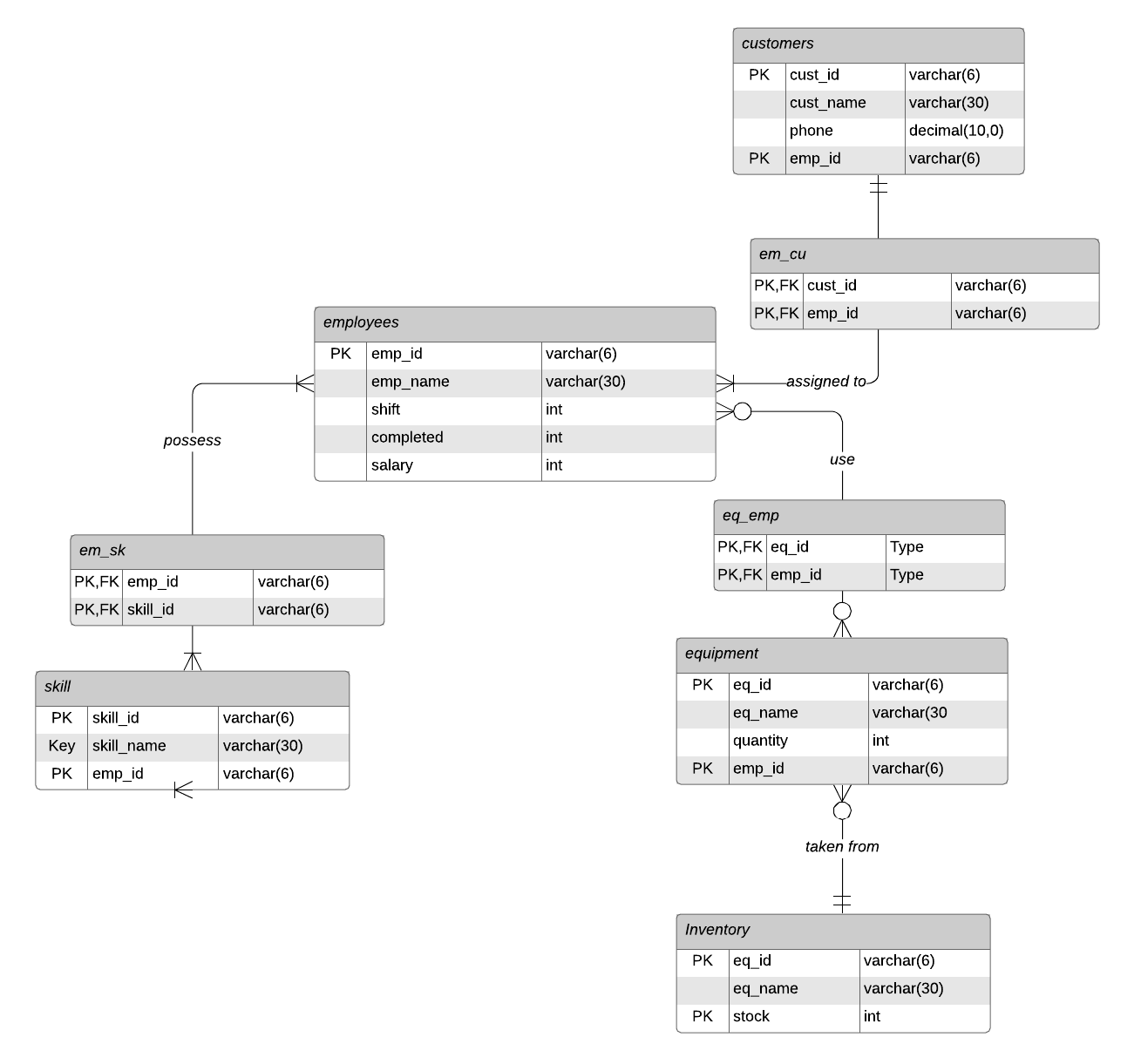
Primary keys

1. Employees- emp\_id
2. Skill- skill\_id,emp\_id
3. Customers- cust\_id, emp\_id
4. Equipment- eq\_id, emp\_id
5. Inventory- eq\_id,stock

Partial Dependencies

1. Employees- No partial dependencies
2. Skill- Skill\_id -> Skill\_name
3. Customers- cust\_id -> cust\_name, phone
4. Equipment- eq\_id -> eq\_name, quantity
5. Inventory- No partial dependencies

The solution which I used was to create conjunction tables which defines the relationships between two entities. For example, a conjunction table em\_sk contains emp\_id, skill\_id and relates the employees with their skills. Here emp\_id and skill\_id are both primary keys and they are foreign keys to their respective parent entities. The ERD to visualize these conjunction tables is below:

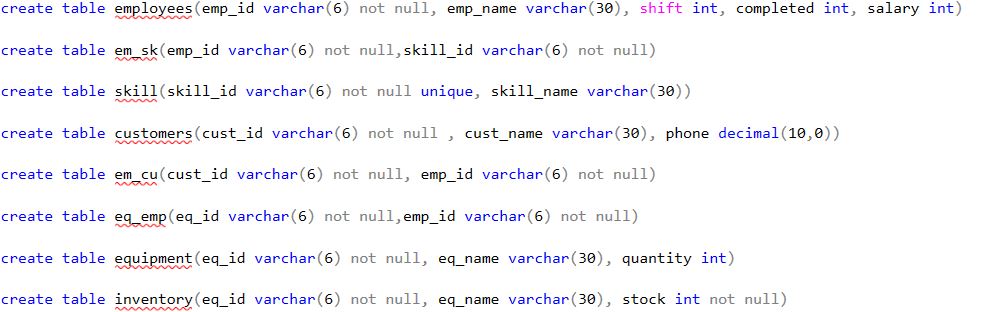
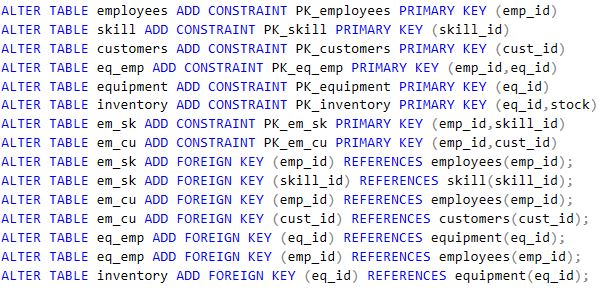
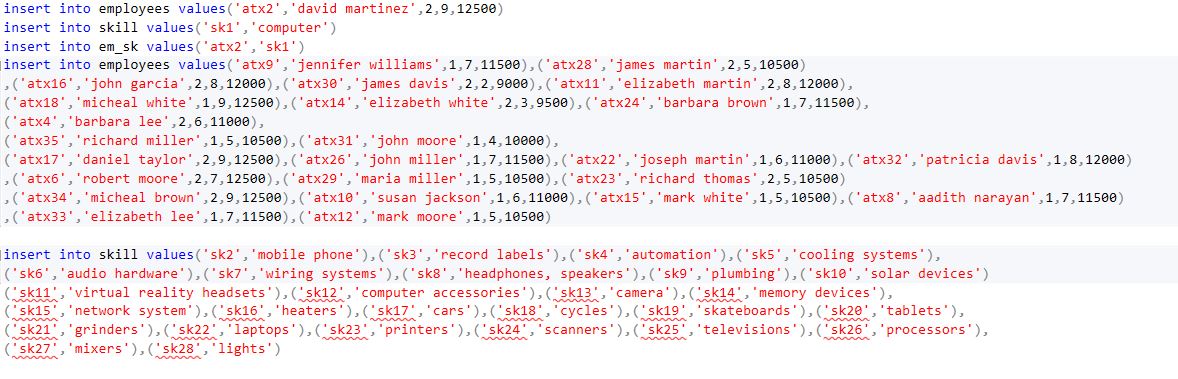
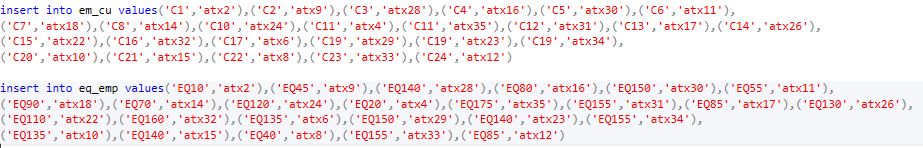
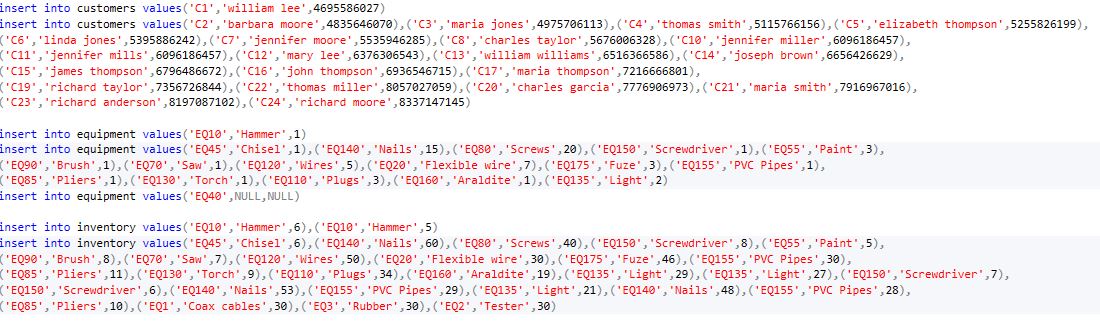
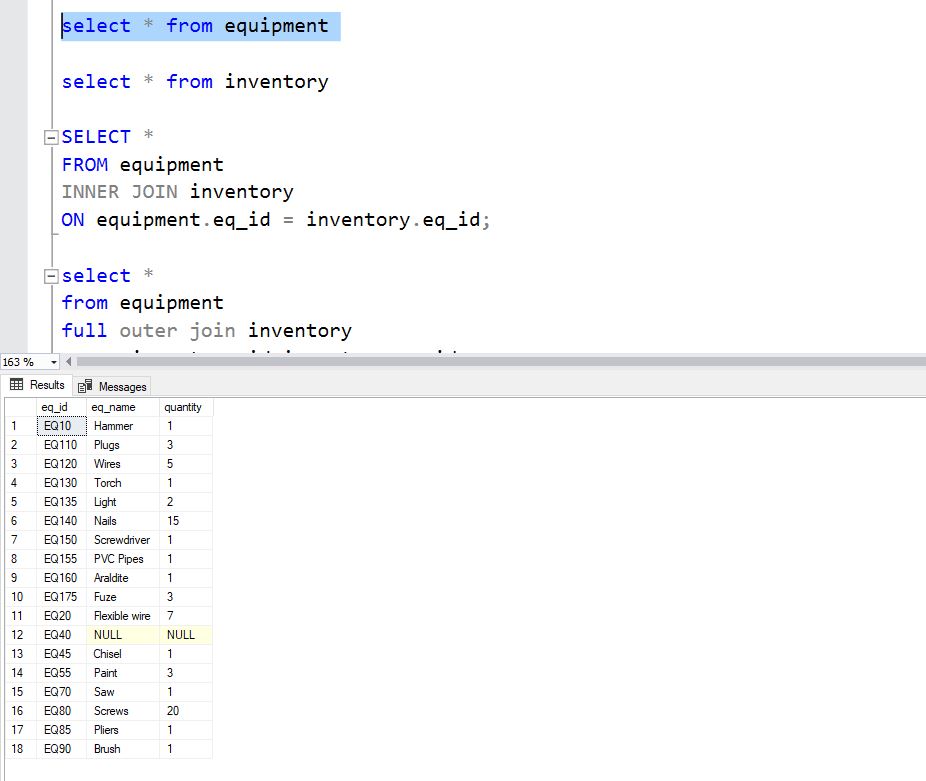


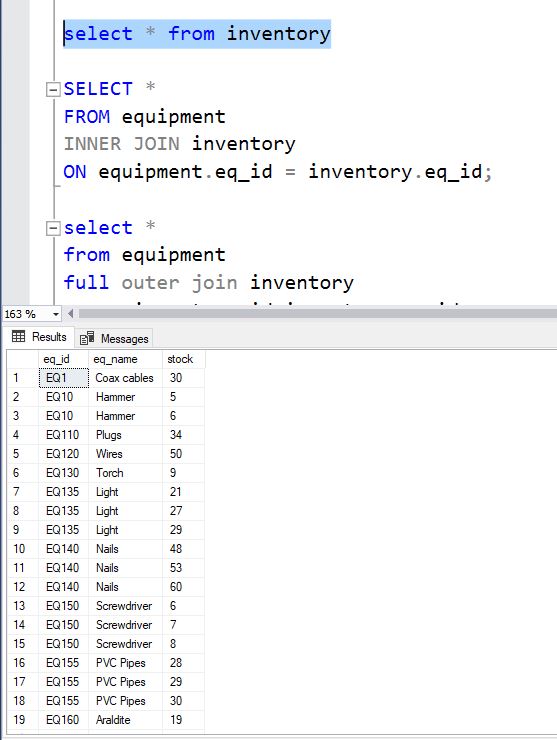
There are no transitive dependencies in this data. The table conforms to 3NF and BCNF.

**SQL Scripting**

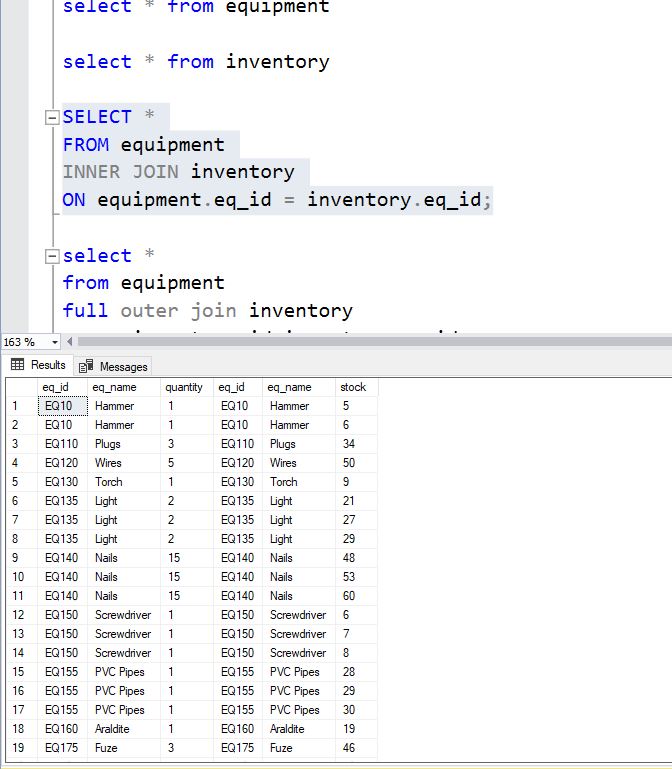
1. A database maintenance is created.



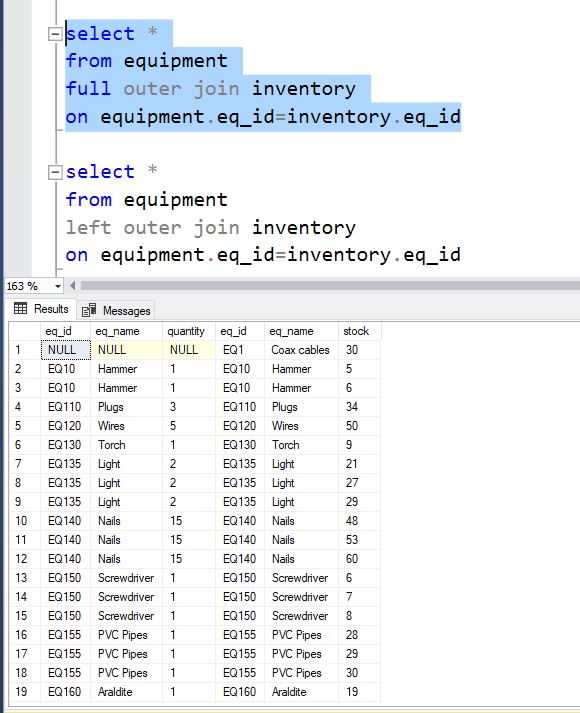
1. The tables are created without specifying key constraints.
2. Primary and Foreign key constraints are created
3. The data is inserted into the table with respect to referential integrity
4. The two tables for performing sql scripting on are equipment and inventory.
5. **Showi**n**g records of “equipment”**



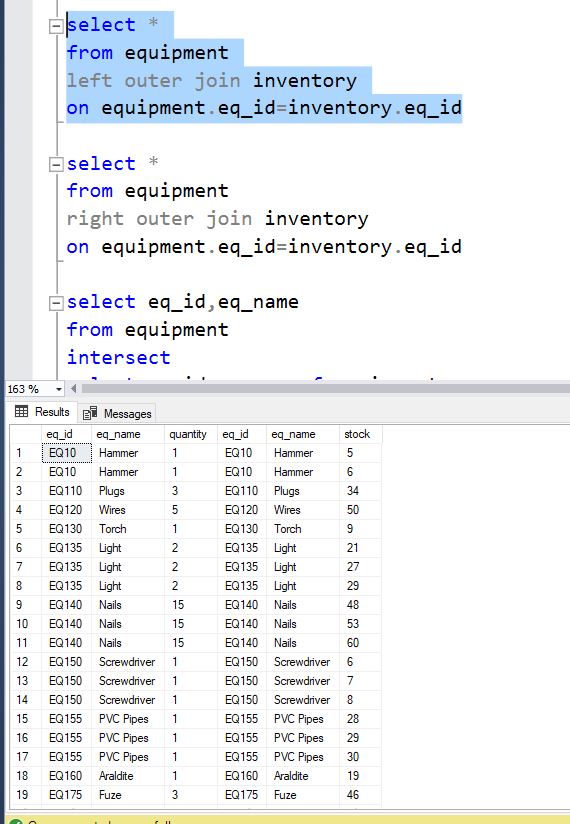
1. **Showing records of “inventory”**



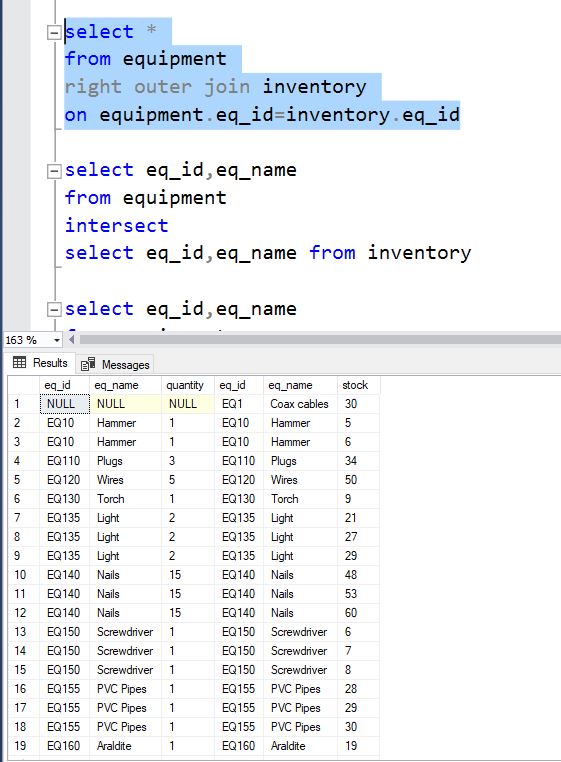
1. **Executing Inner join**

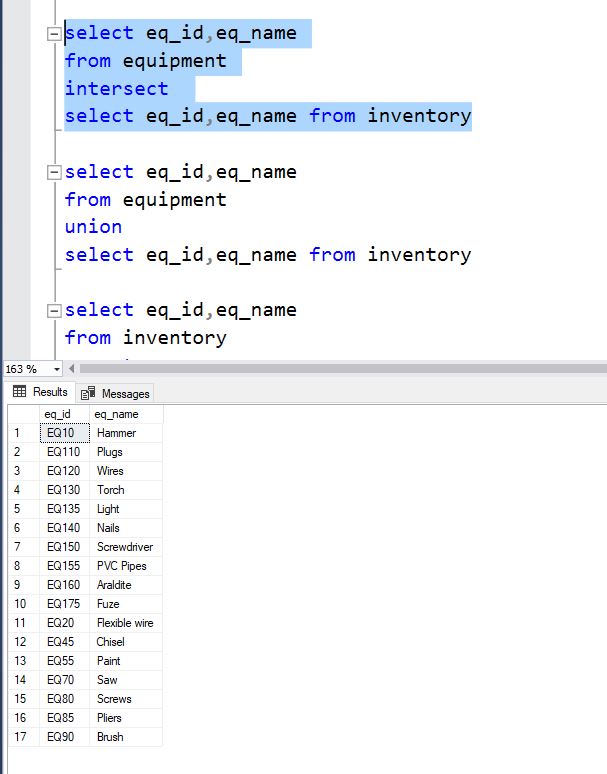
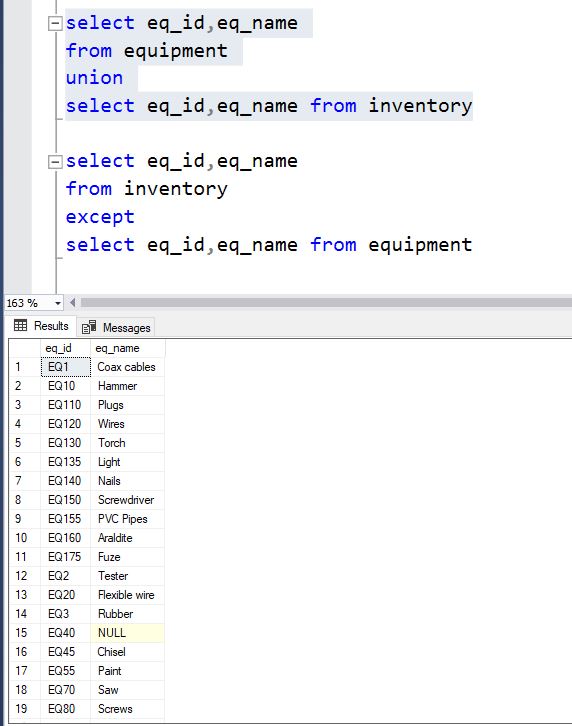


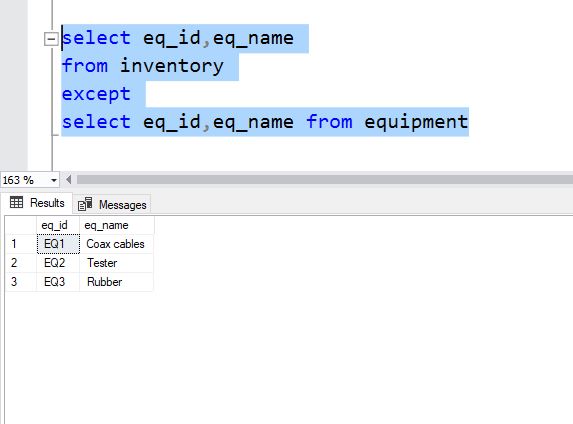
1. **Executing full outer join**



1. **Executing left outer join**



1. **Executing right outer join**
2. **Executing intersect**
3. **Executing union**



1. **Executing intersect**