# VISVESVARAYA TECHNOLOGICAL UNIVERSITY

**BELAGAVI-590018**

# image.png

**A PROJECT REPORT**

**ON**

**HOSTEL MANAGEMENT SYSTEM**

**BY**

**MUHAMMED ZAINUDDIN MOOSA**  **ABHIJITH MALLYA**

4SF20CI040 4SF20CI002

In the partial fulfillment of the requirement for V Sem. B. E. (AIML)

**DBMS LABORATORY WITH MINI PROJECT**

Under the guidance of

**Dr. Pushpalatha K**

Designation, Dept. of CSE

****

**Department of Computer Science & Engineering**

**SAHYADRI**

**COLLEGE OF ENGINEERING & MANAGEMENT**

**Adyar, Mangaluru-57500**

**2022-23**

**SAHYADRI**

**COLLEGE OF ENGINEERING & MANAGEMENT**

# (Affiliated to Visvesvaraya Technological University, BELAGAVI)

**Adyar, Mangaluru – 575 007**

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

*CERTIFICATE*

This is to certify that the project entitled “**HOSTEL MANAGEMENT SYSTEM”** issubmitted in partial fulfillment for the requirement of V sem. B. E. (Computer Science & Engineering), “DBMS LABORATORY WITH MINI PROJECT” during the year 2018 – 19 is a result of bonafide work carried out by

MUHAMMED ZAINUDDIN MOOSA ABHIJITH MALLYA

4SF20CI040 4SF20CI002

**………………………… …………………………**

Dr. Pushpalatha K Dr. Pushpalatha K

Assoc. Prof. Dept. of AIML HOD,Dept. of AIML

SCEM, Mangaluru SCEM, Mangaluru

**Signature of the Examiners**

**1. …………………………..**

**2. ………………………….**

**ABSTRACT**

“HOSTEL MANAGEMENT SYSTEM” is a mini project developed in ORACLE SQL for managing various activities in the hostel. For the past few years the number of educational institutions is increasing rapidly. Thereby the number of hostels is also increasing for the accommodation of the students studying in this institution. And hence there is a lot of strain on the person who are running the hostel and software’s are not usually used in this context. This particular project deals with the data required on managing a hostel and helps perform some queries to derive any set of information. Identification on how the system is vulnerable to SQL injections along with multi-lingual data support. We can improve the efficiency of the project by normalization, thus overcome the drawbacks of the existing system. The system is more user friendly because we provide GUI support.

**ACKNOWLEDGEMENT**

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

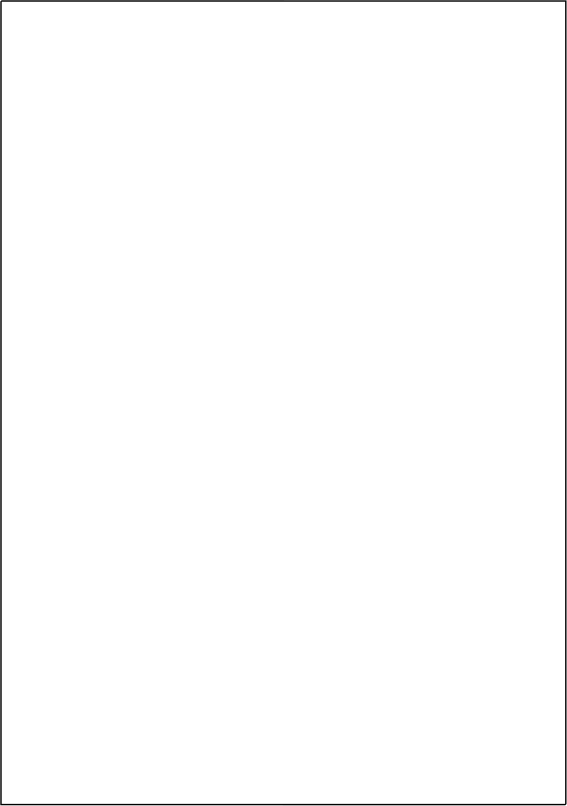
We are highly indebted to SAHYADRI for their guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project. We would like to express our gratitude towards our Faculty Dr. Pushpalatha K. for her kind cooperation and encouragement which helped us completes this project. Our thanks and appreciations also go the team members in developing the project and willing to help each other out with their abilities.

**PAGE INDEX**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Topic** | **Page No.** |
|  | Introduction  Design  E-R diagram  Relational schema(ER to relational schema)  Schema diagram  Normalization  Implementation  Results  Conclusion  References |  |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| Figure No. | Figure Name | Page No. |
| 3.1.1  3.1.2  3.2.1  3.2.2  3.3.1  5.2.1  5.3.1.1  5.3.1.2  5.3.1.3  5.3.1.4  5.3.2.1  5.3.2.2  5.3.3.1  5.3.3.2  5.3.3.3  5.4.1.1  5.4.1.2  5.4.2.1  6.1  6.2  6.3  6.4  6.5  6.6  6.7  6.8  6.9  6.10  6.11  6.12  6.13  6.14  6.15 | E-R Notation  E-R Diagram  Mapping of Regular Entities  Mapping of 1: N relationship  Schema Diagram  Connecting to Database  Insert for Products  Insert for Customer  Insert for Category  Insert for Orders  Update for Products  Update for Customer  Delete for Customer  Delete for Product  Delete for Category  Trigger for Update Quantity  Trigger for Product Name  Stored Procedure for Customer Details  Login Page  Home Page  Product Details Page  Add Product Page  Update Product Page  Delete Product Page  Customer Details page  Add Customer Page  Update Customer Page  Delete Customer Page  Print Category Details Page  Category Details Page  Add Category Page  Delete Category Page  Order Details Page | 6  7  9  10  11  18  18  19  19  20  21  21  22  22  23  23  24  24  25  25  26  26  27  27  28  28  29  29  30  30  31  31  32 |

INTRODUCTION

chapter-1

1.1 INTRODUCTION TO DBMS

The “Database Management System” in short DBMS is a software system that uses a standard method of cataloguing retrieving and running queries on data. DBMS is a collection of inter-related data which helps in efficient retrieval, insertion and deletion of data from database and organises the data in the form of tables, views, schemas, reports etc.

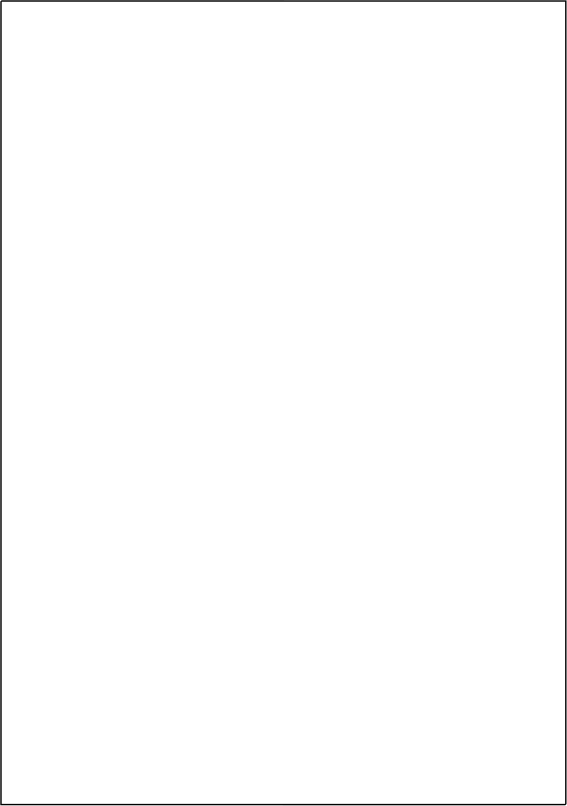
DBMS include change management, performance monitoring/tuning and backup and recovery. Many database management systems are also responsible for automated rollbacks, restarts and recovery as well as the logging and auditing of activity. The DBMS can offer both logical and physical data independence. That means it can protect users and applications from needing to know where data is stored or having to be concerned about changes to the physical structure of data.

It provides us with many functionalities and is more advantageous than the traditional file system in many ways such as 1) processing queries and object management. 2)Controlling redundancy and inconsistency. 3)Efficient memory management and indexing.4) Concurrency control and transaction

1.2 BACKGROUND OF THE PROJECT

A hostel management system is a software application that is used to automate and streamline the daily operations of a hostel. This includes tasks such as managing reservations, billing, and room assignments. A hostel management DBMS, or database management system, is the underlying technology that is used to store and manage all of the data associated with a hostel, such as information about guests, rooms, and reservations. The goal of a hostel management DBMS is to make it easy for staff to access and update information, and to provide accurate and up-to-date reports to management.

1.3 NECESSITY OF PROJECT



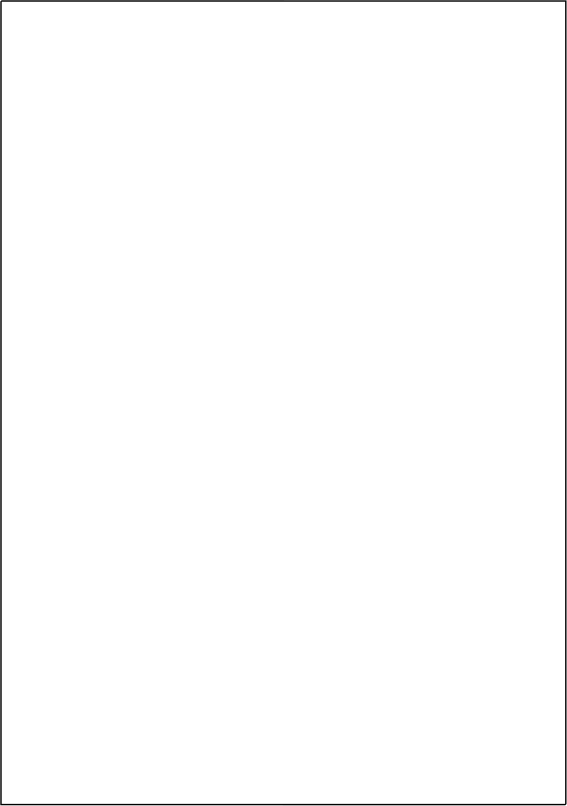
The necessity of a hostel management DBMS arises from the need to effectively manage and organise the various tasks and processes associated with running a hostel. Some of the key reasons why a hostel management DBMS is necessary include:

* Efficiency: A hostel management DBMS can automate many of the manual tasks that are associated with running a hostel, such as room assignments and billing. This can help to improve the efficiency of the operation and reduce the potential for errors.
* Accuracy: A hostel management DBMS can provide staff with easy access to accurate and up-to-date information about guests, rooms, and reservations. This can help to improve the accuracy of the operation and reduce the potential for errors.
* Data Management: A hostel management DBMS can provide a centralised location for storing and managing all of the data associated with a hostel. This can make it easy for staff to access and update information, and to generate reports and analyze data.
* Cost Savings: A hostel management DBMS can help to reduce the costs associated with running a hostel by automating many of the manual tasks that are associated with the operation.
* Better customer service: A hostel management DBMS can help to provide better customer service by allowing staff to quickly and easily access information about guests and their reservations, which can help to improve the guest experience.
* Better decision making: A hostel management DBMS can provide management with real-time data and reports, which can help them to make informed business decisions and improve overall performance.

Overall, a hostel management DBMS can help to improve the efficiency and accuracy of the operation, and can help to provide better customer service and support informed decision making.

1.4 APPLICATIONS AND ADVANTAGES

This software has several applications and advantages that make it a valuable tool for managing a hostel. Some of the key applications and advantages include:

* Automation of tasks: A hostel management DBMS can automate many of the manual tasks that are associated with running a hostel, such as room assignments, billing, and reservations management. This can help to improve the efficiency of the operation and reduce the potential for errors.
* Data Management: A hostel management DBMS can provide a centralized location for storing and managing all of the data associated with a hostel. This can make it easy for staff to access and update information, and to generate reports and analyze data.
* Real-time information: A hostel management DBMS can provide staff and management with real-time information about guests, rooms, and reservations. This can help to improve the accuracy of the operation and reduce the potential for errors.
* Cost Savings: A hostel management DBMS can help to reduce the costs associated with running a hostel by automating many of the manual tasks that are associated with the operation.

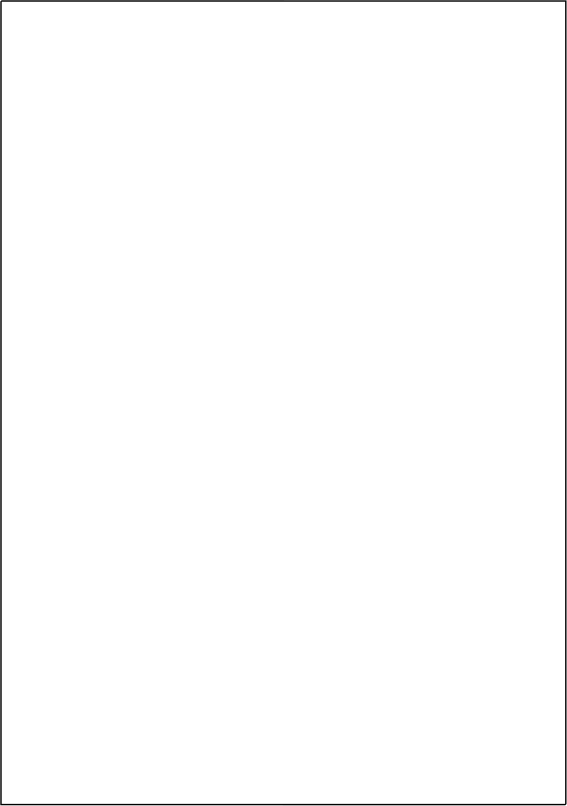
Overall, a hostel management DBMS can help to improve the efficiency and accuracy of the operation, and can help to provide better customer service and support informed decision making.

**1.5 IMPLEMENTATION**

The Hostel management system is implemented using admin registration/login module, menu module, admin access module, with insert, update, delete operations. Oracle Database XE holds the database and java for the front end which displays the provided modules.

1.6 ORACLE

Oracle database is a relational database management system. It is also called Oracle DB, or simply Oracle. It is produced and marketed by Oracle Corporation. Oracle database was the first DB that designed for enterprise grid computing and data warehousing. Enterprise grid computing provides the most flexible and cost-effective way to manage information and applications. It uses SQL queries as a language for interacting with the database. An Oracle database offers the following features to meet the database management requirements

* scalability and performance
* availability
* backup and recovery
* security

Oracle is a powerful database server management software that can serve the requirements of Enterprise level and small level applications as well. It includes almost all the features required to support modern applications and therefore, is widely used.

1.7 ORACLE XE

Oracle XE is a smaller version of the Oracle Database that is designed to be easy to install, manage, and use. It has a smaller footprint and requires less memory and disk space compared to the full version of Oracle Database. It can be installed on Windows and Linux operating systems, and has a graphical user interface (GUI) for managing the database.

Oracle XE includes many of the same features found in the paid versions of Oracle Database, such as data concurrency, backup and recovery, and security features. However, it is limited to storing up to 11 GB of user data, and can only support up to 2 concurrent connections. It also has a smaller set of options for backup and recovery.

Oracle XE is intended for small businesses, developers, and students, and is a great option for learning about Oracle Database and for developing and testing applications. It is also a good option for small-scale deployment of applications. It can be downloaded from the Oracle website for free.

1.8 JAVA

Java is a popular programming language. It is owned by Oracle, and more than 3 billion devices run Java. Java works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc.). It is open-source secure, fast, free and powerful. As Java is close to C++ and C#, it makes it easy for programmers to switch to Java.

Java is an Object-Oriented programming language developed by James Gosling in the early 1990s. The team initiated this project to develop a language for digital devices such as set-top boxes, television, etc.

Java is used in all kinds of applications like Mobile Applications (Android is Java-based), desktop applications, web applications, client-server applications, enterprise applications, and many more.

1.9 ECLIPSE

Eclipse is an open-source, cross-platform Integrated Development Environment (IDE) that is widely used for developing Java applications. It is written primarily in Java and can be used to develop applications in other languages as well, such as C, C++, Python, and Ruby. Eclipse provides a wide range of features that make it a popular choice among developers, including:

* A powerful code editor with code completion, refactoring, and debugging tools
* A built-in Java development environment with support for JUnit testing
* A visual layout editor for building user interfaces
* A plugin architecture that allows for easy integration of additional tools and frameworks
* Built-in support for version control systems such as Git and SVN

Eclipse is a widely-used and popular tool for development and it has a large and active community of developers who contribute to the development and maintenance of the IDE. It is also extensible and customizable, allowing developers to tailor the IDE to their specific needs.

It is available for Windows, Linux, and MacOS.

Requirement Specification

CHAPTER 2

2.1 Hardware Specification

Operating System: Windows OS/LINUX/MacOS.

Memory: Minimum of 1GB of RAM, Minimum of 2GB hard disk space.

2.2 Software Requirements

Backend: Oracle Database 10g Release 2 (10.2.0.1) Express Edition

Programming Language: Java

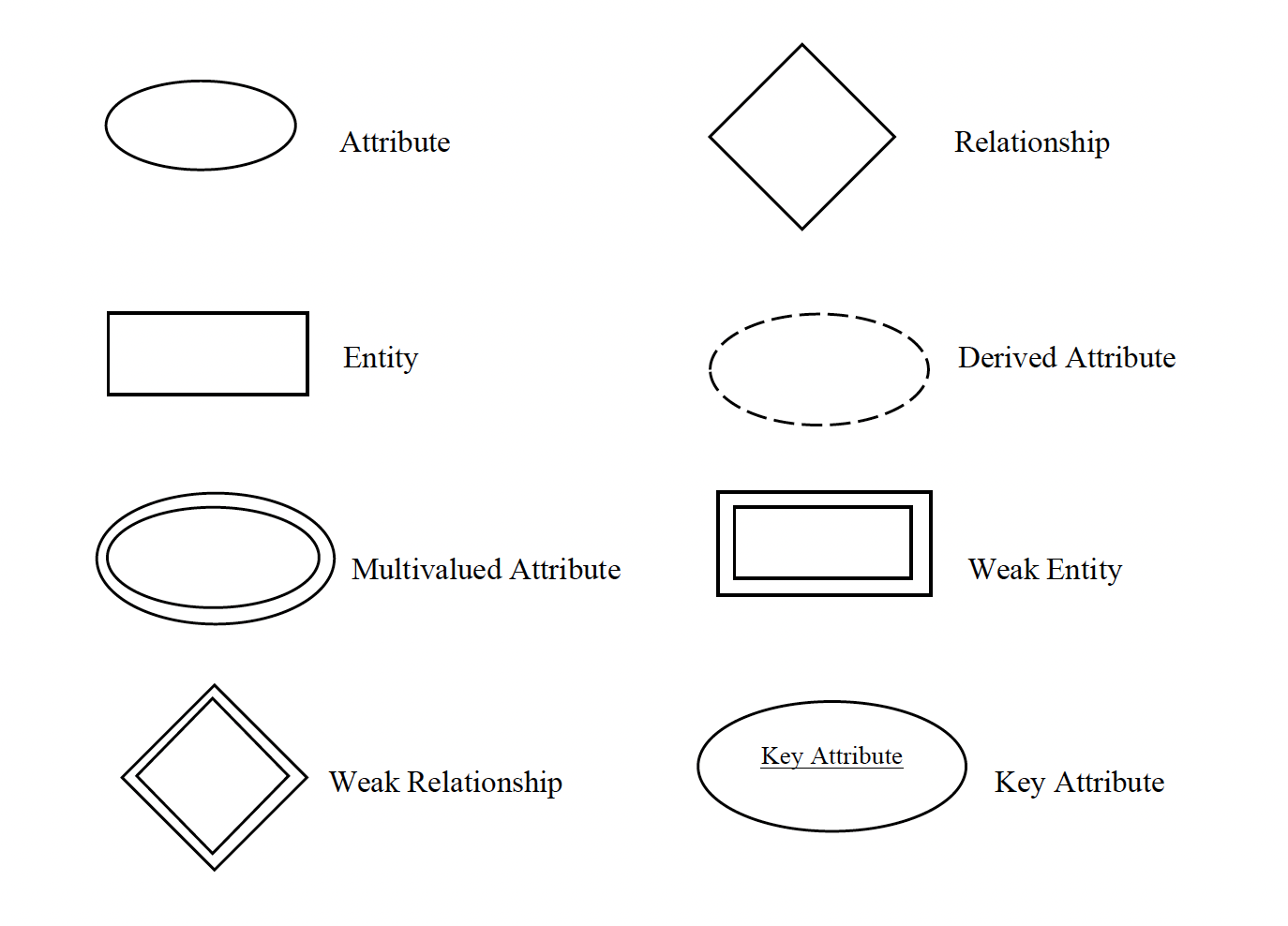
IDE: Eclipse

DESIGN

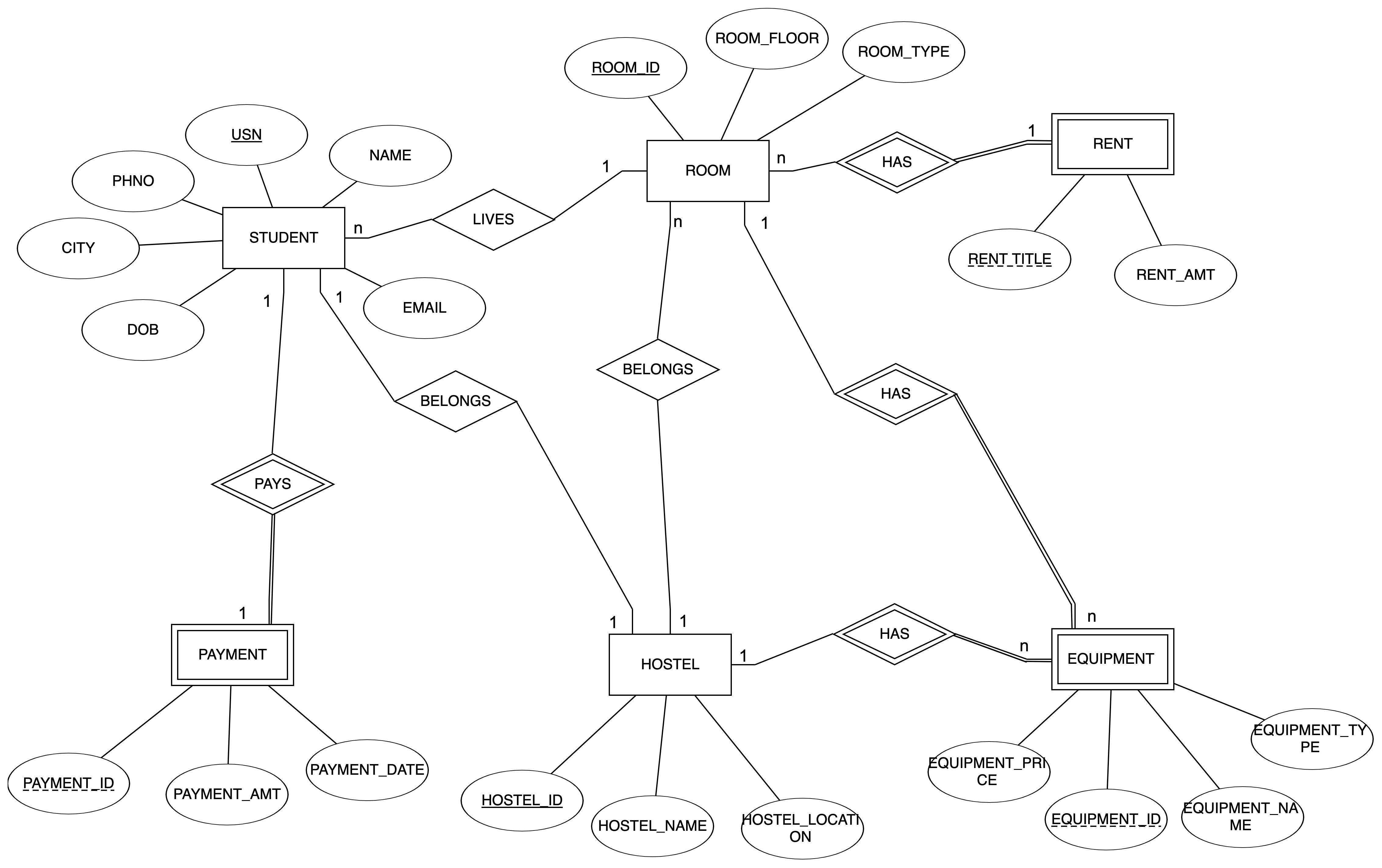
CHAPTER 3

3.1 ER DIAGRAM

An entity-relationship model describes interrelated things of interest in a specific domain of knowledge. The ER Diagram of our project is shown in the figure:3.1.1

Symbols in Entity Relationship:

**Figure 3.1.1**: - ER Notation



**Figure 3.1.2**: E-R Diagram

3.2 RELATIONAL SCHEMA

Mapping From ER Diagram to Schema Diagram

1. Mapping of regular entities :-This step involves mapping all the regular entity types to tabular format by identifying their primary keys.

2. Mapping of 1:1 Relation:-In this step foreign keys are assigned using foreign key approach. The primary key of the participating relation R or S is added as primary key to second entity types by looking at the participating constraints.

3. Mapping of 1:N Relation:-Foreign key approach is used to add one sided primary key to the n sided entity at foreign key.

4. Mapping of M:N Relation :-Here we use the cross reference approach where the relationship is converted to a new relation within attributes on primary keys of both participating relation.

5. Mapping of Weak Entity :-When mapping weak entity types along with other attributes the partial key and primary key of parent entity together will form their primary key of the new relation.

6. Mapping of N-ary Relation:-For mapping N array relationship we create a new relation with a relationship name in its attribute and primary keys of all participating entity types.

7. Mapping of Multivalued Relation :-For multivalued attributes a separate relation has to be created along with primary key of parent relation. A relational schema for a database is an outline of how data is organized.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| USN | NAME | EMAIL | DOB | MOBILE | ROOM\_ID | HOSTEL\_ID |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ROOM\_ID | ROOM\_FLOOR | ROOM\_TYPE | RENT\_TITLE | HOSTEL\_ID |

|  |  |
| --- | --- |
| RENT\_TITLE | AMOUNT |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| E\_ID | E\_NAME | E\_TYPE | E\_PRICE | ROOMID |

|  |  |  |
| --- | --- | --- |
| HOSTEL\_ID | TYPE | LOCATION |

|  |  |  |  |
| --- | --- | --- | --- |
| PAYMENT\_ID | PAYMENT\_DATE | AMOUNT | USN |

IMPLEMENTATION

CHAPTER 5

5.1 Table Structure

5.1.1 STUDENT

CREATE TABLE STUDENT(

USN VARCHAR(10) PRIMARY KEY,

STUDENT\_NAME VARCHAR(20),

STUDENT\_PHONE VARCHAR(10) CHECK( LENGTH(STUDENT\_PHONE)=10 ),

STUDENT\_EMAIL VARCHAR(30) CHECK(STUDENT\_EMAIL LIKE '%@gmail.com') ,

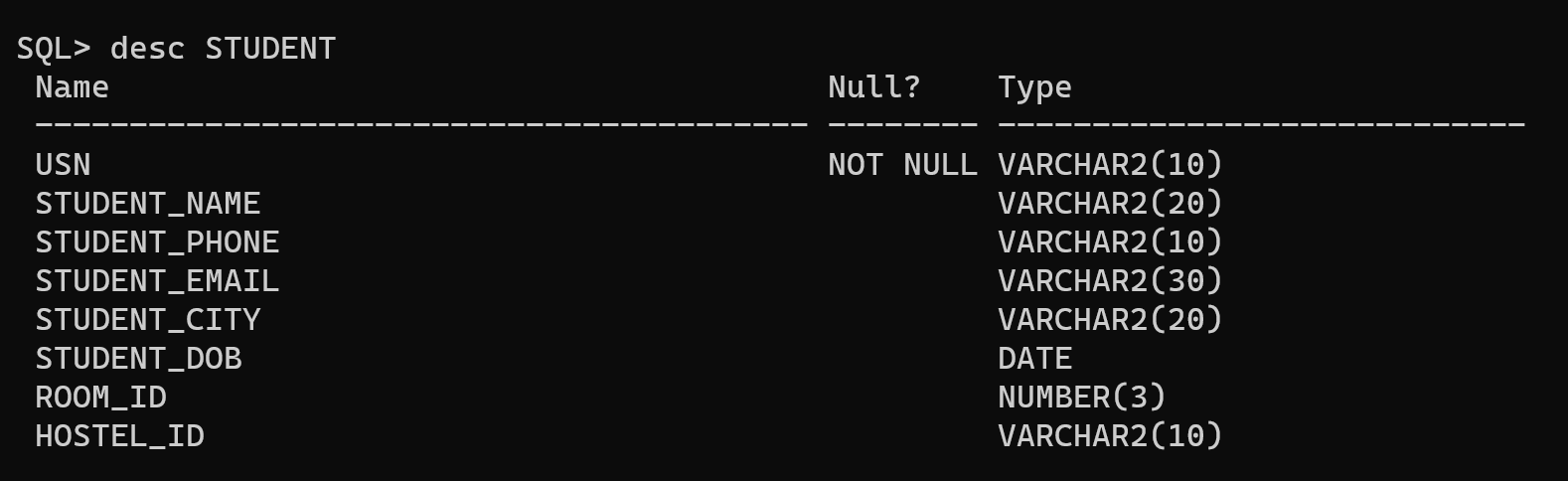
STUDENT\_CITY VARCHAR(20),

STUDENT\_DOB DATE CHECK( STUDENT\_DOB BETWEEN TO\_DATE('01-01-2000', 'DD-MM-YYYY') AND TO\_DATE('01-01-2005', 'DD-MM-YYYY')),

ROOM\_ID REFERENCES ROOM ON DELETE CASCADE,

HOSTEL\_ID REFERENCES HOSTEL ON DELETE CASCADE

);



5.1.2 EQUIPMENT

CREATE TABLE EQUIPMENT

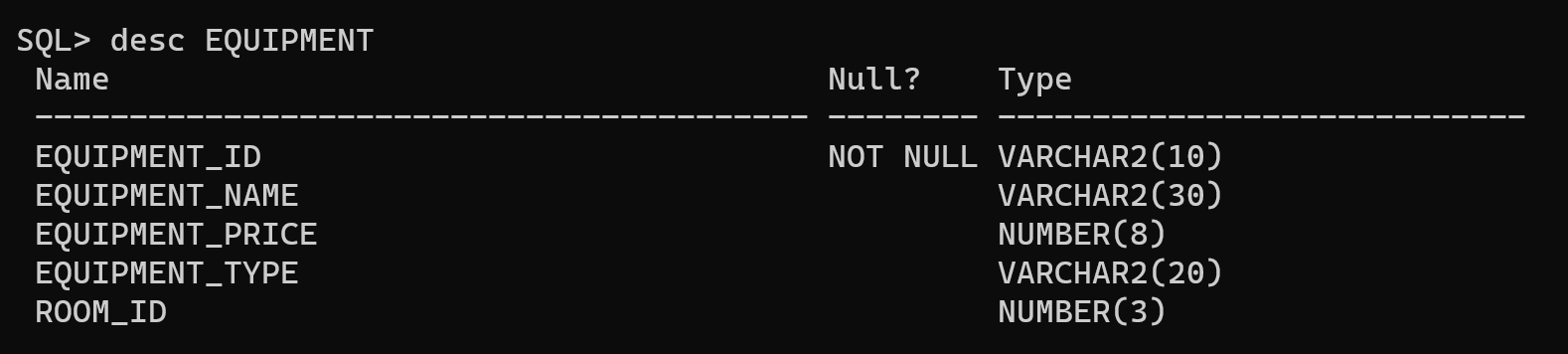
(EQUIPMENT\_ID VARCHAR(10) PRIMARY KEY,

EQUIPMENT\_NAME VARCHAR(30),

EQUIPMENT\_PRICE NUMBER(8),

EQUIPMENT\_TYPE VARCHAR(20),

ROOM\_ID REFERENCES ROOM ON DELETE CASCADE);



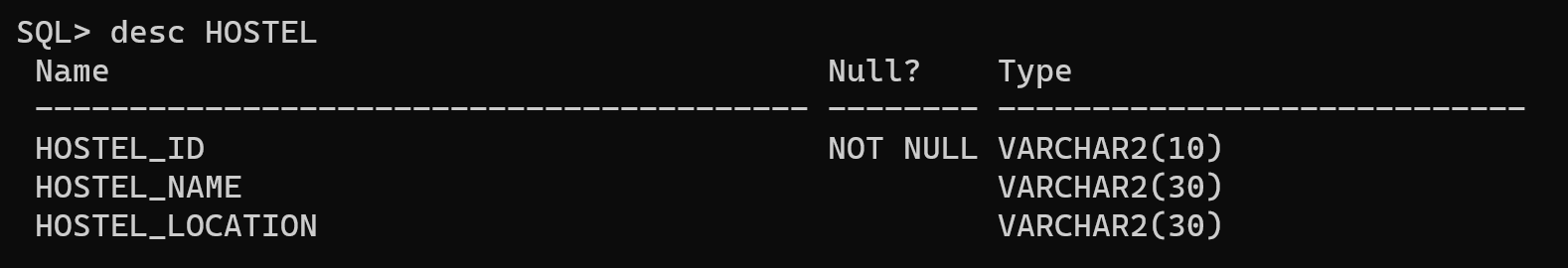
5.1.3 HOSTEL

CREATE TABLE HOSTEL

(HOSTEL\_ID VARCHAR(10) PRIMARY KEY,

HOSTEL\_NAME VARCHAR(30),

HOSTEL\_LOCATION VARCHAR(30));

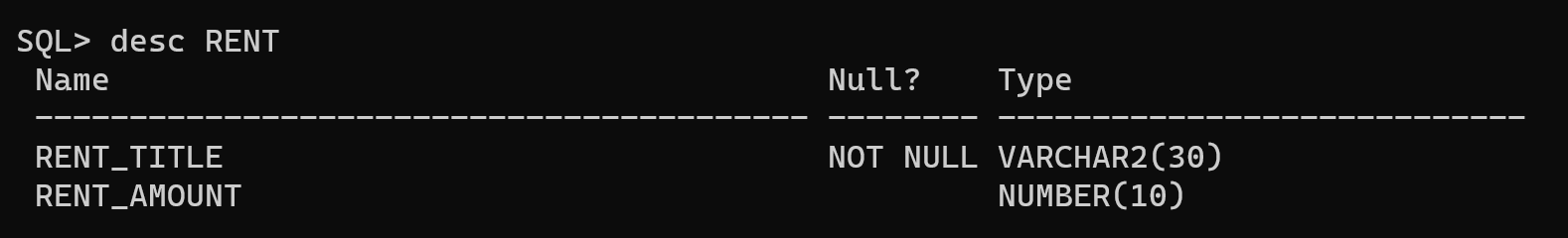


5.1.4 RENT

CREATE TABLE RENT

(RENT\_TITLE VARCHAR(30) PRIMARY KEY,

RENT\_AMOUNT NUMBER(10)

);

5.1.5 ROOM

CREATE TABLE ROOM

(ROOM\_ID NUMBER(3) PRIMARY KEY,

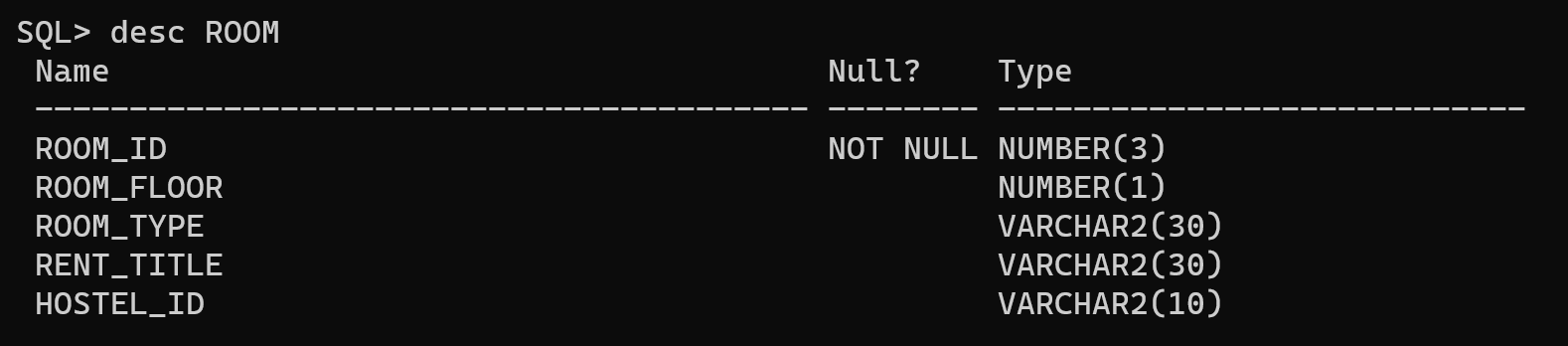
ROOM\_FLOOR NUMBER(1),

ROOM\_TYPE VARCHAR(30),

RENT\_TITLE REFERENCES RENT ON DELETE CASCADE,

HOSTEL\_ID REFERENCES HOSTEL ON DELETE CASCADE

);



5.1.6 PAYMENT

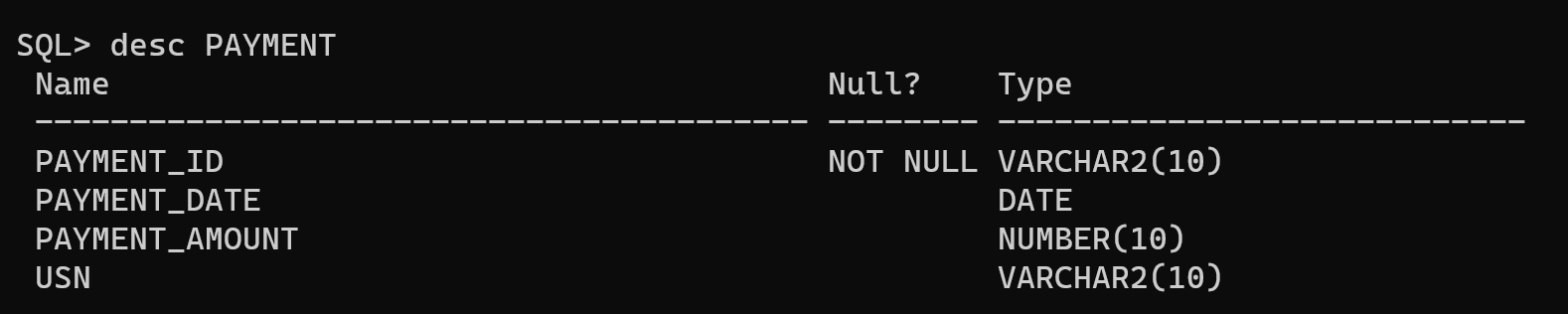
CREATE TABLE PAYMENT

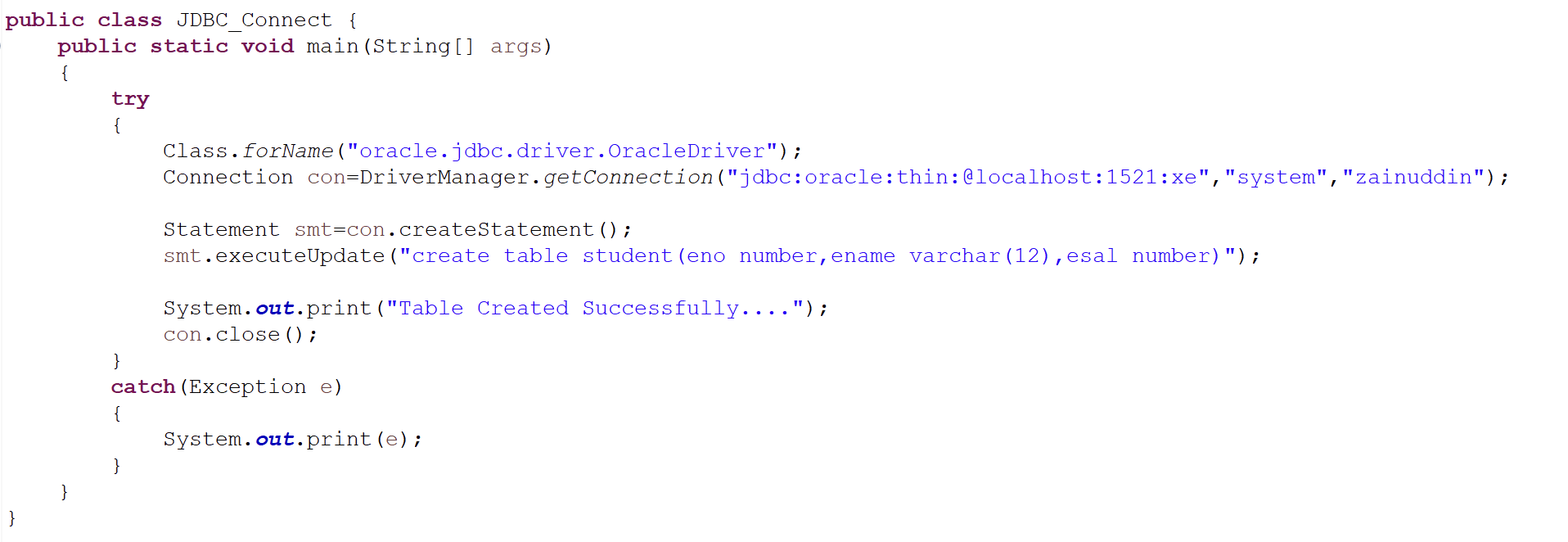
(PAYMENT\_ID INT PRIMARY KEY ,

USN REFERENCES STUDENT ON DELETE CASCADE,

RENT\_TITLE REFERENCES RENT ON DELETE CASCADE,

PAYMENT\_AMOUNT NUMBER(10),

PAYMENT\_DATE DATE);

5.2 JDBC DRIVER IS USED TO CONNECT JAVA AND ORACLE 10G XE

**Fig 5.2.1:** Connecting to the database

5.3 INSERT/UPDATE/DELETE OPTION HAS BEEN IMPLEMENTED AS FOLLOWS

5.3.1 Insert