# Project Report Submitted for

**DATABASE MANAGEMENT SYSTEM**

**(UCS310)**

## Submitted by:

**(102153017) Naman Sharma**

**(102153018) Moksh Baweja**

**(102153021) Nalin Singhal**

**(102103613) Aryan Shanker Saxena**

## BE Second Year

## Submitted to :-

## Dr. Nitigya Sambyal



# Computer Science and Engineering Department TIET, Patiala

**Jan-May 2023**

INDEX

|  |  |  |
| --- | --- | --- |
| S.No. | Title | Page No. |
| 1. | Introduction | 3 |
| 2. | Requirement Analysis | 4 |
| 3. | ER-Diagram | 6 |
| 4. | ER to Table | 7 |
| 5. | Normalization | 8 |
| 6. | SQL & PL/SQL Query snapshots | 9 |
| 7. | Conclusion | 16 |
|  |

INTRODUCTION

Due to the rapid expansion of the internet and the tourist sector, more and more people are choosing to book their travel through online channels. A trustworthy, secure, and user-friendly online trip booking database system is therefore greatly needed. A system like this may give clients the tools they need to conveniently search, compare, and book flights, hotels, and other travel-related services in addition to giving travel organisations, hotel owners, tour operators, and administrators the resources they require to run their businesses successfully.

This report's overall goal is to present a thorough grasp of the prerequisites for an online travel booking database system and to emphasise the advantages such system may give for both travellers and travel agencies.

REQUIREMENT ANALYSIS

Requirement analysis for an online traveling booking database system involves identifying the needs and expectations of users, the technical requirements for the system, and the business requirements of the project.

Technical Requirements:

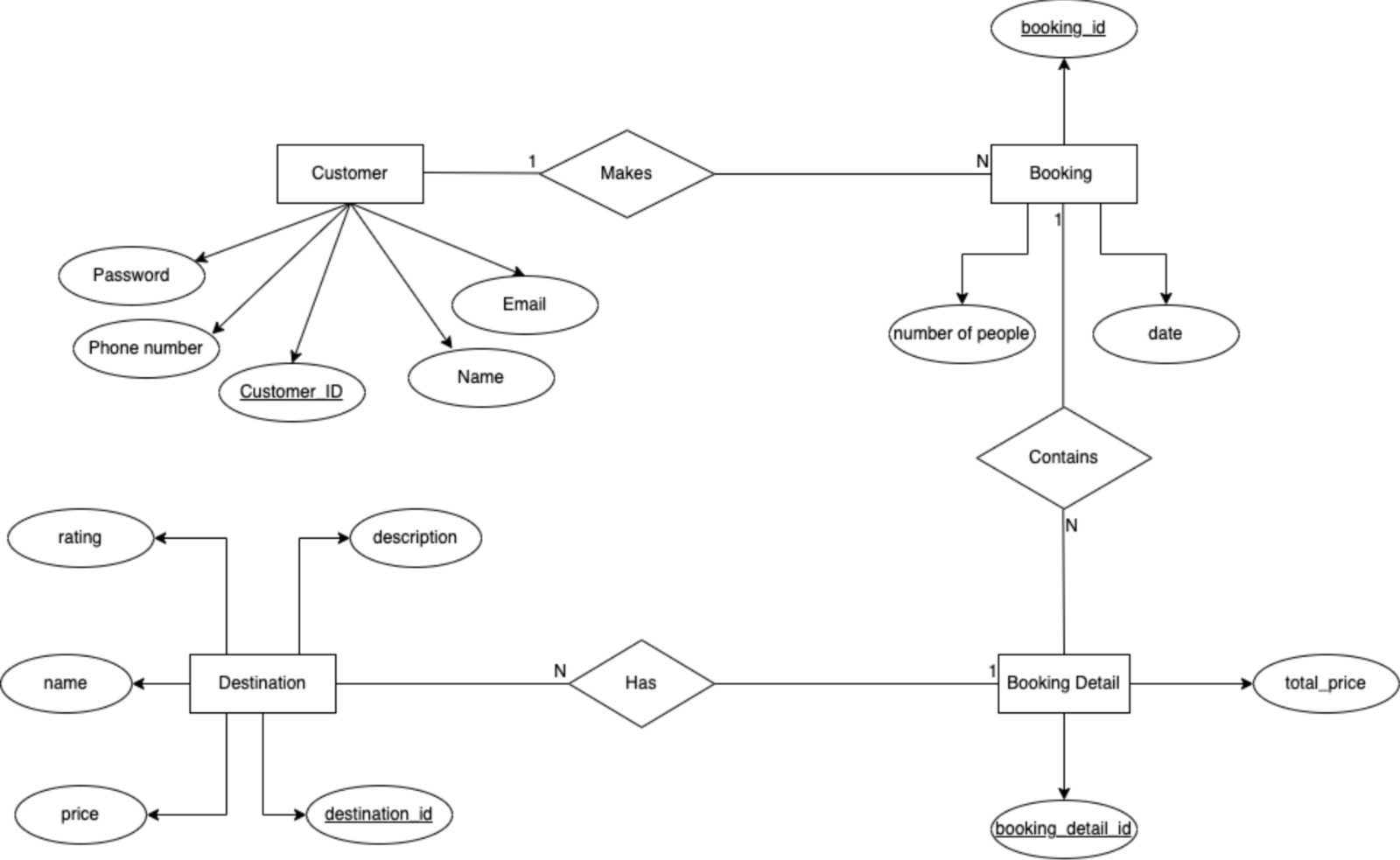
1. Security: The system should have robust security features to protect user data, payment information, and other sensitive data from unauthorized access and attacks.
2. Scalability: The system should be scalable to handle a large number of users, transactions, and data volumes.
3. Reliability: The system should be highly available and reliable, with minimal downtime or disruptions.
4. Performance: The system should be optimized for fast performance and response times, especially during peak usage periods.
5. Integration: The system should be able to integrate with various travel-related services such as payment gateways, airline booking systems, and hotel booking systems.
6. Database Management: The system should have a reliable and efficient database management system to store and manage large volumes of data.
7. User Interface: The system should have a user-friendly interface that is easy to use and navigate, with clear instructions and feedback.

Business Requirements:

1. Booking and Reservation: The system should allow users to search and book flights, hotels, car rentals, and other travel-related services.
2. Payment Processing: The system should support multiple payment options and provide a secure payment processing system.
3. Reporting and Analytics: The system should provide real-time reporting and analytics capabilities to help travel businesses monitor their operations and make data-driven decisions.
4. User Management: The system should allow users to register, log in, and manage their accounts.
5. Customer Support: The system should provide reliable customer support through multiple channels such as phone, email, and chat.
6. Marketing: The system should have marketing features to promote travel-related services to users, including targeted promotions and special offers.

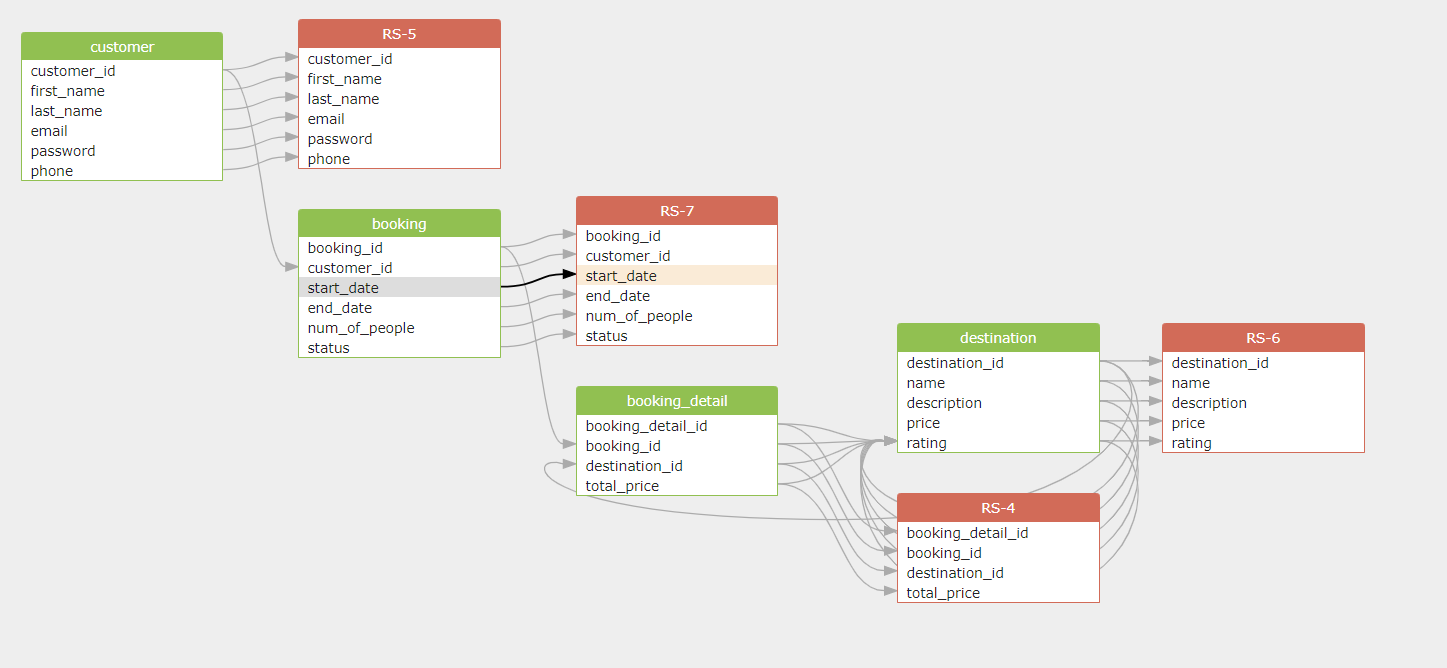
Overall, an effective online traveling booking agency database management system should meet these technical and business requirements to ensure a seamless and satisfying user experience, improve travel-related businesses' operations, and increase revenue.

E-R DIAGRAM



E-R TO TABLE

Attached below is the effective conversion of Entity-Relationship Diagram to respective Tables.



NORMALIZATION

The normalization of the database has been done up to the third normal form (3NF). Below is a breakdown of how each table is normalized:

Customer table: The Customer table appears to be in 1NF, with each row containing unique data and each column containing atomic values.

Destination table: The Destination table also appears to be in 1NF, with each row containing unique data and each column containing atomic values.

Booking table: The Booking table is in 2NF, with the primary key (booking\_id) uniquely identifying each row, and each non-key column (customer\_id, booking\_date) being fully dependent on the primary key.

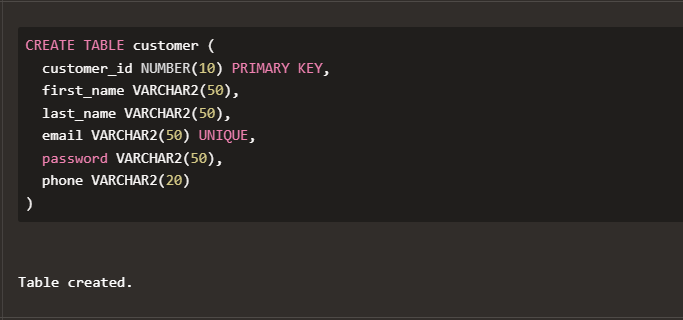
Booking\_Detail table: The Booking\_Detail table is in 3NF, with the primary key (booking\_detail\_id) uniquely identifying each row, and each non-key column being independent of other non-key columns.

Overall, the normalization of the database up to 3NF should help ensure that data is stored efficiently and consistently, and that updates and modifications to the data can be made without causing data inconsistencies or redundancies.

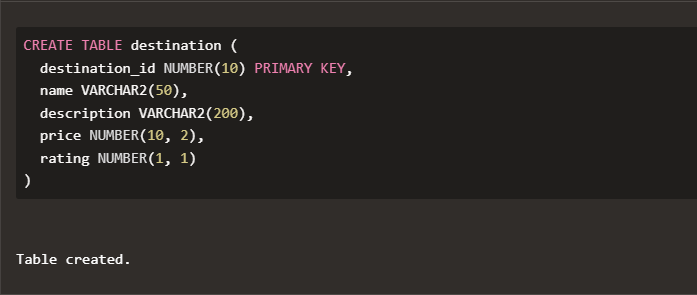
SQL & PL/SQL SNAPSHOTS

Creating the tables :

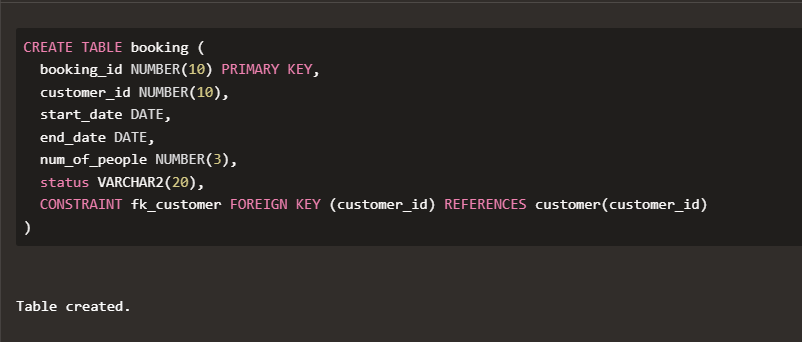
* Customer



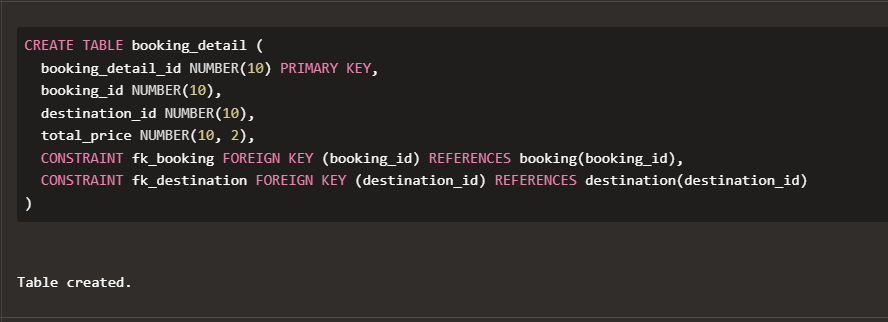
* Destination



* Booking

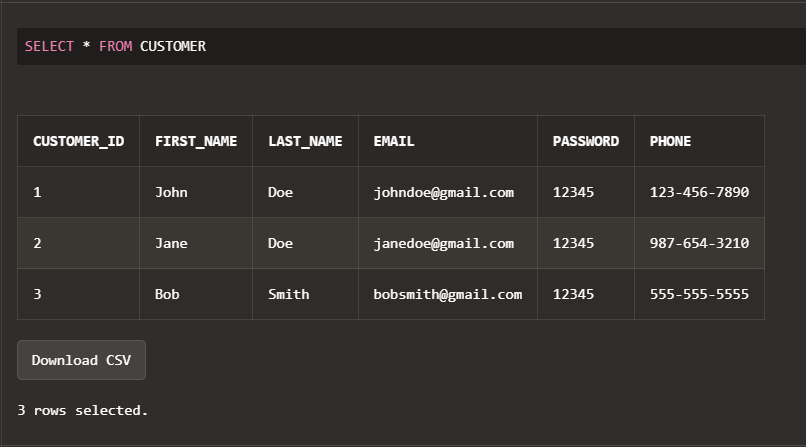


* Booking\_detail

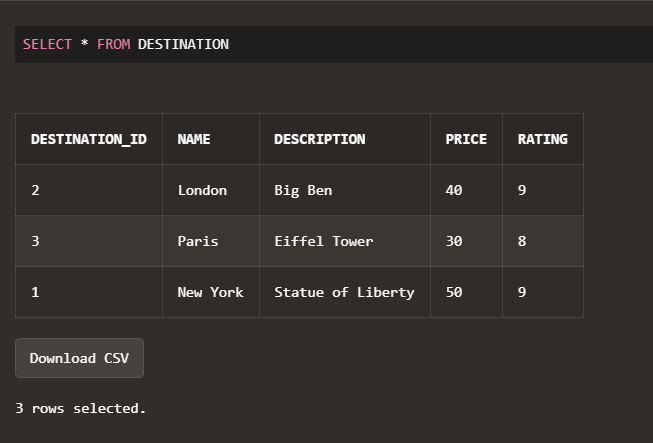


Inserting into tables:

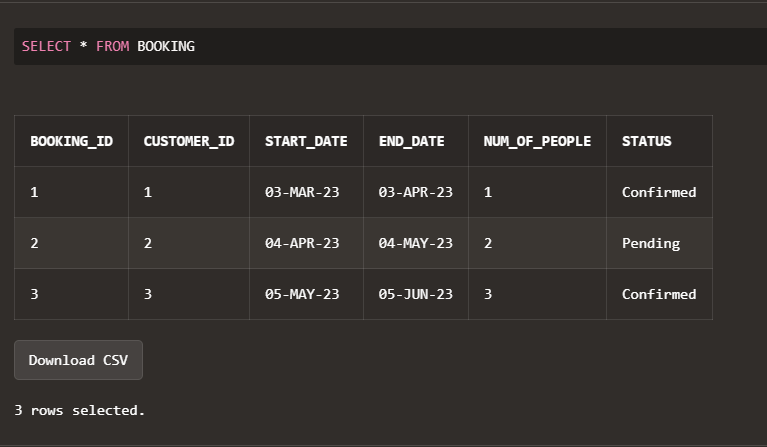
* Customer



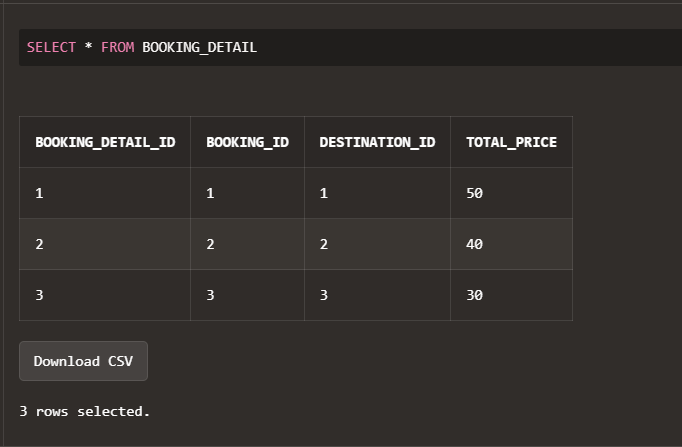
* Destination



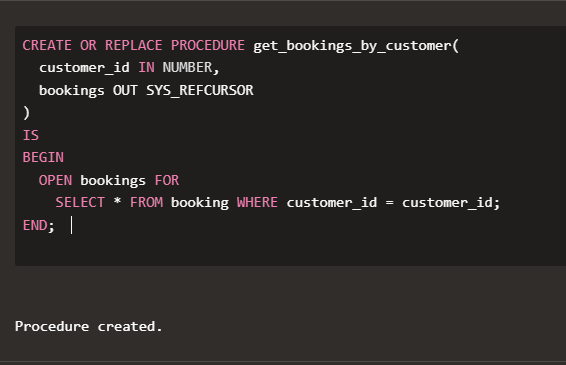
* Booking



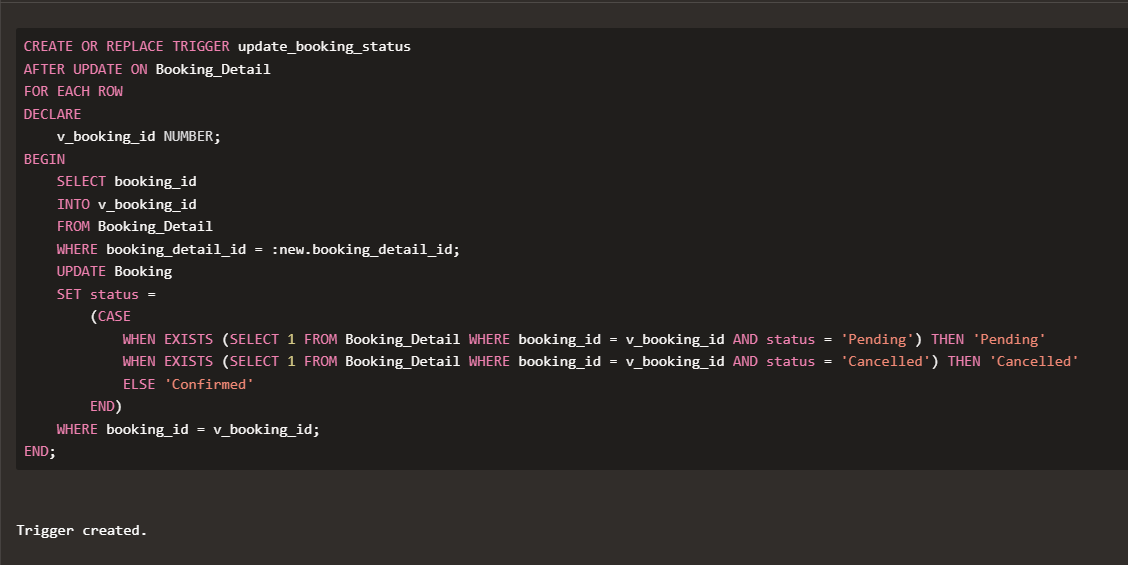
* Booking\_detail



* Procedures

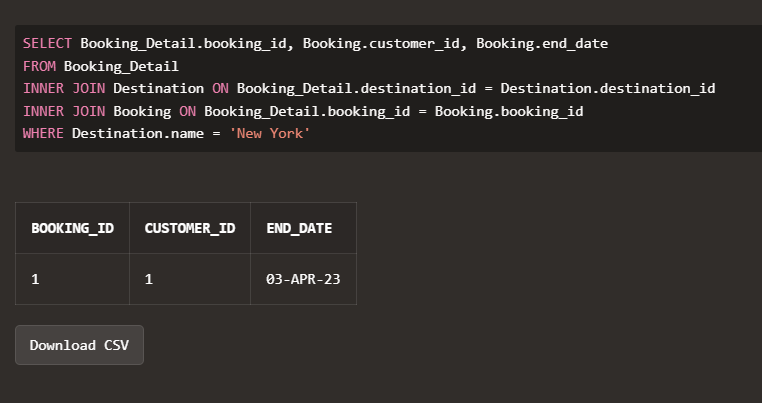


* Triggers

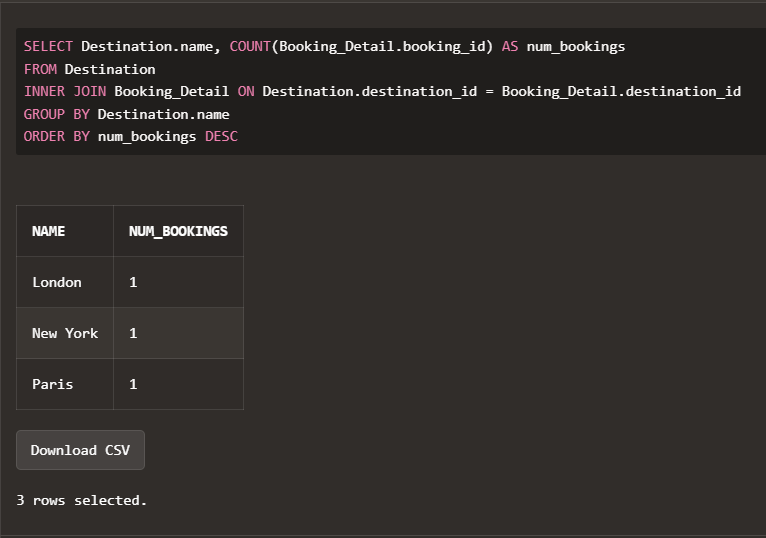


Queries :

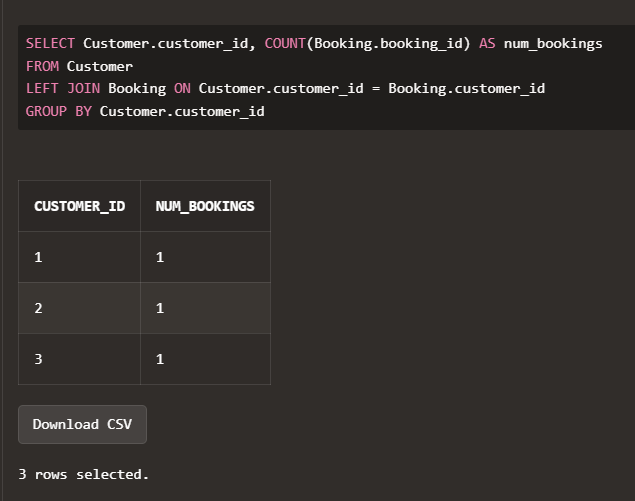
* Retrieve all the bookings made for a particular destination.( New York ).



* Retrieve the most popular destinations along with the number of times they have been booked.



* Retrieve the total number of bookings made by a particular customer



CONCLUSION

To conclude our report, we would like to summarize all that we have accomplished during the working and completion of our project. Firstly, we understood the need for an established and functioning Database System for an online booking & travelling agency, one such that would require an easy and effective access to its clients. Then we grasped the technical and business requirements to establish and maintain such a management system such as Security, Scalability, Reliability, Performance, Integration etc. Constructing an E-R model for our Database system was our primary goal for the initial stage of our project. After converting the E-R model into respective tables, we had to normalize the tables, so as to combat any redundancies that would arise, as those would be quite a problem for the projected scope of a database such as ours. Following the creation of the tables and inserting meaningful values into it, we executed some queries that took the advantage of concepts pertaining to SQL like Triggers, cursors and procedures, providing snapshots of the same.