COURSE : DATABASE SYSTEMS AND SECURITY (7CI022)

**STUDENT ID:**

**OLOYEDE FESTUS OYEBISI / 2345336**

Table of contents

[INTRODUCTION 3](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051841)

[Scenario 3](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051842)

[Business Rules 3](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051843)

[Information Needs 4](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051844)

[Data Modelling 5](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051845)

[ENTITY-RELATIONSHIP (ER) MODEL 6](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051846)

[Entities & Attributes 6](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051847)

[Relationships 7](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051848)

[Logical Model 9](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051849)

[Relational Model 10](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051850)

[NORMALIZATION 11](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051851)

[Normalization Process 11-15](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051852)

[Table Creation and Population 15](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051853)

[Creating tables 16-17](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051854)

[Populating tables 17-20](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051855)

[Querying Database 20-25](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051856)

[Security 22-24](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051858)

[Integrity 23-25](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051859)

[Data Ethics 25](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051860)

[CONCLUSION 26](file:///C:\Users\festu\Downloads\database%20dummy%20report(1).docx#_Toc139051861)

**INTRODUCTION**

SCENARIO

This report is based on the design of an information system for a hotel (hotel reservation system) named “FLOW HOTEL AND SUITES”. This system facilitates the process of booking and managing hotel accommodations. It allows guests to search for available rooms, make reservations, and sometimes customize their stay with specific preferences.

Additionally, this information system assists the managers, the owners/executives, and shareholders of the hotel to keep track of the income made by the company over a period. In other words, this encourages transparency. It also provides tools to manage room inventory, track reservations, manage guest records, handle check-ins and check-outs, process payments, and generate reviews.

Generally, this system will streamline the booking process for both the hotel guests and staffs, thereby enhancing efficiency and improving the overall guest experience. This will undoubtedly help the organisation to perform financial, administrative, and operational analysis, thereby helping in making key decisions that aid in the advancements of the company.

**BUSINESS RULES**

Pinpointed business rules may be defined as follows:

* Each guest makes one or more reservations and each reservation is made by a guest, but not all guests make reservations.
* Each guest may or may not leave one or more reviews after their stay
* Each room accommodates one or more guests.
* For each reservation, a deposit or full payment must be made.

**INFORMATION NEEDS**

There are different levels of information needed by the stakeholders in the organisation/establishment.

OPERATIONAL

* Guests need to know the type of rooms available, so they can make reservations.
* Guests need to know the cost of rooms they are reserving.
* The organisation needs information on all guests making reservations and the type of room that is commonly booked to draw meaningful insights.
* The organisation needs the biodata of all reviews made to know where to improve and maintain.
* The organisation needs information on all reservations made, check-in and check-out dates.

ADMINISTRATIVE

* The organisation needs information of guests and reservations made for reference purposes and to efficiently manage bookings and room allocations.
* The organisation needs information of all guests checking in, along with other occupants during their stay.

ACCOUNTING AND BUDGETING

* The organization needs information on all payments made and respective payment methods for budgeting and transparency.
* The organisation needs information on all reservation deposits made and balance to be made available before check-in.

**DATA MODELLING**

This information system is designed to record and accommodate the information needs of all stakeholders of the organization by the provision of relevant data, in line with the business rules that has been defined. This information system would be designed with the following entities and attributes:

1. **GUEST:** This entity provides information about each guest coming into the hotel. Attributes include “**GUEST\_ID”** (guest unique identifier/primary key), NAME (name of guest), ADDRESS (address of guest), GENDER (gender of guest), POSTCODE (postcode of guest), EMAIL\_ADDRESS (email address of guest), MOBILE\_NUMBER (mobile number of guest), and ROOMS\_ID (rooms unique identifier).
2. **ROOM:** This entity provides info on the rooms the hotel has to offer. Attributes include “**ROOM\_ID**” (room unique identifier/primary key), ROOM\_TYPE (type of room on offer), ROOM\_COST\_PER\_NIGHT (cost of the room per night), and MAXIMUM\_OCCUPANCY (maximum number of occupants available).
3. **RESERVATION**: This entity provides information on the reservations made by customers. Attributes include “**RESERVATION\_ID**” (reservation unique identifier/primary key), CHECK\_IN\_DATE (proposed check in date for the reservation), CHECK\_OUT\_DATE (proposed check out date for the reservation), RESERVATION\_COST (cost of the reservation), and GUEST\_ID (guest unique identifier).
4. **PAYMENT**: This entity provides information on the payments made by the customers. Attributes include “**PAYMENT\_ID**” (payment unique identifier/primary key), PAYMENT\_DATE (date payment was made), PAYMENT\_METHOD (method of payment), PAYMENT\_AMOUNT (amount of that particular payment), and RESERVATION\_ID (reservation unique identifier).
5. **REVIEW**: This entity provides information on the reviews left by customers after leaving. Attributes include “**REVIEW\_ID” (**review unique identifier/primary key), RATINGS (ratings on the reviews), COMMENTS (comments left in the reviews), and GUEST\_ID (guest unique identifier).

**ENTITY-RELATIONSHIP (ER) MODEL**

An Entity-Relationship model is a diagram which explains and shows the logical structure of a database. The ER model displays the relationships between entities in a certain database. The main components of an ER model are Entities, Relationships, and Attributes.

**ENTITIES & ATTRIBUTES**

The entities of this database have been defined as follows (primary keys are boldened):

1. Guest: **GUEST\_ID**, NAME, ADDRESS, POSTCODE, GENDER, EMAIL\_ADDRESS, MOBILE\_NUMBER and ROOM\_ID. The foreign key here is ROOMS\_ID.
2. Room: **ROOM\_ID**, ROOM\_TYPE, ROOM\_COST\_PER\_NIGHT, and MAXIMUM\_OCCUPANCY.
3. Reservation: **RESERVATION\_ID**, CHECK\_IN\_DATE, CHECK\_OUT\_DATE, RESERVATION\_COST, and GUEST\_ID. The foreign key here is GUEST\_ID.
4. Payment: **PAYMENT\_ID**, PAYMENT\_DATE, PAYMENT\_METHOD, PAYMENT\_AMOUNT, and RESERVATION\_ID. The foreign key here is RESERVATION\_ID.
5. Review: **REVIEW\_ID,** RATINGS, COMMENTS, and GUEST\_ID. The foreign key here is GUEST\_ID.

**RELATIONSHIPS**

These are the associations between two or more entities of a database. The following relationships are defined for this database:

1. Guest and Reservation:

* Relationship: Guest (1 ..1) makes Reservations (0..\*)

Each guest can make zero to many reservations. Each reservation is made by only one guest.

* Cardinality: One to Many (1:N)
* Participation: Source optional & Target mandatory

1. Room and Guest:

* Relationship: Rooms (1..1) can harbour Guests (0..\*).

Each room can harbour zero to multiple guests, and each guest can only be in one room.

* Cardinality: One to Many (1:N)
* Participation: Source optional, Target mandatory.

1. Guest and Review:

* Relationship: Guest (1..1) can make Reviews (0..\*)

Each guest can make zero to many reviews, and each review must be made by a guest.

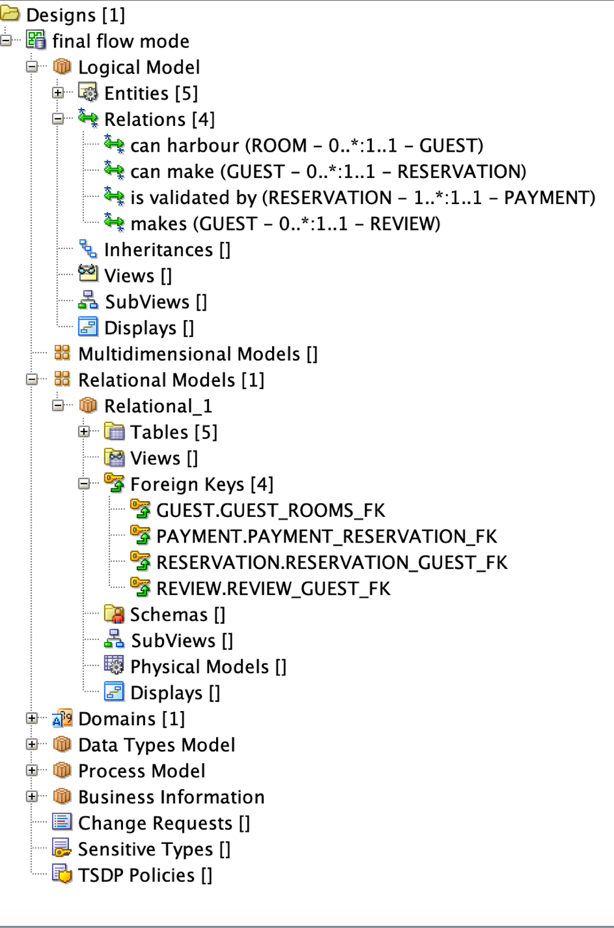
* Cardinality: One to Many (1:N)
* Participation: Source optional, Target Mandatory

1. Reservation and Payment:

* Relationship: Reservation (1..1) must be made with Payments (1..\*)

Each reservation must be made with one or more payments, and each payment must have been actioned by a reservation.

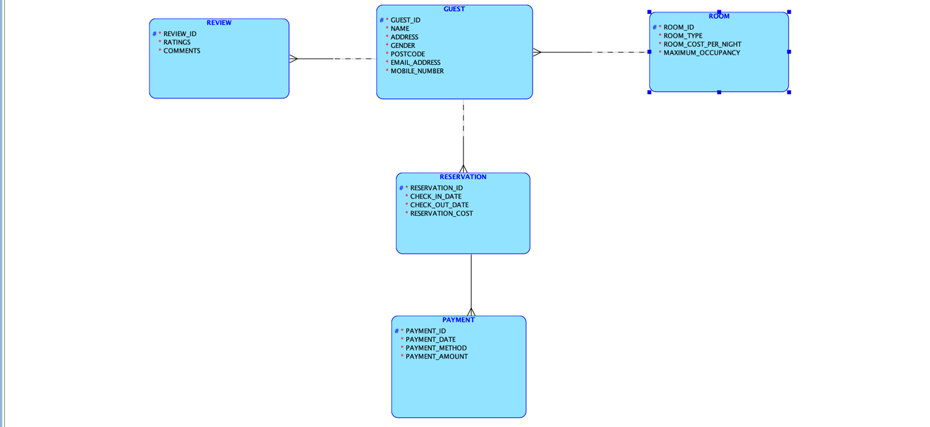
* Cardinality: One to Many (1:N)
* Participation: Source mandatory, Target mandatory.



*Fig 1: The Entity relationship definitions*

**LOGICAL MODEL**

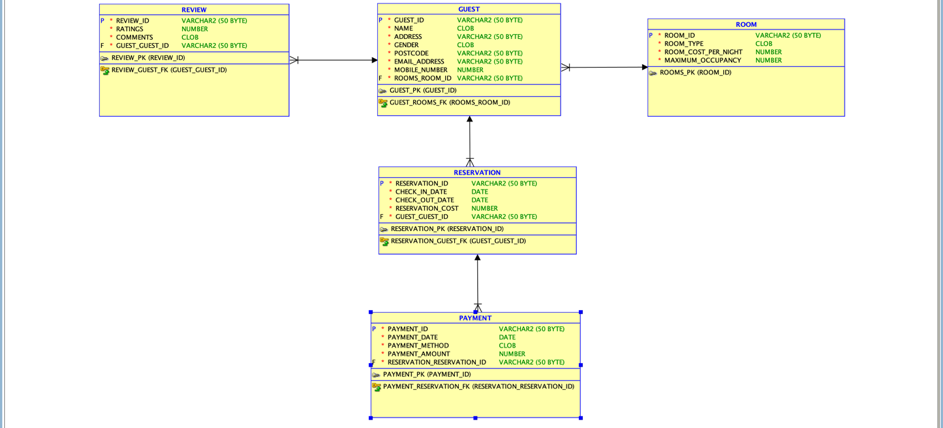
The logical model is built using Oracle SQL Data Modeller by defining entities, attributes and the relationship between attributes. Figure 1 shows this step and Figure 2 shows logical model**.**

****

*Figure 2: Logical ER Model*

**Relational Model**

Oracle SQL Data Modeller automatically engineers a relational model based once the relationships are established on the logical model. Figure 3 shows the automatically engineered relational model.



*Figure 3: Relational ER Model*

**NORMALIZATION**

Normalization refers to the process of organizing data in a database efficiently. It involves breaking down large tables into smaller ones and defining relationships between them to reduce redundancy.

Figure 4 shows a dummy dataset for Guest’s reservations in an unnormalized form (UNF). I would attempt to normalize this relation to 3NF.

**NORMALIZATION PROCESS**

This describes the steps taken to normalize my dataset from Unnormalized form to Third normal form in relation to FLOW HOTEL’S information system:

1. **Unnormalized Form (UNF):** This refers to a context where tables or relations have not been structured to eliminate redundancy and dependency issues. This means the table contains repeating groups i.e., there is more than one value at the intersection of each row and column. Figure 4 below shows my dummy data is not unnormalized because it contains no repeating groups. Upon further analysis of the relation table, I have identified **PAYMENT\_ID** as the key attribute because it fully functionally determines every attribute in the relation. I will label this relation (entity) **PAYMENT**.



*Figure 4: UNF/1NF/2NF PAYMENT DATA TABLE*

1. **1st Normal Form (1NF):** For a table or set of data to be in 1st Normal Form, it means it does not contain any repeating groups. This can be seen in figure 4 above, where my data table does not contain any repeating groups, with the primary key **PAYMENT\_ID**, uniquely identifying each row in the table.
2. **2nd Normal Form (2NF)**: For a table to be in 2NF, the table must be in 1NF and contain no partial dependencies. Partial dependencies between entities must have been reduced. This means that the relation must also have a primary key of a single attribute. My primary key for this table is **PAYMENT\_ID,** as it fully functionally determines every other attribute, and as seen in figure 4, this is functional dependency definition for the **PAYMENT** relation:

**PAYMENT\_ID ⇒ RESERVATION\_ID, GUEST\_ID, NAME, ROOM\_ID, ROOM\_TYPE, ROOM\_COST, MAXIMUM\_OCCUPANCY, CHECK\_IN\_DATE, CHECK\_OUT\_DATE, RESERVATION\_COST, PAYMENT\_DATE, PAYMENT\_METHOD, PAYMENT\_AMOUNT, ADDRESS, POSTCODE, GENDER, EMAIL\_ADDRESS, MOBILE\_NUMBER, REVIEW\_ID, RATINGS, COMMENTS**.

1. **3rd Normal Form (3NF):** A table is in 3NF if it is in 2NF, and has no transitive dependencies for non-prime attributes. The main goal of 3NF is to eliminate redundancy and prevent data anomalies by ensuring every non-prime attribute is not only fully functionally dependent on the primary key, but also directly dependent on it. Transitive dependencies which exist in the 2NF PAYMENT relation include:

Transitive Dependency 1: **RESERVATION\_ID ⇒ GUEST\_ID, CHECK\_IN\_DATE, CHECK\_OUT\_DATE, RESERVATION\_COST**.

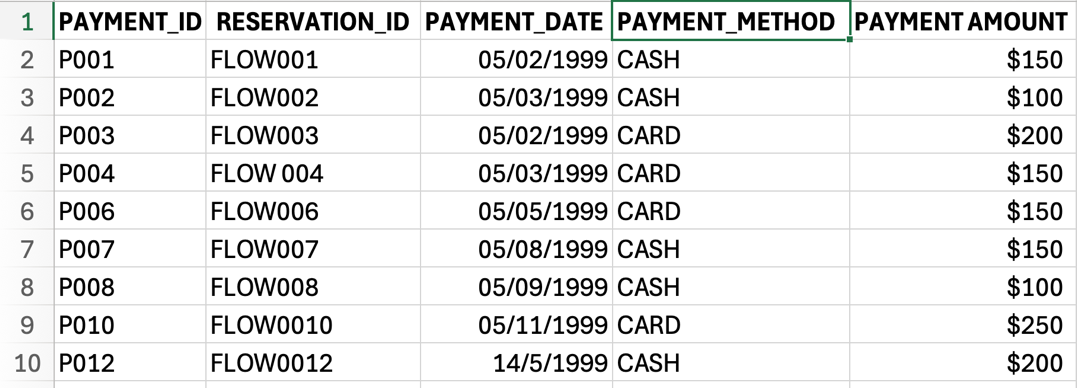
Transitive Dependency 2: **ROOM\_ID** ⇒ **ROOM\_TYPE, ROOM\_COST\_PER\_NIGHT, MAXIMUM\_OCCUPANCY.**

Transitive Dependency 3: **REVIEW\_ID ⇒ GUEST\_ID, RATINGS, COMMENTS**.

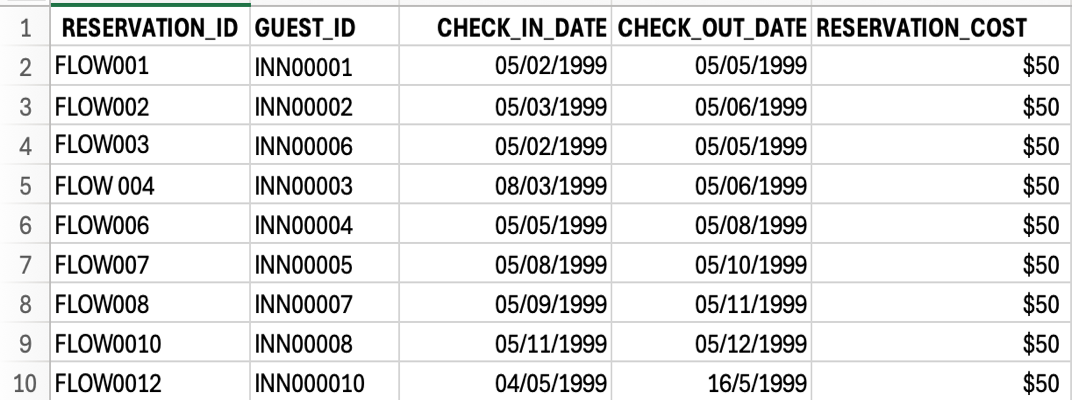
Transitive Dependency 4: **GUEST\_ID ⇒ NAME, ROOM\_ID, ADDRESS, POSTCODE, GENDER, EMAIL\_ADDRESS, MOBILE\_NUMBER**.

To remove these transitive dependencies from the **PAYMENT** relation, transitive attributes must be eliminated from the relation. Foreign keys then emerge from the parent **PAYMENT** relation, and TD1, TD2, TD3, TD4 can be used to create the **RESERVATION, ROOM, REVIEW**, and **GUEST** entities respectively. **ROOM\_ID** emerges as a foreign key in the **GUEST** entity, and **GUEST\_ID** emerges as a foreign key in the **RESERVATION** and **REVIEW** entities. RESERVATION\_ID is also foreign key in the final **PAYMENT** entity. This leaves our final 3NF **PAYMENT** TABLE as;  
  
**PAYMENT\_ID ⇒ PAYMENT\_DATE, PAYMENT\_METHOD, PAYMENT\_AMOUNT, RESERVATION\_ID.**

After this, the data base and the individual relations are now in 3NF as shown in the figures and tables below:



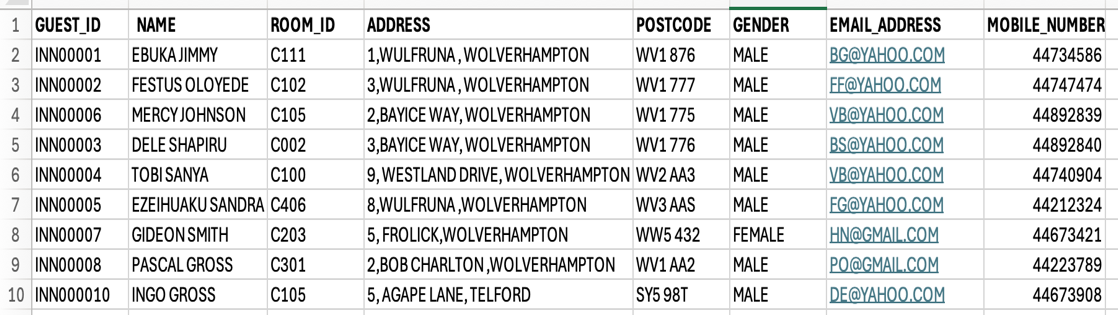
*Figure 5: 3NF PAYMENT TABLE*



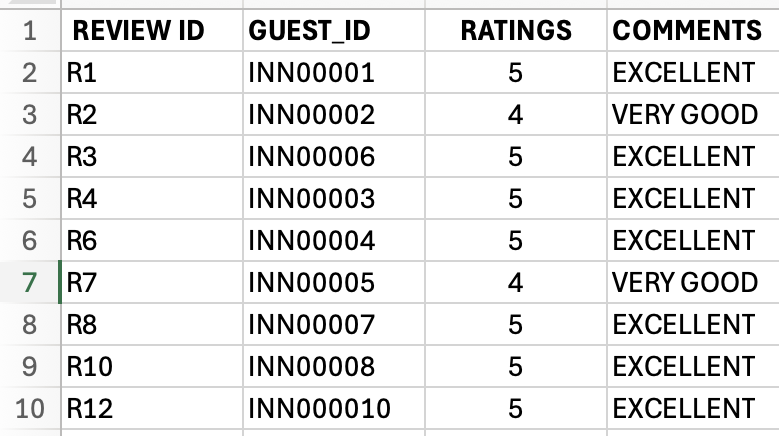
*Figure 6: 3NF RESERVATION TABLE*



*Figure 7: 3NF ROOM TABLE*



*Figure 8: 3NF GUEST TABLE*



*Figure 9: 3NF REVIEW TABLE*

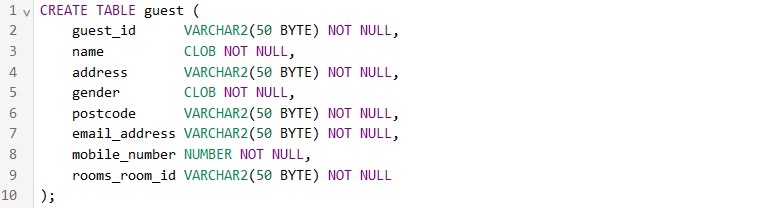
This [DUMMY DATA SHEET](https://1drv.ms/x/c/baad88a9112db246/ETLNsXxQjxZFpQtomBzd2pEBxMFcDE_hV3xLPNJp0U_gjg) shows the relations and transformations after each normalization process from UNF to 1NF to 2NF, and also 3NF for better understanding.

**TABLE CREATION AND POPULATION**

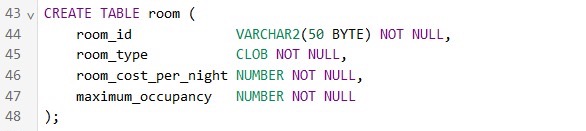
The database is created and populated using Oracle Live SQL. Five tables were created for GUEST, ROOM, RESERVATION, PAYMENT, and REVIEW. The SQL statements used in creating, altering, and populating these tables are shown in the following sections.

**CREATING TABLES**

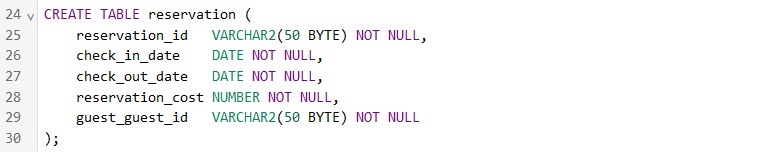
The GUEST, ROOM, RESERVATION, PAYMENT, and REVIEW tables are created using CREATE TABLE statements, as shown in the figures below:



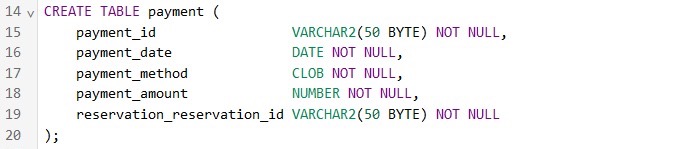
*Figure 10: CREATE TABLE GUEST STATEMENT*

****

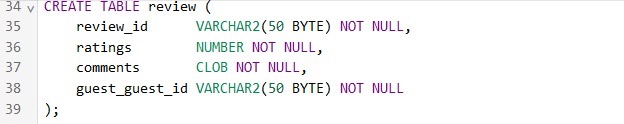
*Figure 11: CREATE TABLE ROOM STATEMENT*



*Figure 12: CREATE TABLE RESERVATION STATEMENT*



*Figure 13: CREATE TABLE PAYMENT STATEMENT*



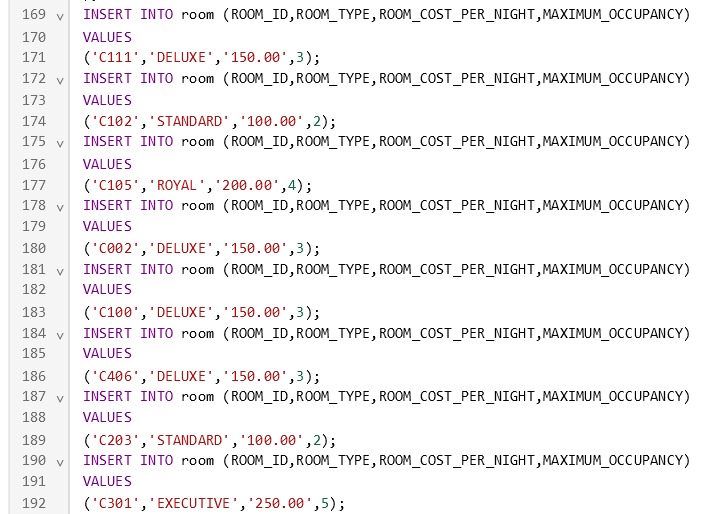
*Figure 14: CREATE TABLE REVIEW STATEMENT*

**POPULATING TABLES**

Dummy values are inserted into the GUEST, ROOM, RESERVATION, PAYMENT, and REVIEW tables respectively using INSERT statements, as shown in the figures below:



*Figure 15: INSERT INTO GUEST*



*Figure 16: INSERT INTO ROOM*



*Figure 17: INSERT INTO RESERVATION*

****

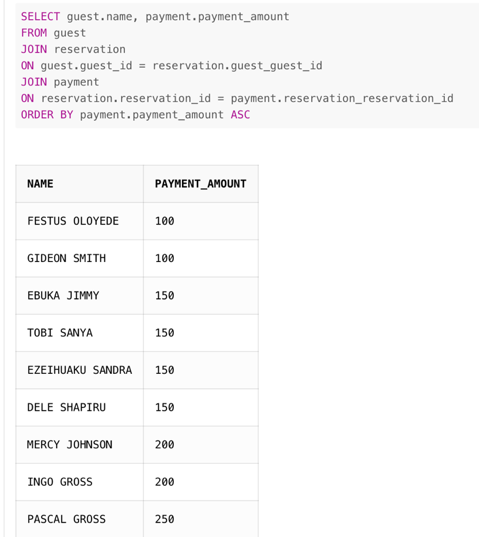
*Figure 18: INSERT INTO PAYMENT*



*Figure 19: INSERT INTO REVIEW*

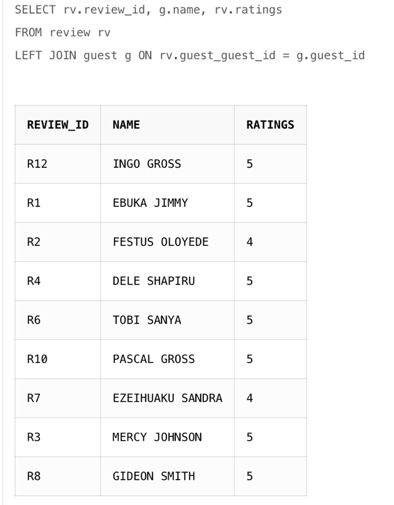
**QUERYING DATABASE**

**QUERY 1:** This query below selects payment amount of each guest in ascending order.



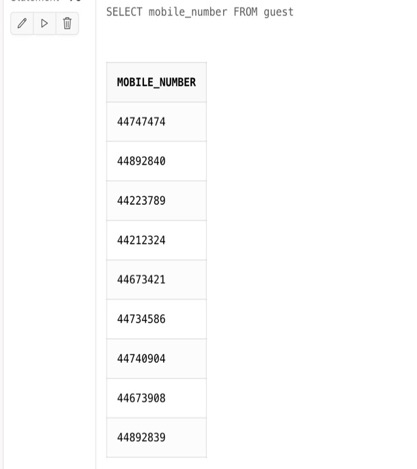
*Figure 20: Querying database 1*

**QUERY 2:** This query shows a list of reviews with the corresponding guest name and rating using a left join between the review and guest tables.



*Figure 21: Querying database 2*

**QUERY 3:**  This query shows the mobile numbers of guests in the hotel.



*Figure 22: Querying database 3*

**SECURITY, INTEGRITY, AND ETHICAL ASPECTS OF DATA GOVERNANCE**

**SECURITY**

Data security is very critical and essential to any organization or business, as a robust data security management and strategy process enables an organization to protect its information and personal data against cyberattacks. It also helps them reduce the risk of human error and insider threats, which continue to be the cause of many data breaches.

Every organization should have policies put in place regarding data access and protection, because organizations are legally obliged to protect customer data from getting stolen or ending up in the wrong hands. High profile hacks can lead to customers losing trust in the business and fines/legal payments to repair damages.

Some general benefits of data security include:

* Keeps the reputation of the company/organization clean, as a great reputation is very important in the business world.
* Gives the organization a competitive edge, as this can easily set them apart from other competitors who may be struggling to do the same.
* Keeps the personal information and records collected by the company safe, secure, and well-protected.

**DATA SECURITY IN RELATION TO FLOW HOTEL AND SUITES**

Flow Hotel and Suites is a hotel organization that collects different types of personal information from guests, therefore data security practice is paramount to avoid any type of hacks. Clear policies have been put in place regarding data access and our staff members get trained in secure data handling practices.

These are some measures taken by Flow Hotel and Suites to ensure data security in the organization’s database:

* **Every staff gets trained**: Every hotel staff has been provided with the fundamental skills to ensure a secure environment for guests and their personal data. We also conduct regular tests to make sure the employees are keeping abreast with all the learnings and teachings in this aspect.
* **Use of cybersecurity tools**: Security tools such as firewalls, network monitor, and anti-malware have been put in place to protect the hotel against cybersecurity threats.
* **Payment card information encryption**: Strong data encryption methods are employed to make sure card information of our guests are well protected from any type of malicious attacks.

**INTEGRITY**

Data integrity is the accuracy, consistency, and completeness of data as it is maintained over time and across formats. This involves ensuring that your data collected is error-free, and conforms with the integrity controls and methods of regulatory compliance. This is very important in organizations as poor data quality can lead to:

* Poor decision making
* Potential legal issues
* Inadequate customer service

Types of data integrity and constraints include:

* **Domain Integrity**: This refers to the processes that ensure accuracy in each piece of data included in a domain, or a set of acceptable values a column may contain.
* **Entity Integrity**: This involves the creation of primary keys in order to identify data collected as unique entities, to ensure no data is listed more than once or is null. These are basically the rules governing each relation and their formats. The primary keys in Flow Hotel and Suites database are GUEST\_ID, PAYMENT\_ID, RESERVATION\_ID, ROOM\_ID, and REVIEW\_ID.



*Figure 23: GUEST ENTITY CONSTRAINT*



*Figure 24: ROOM TABLE CONSTRAINT*



*Figure 25: RESERVATION TABLE CONSTRAINT*



*Figure 26: PAYMENT TABLE CONSTRAINT*



*Figure 27: REVIEW TABLE CONSTRAINT*

* **Referential Integrity**: This is the series of processes used to store and access data uniformly, allowing rules to be embedded into the database’s structure regarding the use of foreign keys. This ensures and oversees the logical dependency of a foreign key on a primary key.

In relation to Flow Hotel and Suites, we take Integrity seriously, as this allows our guests to correctly register their information. We ensure that data entered into a row or column reflects the standard allowable value for that domain, row, or column. For example, the MOBILE\_NUMBER attribute in the GUEST entity table can only collect digits/numerical entries, and this has been strictly put in place to maintain structure and eliminate any potential violation of the company’s data integrity rules.

**DATA ETHICS**

These are the principles that regulate how an organization collects, protects, and uses data. Organizations must ensure they use data in a consistent manner that’s in line with the company values.

In Flow Hotel and Suites, we have taken the following ethical data use routes, to ensure we offer the best services to our hotel guests:

* **Transparency**: This involves clear communication with our guests on what data would be collected, if it will be stored, and who it might be shared with. This ensures that our guests know exactly how the organization would be using their information.
* **Data Privacy**: Personal information collected from our guests such as email address, home address, and mobile number, which can be linked directly to them are well protected from public exposure. Private policies and security protocol have been enacted to secure these data and to also make our guests aware of how their data is stored and used.
* **Individual Agency & Consent**: This allows individuals to consent to, and make a choice on what personal data should be collected, stored, and made accessible to certain staff.

**CONCLUSION**

We successfully designed an information system for FLOW HOTEL AND SUITES. We defined the scenario, business rules, information needs, and created an appropriate Entity-Relationship (ER) model with proper entity relationships. We also normalized the database to Third normal Form.

The information system was implemented by creating the required entity and constraints for each entity. We also populated each entity with dummy data and queried the tables to draw insights.

In conclusion, we discussed data security, ethics, and integrity using the context of our scenario.