

Project Report

CSC623 | THEORY OF RELATIONAL DATABASES

Team: Pornkamon Sirivoravijit | Roxana Rodriguez | Sunity Sharma

INSTRUCTOR: DR. VANESSA AGUIAR

CONTENTS

Introduction	2
Case Study: SuperMaids Cleaning Company	3
Part 1: Develop a conceptual data model.....	4
A. Identify the main entity types.....	4
B. Identify the main relationship types between the entity types identified in "a". ...	5
C. Determine the multiplicity constraints for each relationship identified in "b".	5
D. Identify attributes and associate them with entity or relationship types.	6
E. Determine candidate and primary key attributes for each (strong) entity type.	6
F. Generate the ER diagram for the conceptual level (no FKs as attributes).....	7
PART 2: DEVELOP A LOGICAL DATA MODEL BASED ON THE FOLLOWING REQUIREMENTS	8
A. Derive relations from the conceptual model.....	8
B. Validate THE Logical MODEL USING NORMALIZATION TO 3NF.....	9
C. VALIDATE THE Logical MODEL AGAINST USER Transactions	11
D. Define integrity constraints.....	12
E. generate the er-diagram for the logical level (contains FK as attributes).....	14
PART 3: Translate the logical data model for the DBMS.....	15
A. Develop SQL code to create the entire database schema, reflecting the constraints identified in previous steps.....	15
B. Create at least 5 tuples for each relation in your database.....	16
C. Develop 5 SQL queries using embedded SQL.....	17
D. GITHUB Repository and Documentation details	18

INTRODUCTION

In the realm of database management, we embark on a journey into the project of the SuperMaids Cleaning Company. The fundamental aim of this project is to architect an Entity-Relationship (ER) diagram and construct a robust database infrastructure to efficiently capture, organize, and preserve the wealth of data generated by this dynamic cleaning company.

In this project, we dig into the intricate relationships between clients, employees, and equipment. Our mission is to create a data repository that not only records but also strategically manages information related to service schedules, client preferences, staff details, and equipment utilization. By undertaking this project, we intend to streamline the company's operations, enhance data accessibility, apply a three normal form (3NF), and facilitate informed decision-making.

The ensuing sections of this report will unfold the comprehensive process of conceptualizing, designing, and implementing a database solution tailored to the unique needs and challenges faced by the SuperMaids Cleaning Company. Through the development of both a conceptual and logical data model, we aim to provide a comprehensive database management solution that aligns seamlessly with the company's business objectives and empowers its operations.

CASE STUDY: SUPERMAIDS CLEANING COMPANY

The SuperMaids Cleaning Company specializes in providing cleaning services for clients. Each type of client has a set of requirements. For example, The Cardboard Box Company requires cleaning services from Monday to Friday 7am until 9am and 5pm until 7pm each day, but P. Nuttall only requires cleaning services on a Wednesday from 10am until 1pm.

Whenever a new client is taken on, it is determined whether any special equipment is required and when. For example, three industrial floor cleaners may be needed on two out of five occasions for one client. Therefore, the following information will be stored for each equipment, in addition to the equipment identifier: description, usage and cost.

For each employee, the following data will be stored: staff number (uniquely identifies an employee), first and last name, address, salary and telephone number. For each client, the following data will be stored: client number (uniquely identifies a client), first and last name, address and telephone number.

PART 1: DEVELOP A CONCEPTUAL DATA MODEL

A. IDENTIFY THE MAIN ENTITY TYPES.

Entity Name	Description	Aliases	Occurrence
Client	General term describing all clients of the SuperMaid Cleaning company	Customer	Each client can request several services
Service	General term describing all services requested by client	Requirement	Each service has a single client
Equipment	General term describing all equipment required by the client service	Tool	Each equipment is required by client, one equipment can be used for different service
Employee	General term describing all employees employed by SuperMaid Cleaning company	Staff	Each employee is assigned to work on the service that requested by client

- Table shows details of the project – before working on Question B to C

Entity1	Relationship	Entity2	Participation	Cardinality	Multiplicity	Type of Relationship
Client	Requests	Service	1	*	1..*	1:*
Service	IsRequestedBy	Client	1	1	1..1	
Service	Requires	Equipment	0	*	0..*	1:*
Equipment	IsUsedFor	Service	1	*	1..*	
Employee	WorksOn	Service	1	*	1..*	1:*
Service	IsAssignedTo	Employee	1	*	1..*	

B. IDENTIFY THE MAIN RELATIONSHIP TYPES BETWEEN THE ENTITY TYPES IDENTIFIED IN "A".

Entity Name	Multiplicity	Relationship	Entity Name2	Type of Relationship
Client	1..*	Requests	Service	1:*
Service	1..1	IsRequestedBy	Client	
Service	0..*	Requires	Equipment	1:*
Equipment	1..*	IsUsedFor	Service	
Employee	1..*	WorksOn	Service	1:*
Service	1..*	IsAssignedTo	Employee	

C. DETERMINE THE MULTIPLICITY CONSTRAINTS FOR EACH RELATIONSHIP IDENTIFIED IN "B".

<u>Client and Service</u> – notation: 1..*
<u>Service and Client</u> – notation: 1..1
<u>Service and Equipment</u> – notation: 0..*
<u>Equipment and Service</u> – notation: 1..*
<u>Employee and Service</u> – notation: 1..*
<u>Service and Employee</u> – notation: 1..*

❑ Assumptions:

- Each client may have multiple services associated with them.
- Every service must be linked to a client.
- Some services may not require the use of equipment.
- Each piece of equipment must be associated with a service.
- Phone number is a candidate key.

D. IDENTIFY ATTRIBUTES AND ASSOCIATE THEM WITH ENTITY OR RELATIONSHIP TYPES.

- **Clients:** Client Number, First Name, Last Name, Address (composite: street, city, postcode), Phone Number
- **Employees:** Staff Number, First Name, Last Name, Address (composite: street, city, postcode), Salary, Phone Number
- **Service:** ServiceID, Day, Start time, Duration, Comment
- **Equipment:** Equipment ID, Description, Usage Details, Cost

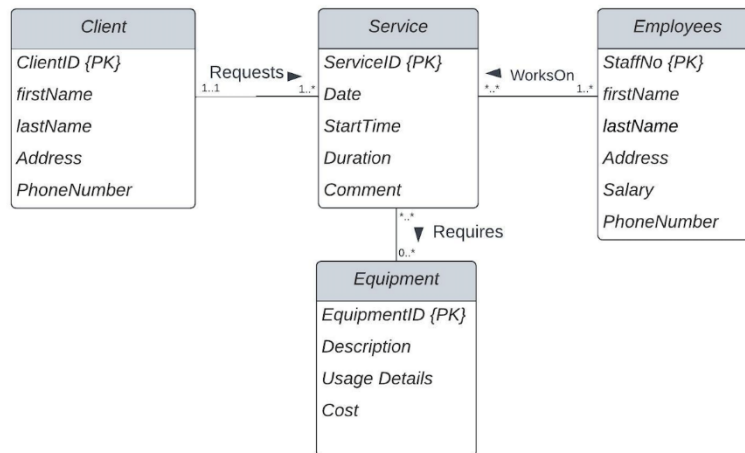
E. DETERMINE CANDIDATE AND PRIMARY KEY ATTRIBUTES FOR EACH (STRONG) ENTITY TYPE.

- **Clients:** Client Number (PK), First Name, Last Name, Address, Phone Number (CK)
- **Employees:** Staff Number (PK), First Name, Last Name, Address, Salary, Phone Number (CK)
- **Service:** Service ID (PK), Day, Start time, Duration, Comment
- **Equipment:** Equipment ID (PK), Description, Usage Details, Cost

F. GENERATE THE ER DIAGRAM FOR THE CONCEPTUAL LEVEL (NO FKS AS ATTRIBUTES).

Conceptual Model | SuperMaids Cleaning Company

Pornkamon Sirivoravijit | Roxana Rodriguez | Sunity Sharma



PART 2: DEVELOP A LOGICAL DATA MODEL BASED ON THE FOLLOWING REQUIREMENTS

A. DERIVE RELATIONS FROM THE CONCEPTUAL MODEL.

1. Strong entity types

- i. **Clients:** ClientID, firstName, lastName, Address, PhoneNumber
 ❑ Primary key – ClientID
- ii. **Employees:** StaffNo, firstName, lastName, Address, Salary, PhoneNumber
 ❑ Primary key – StaffNo
- iii. **Service:** ServiceID, day, date, StartTime, Duration, Comment
 ❑ Primary key – ServiceID
- iv. **Equipment:** EquipmentID, Description, Usage Details, Cost
 ❑ Primary key – EquipmentID

2. Weak entity types

No weak entity types in the conceptual model in this project.

3. One-to-Many (1:*) binary relationship types

- i. Clients to Service
 - **Clients:** ClientID (PK), firstName, lastName, Address, PhoneNumber
 - **Service:** ServiceID (PK), day, date, StartTime, Duration, Comment, ClientNumber
 - Foreign Key - Client Number references Client table (ClientID)
- ii. Employees to Service
 - **Employees:** StaffNo (PK), firstName, lastName, Address, Salary, PhoneNumber
 - **Service:** ServiceID (PK), day, date, StartTime, Duration, Comment, ClientNumber, Staff Number
 - Foreign Key – StaffNo references Employees table (StaffNo)
- iii. Equipment to Service
 - **Equipment:** EquipmentID, Description, Usage Details, Cost

- **Service:** ServiceID (PK), day, date, Start time, Duration, Comment, ClientNumber, Staff Number, EquipmentID
 - Foreign Key – EquipmentID references Equipment table (EquipmentID)

4. Superclass/Subclass relationship types

Optional Disjoint {Or}

EQUIPMENT AND SERVICE

Equipment (EquipmentID, Description, Usage Details, Cost)

Primary Key: EquipmentID

Service: ServiceID, Start Date, Start time, Duration, Comment, Equipment ID

Primary Key: ServiceID

Foreign Key: EquipmentID **references** Equipment (EquipmentID)

B. VALIDATE THE LOGICAL MODEL USING NORMALIZATION TO 3NF

1. Normalization 1NF (Flattening the UNF Table)

Since tables were determined in the conceptual model, we can represent them in 1NF form

- Client(**clientID**, firstName, lastName, address, phoneNumber)
- Employees(**staffNo**, firstName, lastName, address, salary, phoneNumber)
- Service(**serviceID**, day, date, startTime, duration, comment)
- Equipment(**equipmentID**, description, usage, cost)
- Assignment(**assignmentID**, **staffNo**, hours)
- Requirement(**requirementID**, **serviceID(FK)**, **equipmentID(FK)**)

2. Functional Dependencies

- **clientID** > firstName, lastName, address, phoneNumber
- **staffNo** > firstName, lastName, address, salary, phoneNumber
- **serviceID**, **staffNo** > hours
- **equipmentID** > description, usage, cost
- **serviceID** > day, date, startTime, duration, comment

3. 2NF (Create additional tables according to the identified partial dependencies.)

There were no partial dependencies identified, and as a result, no new tables were created.

- Client(**clientID**, firstName, lastName, address, phoneNumber)
- Employees(**staffNo**, firstName, lastName, address, salary, phoneNumber)
- Service(**serviceID**, day, date, startTime, duration, comment)
- Equipment(**equipmentID**, description, usage, cost)
- Assignment(**assignmentID**, serviceID, staffNo, hours)
- Requirement(**requirementID**, serviceID, equipmentID)

4. 3NF (Eliminating redundant relationships by separating transitive dependencies into distinct tables.)

There were no transitive dependencies identified, and as a result, no new tables were created.

- Client(**clientID**, firstName, lastName, address, phoneNumber)
- Employees(**staffNo**, firstName, lastName, address, salary, phoneNumber)
- Service(**serviceID**, day, date, startTime, duration, comment)
- Equipment(**equipmentID**, description, usage, cost)
- **Assignment(serviceID, staffNo, hours)**
- **Requirement(requirementID, serviceID(FK), equipmentID(FK))**

The Services table maintains a one-to-many association with the Client table, requiring the addition of a single foreign key, clientID.

- Service(**serviceID**, **clientID(FK)**, day, date, startTime, duration, comment)

The tables are in 3NF form now.

C. VALIDATE THE LOGICAL MODEL AGAINST USER TRANSACTIONS

In this section, we validate the SuperMaids Cleaning Company with different entries for each table in the system. We check if The Cardboard Box Company and P. Nutall are real clients of our system. Here, we make sure that the transactions linked to them are valid and meet the constraints.

It also involves confirming that the information is accurate and follows our criteria, so our database is compliant.

Clients

ClientID	firstName	lastName	Address	PhoneNumber
C001	The Cardboard Box	Company	2100 NW 42nd Ave, Miami, FL 33142	123-1212123
C002	P.	Nuttall	6019 SW 50th Miami	109-5545544

Employees

StaffNo	firstName	lastName	Address	Salary	PhoneNumber
ST001	Noah	U	5721 W Flagler St, Miami, FL 33144	50,000.00	111-1112222
ST002	Amila	L	3500 NW 37th Ave, Miami, FL 33142	40,000.00	230-2393291

Equipment

EquipmentID	Description	Usage Details	Cost
E001	Mop		\$20.00
E002	Vacuum Cleaner		\$200.00
E003	Clorox		\$10.00
E004	Trash Bag		\$10.00

Service

ServiceID	Day	Date	StartTime	Duration (hour)	Comment	ClientID
S001	Monday	11/27/2023	7am	2	Clean the office	C001
S002	Tuesday	11/28/2023	7am	2	Clean the office	C001
S003	Wednesday	11/29/2023	7am	2	Clean the office	C001
S004	Thursday	11/30/2023	7am	2	Clean the office	C001
S005	Friday	12/1/2023	7am	2	Clean the office	C001
S006	Monday	11/27/2023	5pm	2	Clean the office	C001
S007	Tuesday	11/28/2023	5pm	2	Clean the office	C001
S008	Wednesday	11/29/2023	5pm	2	Clean the office	C001
S009	Thursday	11/30/2023	5pm	2	Clean the office	C001
S010	Friday	12/1/2023	5pm	2	Clean the office	C001
S011	Wednesday	11/29/2023	10am	3	Clean the house	C002

Requirement

Requirement ID	Service ID	Equipment ID
1	S001	E001
2	S001	E002
3	S001	E003
4	S002	E001
5	S002	E002

Assignment

Assignment ID	Service ID	StaffNumber	Hours
1	S001	ST001	2
2	S001	ST002	2
3	S002	ST001	2
4	S002	ST002	2
5	S003	ST001	2

D. DEFINE INTEGRITY CONSTRAINTS

- Primary key constraints

Entity	Column	Constrain	Comment
Client	ClientID	NOT NULL, UNIQUE, PRIMARY KEY NONCLUSTERED	For the primary key each tuple will be storage unique, not allowing null values and the index structure is separate from the actual data rows with the non-clustered constrain.
Service	ServiceID	NOT NULL, UNIQUE, PRIMARY KEY NONCLUSTERED	
Equipment	EquipmentID	NOT NULL, UNIQUE, PRIMARY KEY NONCLUSTERED	
Requirement	RequirementID	IDENTITY (1,1), NOT NULL, UNIQUE, AUTO_INCREMENT, PRIMARY KEY NONCLUSTERED	For the primary key each tuple will be integer and auto increment, and the index structure is separate from the actual data rows with the non-clustered constrain.
Assignment	AssignmentID	IDENTITY (1,1), NOT NULL, UNIQUE, AUTO_INCREMENT, PRIMARY KEY NONCLUSTERED	

- Referential integrity/Foreign key constraints.

SERVICE

Service (ServiceID, Date, StartTime, Duration, Comment, ClientID)

Primary Key (Service ID)

Foreign Key Client ID **references** Client(Client ID) ON UPDATE CASCADE ON DELETE NO ACTION

REQUIREMENT

Requirement(RequirementID, ServiceID, EquipmentID)

Primary Key (Requirement ID)

Foreign Key Service ID **references** Service(Service ID) ON UPDATE CASCADE ON DELETE NO ACTION

Foreign Key EquipmentID **references** Equipment(EquipmentID) ON UPDATE CASCADE ON DELETE SET NULL

ASSIGNMENT

Assignment(Assignment ID, Service ID, StaffNo, Hours)

Primary Key (Assignment ID)

Foreign Key ServiceID references Service(ServiceID) ON UPDATE CASCADE ON DELETE NO ACTION

Foreign Key StaffNo references Employees(StaffNo) ON UPDATE CASCADE ON DELETE NO ACTION

- **Alternate key constraints (if any).**

Entity	Column(s)	Required data	Comment
Client	First Name, LastName, Address, PhoneNumber	Not null	Since First Name , Last Name, Address and Phone Number will serve as candidate key, this should not be null.
Employees	First Name, LastName, Address, Salary,PhoneNumber	Not null	Since First Name , Last Name, Address , Salary, Phone Number will serve as candidate key, this should not be null.

- **Required data.**

Primary keys and foreign keys will be omitted from the following, as they have their own section to determinate their constraints.

Entity	Column(s)	Required data	Comment
Client	First Name, LastName, Address, PhoneNumber	Not null	
Service	Date, Day, StartTime, Duration	Not null	Comment can be null on this table.
Employees	First Name, LName, Address, Salary,PhoneNumber	Not null	
Requirement	ServiceID, EquipmentID	Not null	
Equipment	Cost	Not null	Description and Usage Details can be null
Assignment	StaffNo, Hours	Not null	

- **Attribute domain constraints.**
 1. **Client: Phone Number** <=15
 2. **Service: Duration** > 0

3. Employees: PhoneNumber <=15, Salary > 0
4. Equipment: Cost > 0
5. Assignment: Hours <20

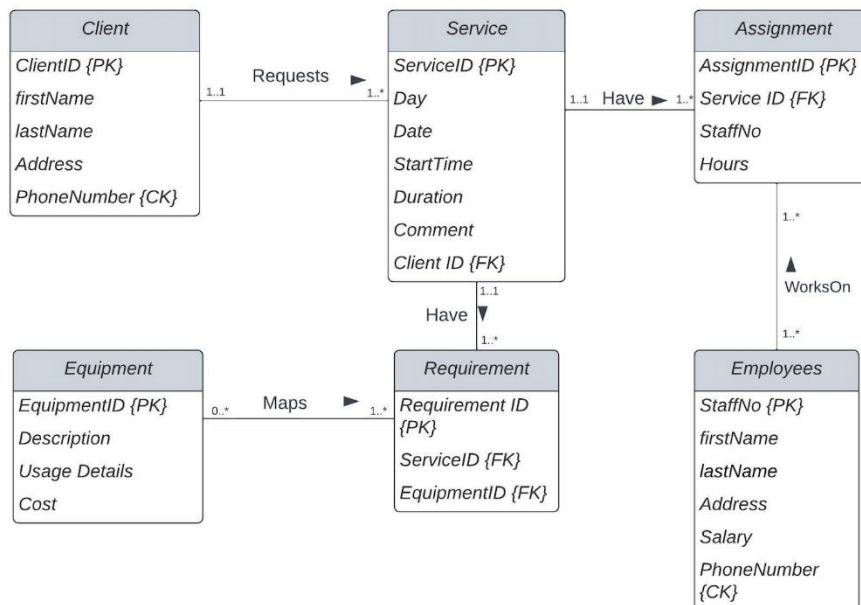
- **General constraints (if any)**

1. The start time should be later than the current time.
2. Telephone numbers should not exceed 15 digits in length.
3. Cleaning services will not be available on weekends – Saturday and Sunday.
4. Days of the week are represented by letters.
5. Phone numbers in Client and Employee Tables have to be unique and can't be null.
6. When a company is the client, its name will be in the "First Name" column, and the word "Company" will be in the last column.
7. The hours from assignment have a limit of less than 20 per day.

E. GENERATE THE ER-DIAGRAM FOR THE LOGICAL LEVEL (CONTAINS FK AS ATTRIBUTES)

Logical Model | SuperMaids Cleaning Company

Pornkamon Sirivoravijit | Roxana Rodriguez | Sunity Sharma



PART 3: TRANSLATE THE LOGICAL DATA MODEL FOR THE DBMS.

A. DEVELOP SQL CODE TO CREATE THE ENTIRE DATABASE SCHEMA, REFLECTING THE CONSTRAINTS IDENTIFIED IN PREVIOUS STEPS.

```
queryClient = ""
CREATE TABLE Client(
    clientID INT PRIMARY KEY NOT NULL,
    firstName VARCHAR(255) NOT NULL,
    lastName VARCHAR(255) NOT NULL,
    address VARCHAR(255) NOT NULL,
    phoneNumber INTEGER NOT NULL CHECK (phoneNumber >= 0 AND LENGTH(CAST(phoneNumber AS TEXT)) <= 15)
);
""
```

```
queryEmployees = ""
CREATE TABLE Employees(
    staffNo INT PRIMARY KEY NOT NULL,
    firstName VARCHAR(255) NOT NULL,
    lastName VARCHAR(255) NOT NULL,
    address VARCHAR(255) NOT NULL,
    salary REAL NOT NULL CHECK (salary >0),
    phoneNumber INTEGER NOT NULL CHECK (phoneNumber >= 0 AND LENGTH(CAST(phoneNumber AS TEXT)) <= 15)
);
""
```

```
queryRequirement = ""
CREATE TABLE Requirement(
    requirementID INT IDENTITY(1,1) PRIMARY KEY,
    serviceID INT REFERENCES Service(serviceID) ON UPDATE CASCADE ON DELETE NO ACTION,
    equipmentID INT REFERENCES Service(serviceID) ON UPDATE CASCADE ON DELETE SET NULL
);
""
```

```
create_service = ""
CREATE TABLE Service(
    serviceID INT PRIMARY KEY,
    clientID INT REFERENCES Client(clientID) ON UPDATE CASCADE ON DELETE NO ACTION,
    day VARCHAR(10) NOT NULL,
    date DATE NOT NULL,
    startTime TIME,
    duration INT CHECK (duration > 0) NOT NULL,
    comment VARCHAR(255),
    CONSTRAINT fk_client
        FOREIGN KEY (clientID)
        REFERENCES Client (clientID)
        ON UPDATE CASCADE ON DELETE NO ACTION
);
""
```

```
create_equipment = ""
CREATE TABLE Equipment(
    equipmentID INT PRIMARY KEY,
    description VARCHAR(255),
    usage VARCHAR(255),
    cost DECIMAL(10, 2) CHECK (cost >0) NOT NULL
);
""
```

```
# Assignment
create_assignment = ""
CREATE TABLE Assignment(
    assignmentID INTEGER PRIMARY KEY,
    serviceID INT REFERENCES Service (serviceID) ON UPDATE CASCADE ON DELETE NO ACTION,
    staffNo INT REFERENCES Employees (staffNo) ON UPDATE CASCADE ON DELETE NO ACTION NOT NULL,
    hours INT CHECK (hours < 20) NOT NULL);
""
```


B. CREATE AT LEAST 5 TUPLES FOR EACH RELATION IN YOUR DATABASE

Service table:

	serviceID	clientID	...	duration		comment
0	S001	C001	...	2		Regular cleaning session
1	S002	C001	...	2		Regular cleaning session
2	S003	C001	...	2		Regular cleaning session
3	S004	C001	...	2		Regular cleaning session
4	S005	C001	...	2		Regular cleaning session
5	S006	C001	...	2		Regular cleaning session
6	S007	C001	...	2		Regular cleaning session
7	S008	C001	...	2		Regular cleaning session
8	S009	C001	...	2		Regular cleaning session
9	S010	C001	...	2		Regular cleaning session
10	S011	C002	...	1		Deep cleaning appointment
11	S012	C003	...	3	Specialized cleaning and disinfection	
12	S013	C004	...	2		Carpet Cleaning
13	S014	C005	...	1		Kitchen Disinfection

[14 rows x 7 columns]

Reservation table:

	equipmentID	description	usage	cost
0	E001	Cleaning supplies	Disinfection	2000
1	E002	Vacuum cleaner	Cleaning	500
2	E003	Sanitizing solution	Disinfection	800
3	E004	Broom and mop set	Cleaning	300
4	E005	Trash bags	Waste disposal	50

Assignment table:

	assignmentID	serviceID	staffNo	hours
0	1	S001	ST001	2
1	2	S001	ST002	2
2	3	S002	ST001	2
3	4	S002	ST002	2
4	5	S003	ST001	2
5	6	S003	ST002	2
6	7	S004	ST001	2
7	8	S004	ST002	2
8	9	S005	ST001	2
9	10	S005	ST002	2
10	11	S006	ST001	2
11	12	S006	ST002	2
12	13	S007	ST001	2
13	14	S007	ST002	2
14	15	S008	ST001	2
15	16	S008	ST002	2
16	17	S009	ST001	2
17	18	S009	ST002	2
18	19	S010	ST001	2
19	20	S010	ST002	2
20	21	S011	ST003	1
21	22	S012	ST003	1
22	23	S013	ST004	1
23	24	S014	ST005	1

```

ALL ROWS IN TABLE CLIENT:
  clientID firstName  lastName      address  phoneNumber
0   C001      John      Doe    123 Main St  555-1234567
1   C002      Jane      Smith   456 Oak St  555-7654321
2   C003      Bob       Johnson  789 Pine St  555-9876543
3   C004      Alice     Williams 101 Elm St  555-2345678
4   C005      Charlie    Smith   222 Maple St 555-8765432
Index(['clientID', 'firstName', 'lastName', 'address', 'phoneNumber'], dtype='object')
ALL ROWS IN TABLE Employees:
  staffNo firstName  lastName      address  salary  phoneNumber
0   ST001      Alice  Johnson    789 Elm St  50000.0  555-8765432
1   ST002      Bob   Williams   101 Pine St  60000.0  555-1234567
2   ST003      Charlie Smith    222 Maple St 55000.0  555-2345678
3   ST004      David  Miller    333 Cedar St 70000.0  555-3456789
4   ST005      Eva    Jones     444 Oak St  75000.0  555-4567890
Index(['staffNo', 'firstName', 'lastName', 'address', 'salary', 'phoneNumber'], dtype='object')
ALL ROWS IN TABLE Requirement:
  requirementID serviceID equipmentID
0              1      S001      E001
1              2      S002      E002
2              3      S003      E003
3              4      S004      E004
4              5      S005      E005
Index(['requirementID', 'serviceID', 'equipmentID'], dtype='object')

```

C. DEVELOP 5 SQL QUERIES USING EMBEDDED SQL

```

New salary increased by 20%:
  staffNo firstName  lastName      address  salary  phoneNumber
0   ST001      Alice  Johnson    789 Elm St  60000.0  555-8765432
1   ST002      Bob   Williams   101 Pine St  72000.0  555-1234567
2   ST003      Charlie Smith    222 Maple St  66000.0  555-2345678
3   ST004      David  Miller    333 Cedar St  84000.0  555-3456789
4   ST005      Eva    Jones     444 Oak St  90000.0  555-4567890

Service Table with new update on 1 hour extra work on each transaction
  serviceID clientID  ... duration      comment
0      S001      C001  ...      3      Regular cleaning session
1      S002      C001  ...      3      Regular cleaning session
2      S003      C001  ...      3      Regular cleaning session
3      S004      C001  ...      3      Regular cleaning session
4      S005      C001  ...      3      Regular cleaning session
5      S006      C001  ...      3      Regular cleaning session
6      S007      C001  ...      3      Regular cleaning session
7      S008      C001  ...      3      Regular cleaning session
8      S009      C001  ...      3      Regular cleaning session
9      S010      C001  ...      3      Regular cleaning session
10     S011      C002  ...      2      Deep cleaning appointment
11     S012      C003  ...      4      Specialized cleaning and disinfection
12     S013      C004  ...      3      Carpet Cleaning
13     S014      C005  ...      2      Kitchen Disinfection

[14 rows x 7 columns]

```

```

Employee Schedule Table:
employee_identified employee_name ... client_name client_address
0 ST001 Alice Johnson ... John Doe 123 Main St
1 ST002 Bob Williams ... John Doe 123 Main St
2 ST001 Alice Johnson ... John Doe 123 Main St
3 ST002 Bob Williams ... John Doe 123 Main St
4 ST001 Alice Johnson ... John Doe 123 Main St
5 ST002 Bob Williams ... John Doe 123 Main St
6 ST001 Alice Johnson ... John Doe 123 Main St
7 ST002 Bob Williams ... John Doe 123 Main St
8 ST001 Alice Johnson ... John Doe 123 Main St
9 ST002 Bob Williams ... John Doe 123 Main St
10 ST001 Alice Johnson ... John Doe 123 Main St
11 ST002 Bob Williams ... John Doe 123 Main St
12 ST001 Alice Johnson ... John Doe 123 Main St
13 ST002 Bob Williams ... John Doe 123 Main St
14 ST001 Alice Johnson ... John Doe 123 Main St
15 ST002 Bob Williams ... John Doe 123 Main St
16 ST003 Charlie Smith ... Jane Smith 456 Oak St
17 ST003 Charlie Smith ... Bob Johnson 789 Pine St
18 ST004 David Miller ... Alice Williams 101 Elm St
19 ST005 Eva Jones ... Charlie Smith 222 Maple St

[20 rows x 7 columns]

```

```

Service Table with new update on removal transaction of S002:
serviceID clientID ... duration comment
0 S001 C001 ... 3 Regular cleaning session
1 S003 C001 ... 3 Regular cleaning session
2 S004 C001 ... 3 Regular cleaning session
3 S005 C001 ... 3 Regular cleaning session
4 S006 C001 ... 3 Regular cleaning session
5 S007 C001 ... 3 Regular cleaning session
6 S008 C001 ... 3 Regular cleaning session
7 S009 C001 ... 3 Regular cleaning session
8 S010 C001 ... 3 Regular cleaning session
9 S011 C002 ... 2 Deep cleaning appointment
10 S012 C003 ... 4 Specialized cleaning and disinfection
11 S013 C004 ... 3 Carpet Cleaning
12 S014 C005 ... 2 Kitchen Disinfection

[13 rows x 7 columns]

```

```

Client Table with email attribute:
clientID firstName lastName address phoneNumber email
0 C001 John Doe 123 Main St 555-1234567 None
1 C002 Jane Smith 456 Oak St 555-7654321 None
2 C003 Bob Johnson 789 Pine St 555-9876543 None
3 C004 Alice Williams 101 Elm St 555-2345678 None
4 C005 Charlie Smith 222 Maple St 555-8765432 None
5 C006 Ana Cruz 125 Main St 565-1234567 ana.crz@gmail.com

```

D. GITHUB REPOSITORY AND DOCUMENTATION DETAILS

Our Project documentation and python code is uploaded at this repository address:

<https://github.com/sunitysharma/SuperMaids-Cleaning-Company/>.