

**AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH
(AIUB)**

FACULTY OF SCIENCE & TECHNOLOGY



Course Title
INTRODUCTION TO DATABASE (2024)

Semester: Spring 2023-2024

Section: [G]

TITLE

SKYLINE - REAL ESTATE MANAGEMENT SYSTEM

Supervised By

SAEEDA SHARMEEN RAHMAN

Submitted By: Group no:

Name	ID
AVIJIT SARKER	23-51032-1
PRASANTA JIT PAUL	23-50937-1
MD.MEHEDI HASAN	22-49936-3
MD.SHAMSUL ALAM	23-50735-1
MD.SABBIR HOSSAIN ROHAN	23-50753-1

TABLE OF CONTENTS

TOPICS	Page no.
Title Page	1
Table of Content	2
1. Introduction	3
2. Case Study	4
3. ER Diagram	5
4. Normalization	6-13
5. Finalization	14
6. Inserted Values in the tables	15-27
7. Query Test in DB	28-34
8. Conclusion	35

Introduction

Welcome to the groundbreaking Real Estate Management System (REMS) tailored exclusively for SKYLINE, poised to revolutionize the operational landscape of this esteemed organization. Our innovative database project stands as a testament to our commitment to enhance management visibility, optimize data accessibility, and drive transformative changes within the real estate industry.

At the core of this REMS lies a steadfast objective: to streamline and centralize vast amounts of information, empowering every department and the board of directors with effortless access to critical data. Through advanced data organization and visualization measures, the REMS ensures the protection of sensitive information while facilitating seamless access, thereby serving as a comprehensive repository for various crucial datasets.

One of the primary challenges addressed by this REMS is the reduction of manual, time-consuming processes. By digitizing workflows and automating tasks, the system eliminates the need for laborious pen-and-paper methods, thus saving valuable time and resources. Consequently, accessing data, generating reports, and conducting analysis become streamlined and efficient, empowering employees to make data-driven decisions swiftly.

Furthermore, the REMS heralds a transformative shift in the operational paradigm of SKYLINE. By providing a comprehensive view of the company's financial health in real-time, decision-makers can monitor revenue, expenses, and profitability with unprecedented clarity. This enables them to identify trends, seize opportunities, and mitigate risks more effectively, ultimately fostering strategic decision-making and enhancing overall operational efficiency.

In addition to its operational benefits, the REMS promotes transparency and collaboration within the company. Authorized personnel from any department or board member can access relevant information promptly, fostering a culture of openness and empowering employees with valuable insights. This facilitates cross-departmental cooperation, leading to improved communication and more informed decision-making processes.

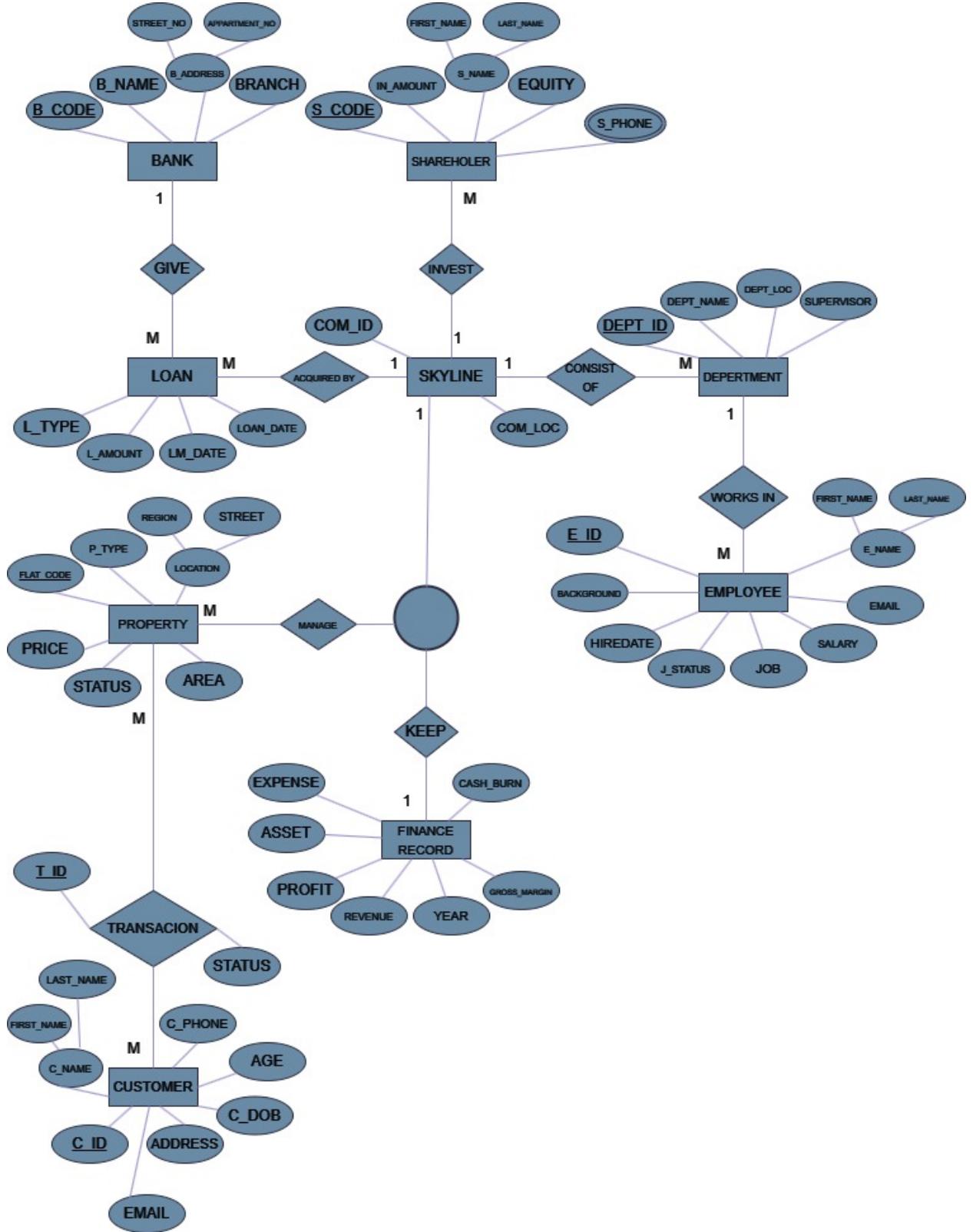
In conclusion, the Real Estate Management System developed for SKYLINE represents a paradigm shift in the industry, centralizing data, streamlining operations, and providing invaluable insights. By replacing manual processes with digital efficiency, the REMS optimizes time, enhances strategic decision-making, and drives day-to-day operational effectiveness. SKYLINE's adoption of this cutting-edge technology ensures a competitive advantage, positioning the company as a leader in the real estate market.

Case Study / Scenario

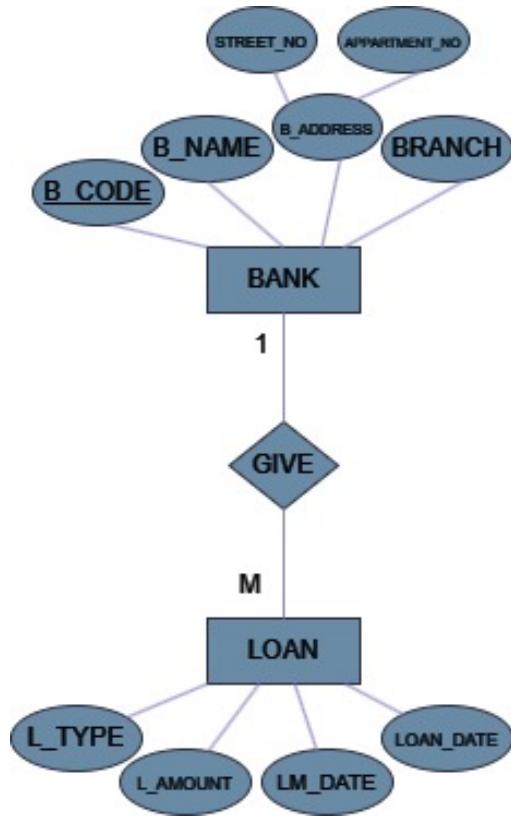
StudentID1: Name:	StudentID3: Name:
StudentID2: Name:	StudentID4: Name:
CO2: Understand the fundamental concepts underlying database systems and gain hands-on experience with ER diagram Case study	
PO-c2: Develop process for complex computer science and engineering problems considering cultural and societal factors.	Marks

In a real estate company, there are many departments based on work roles and fields. Each department has a unique department Id, name, location, and supervisor. Many employees work for the department they are associated with. Each employee has a name, background qualification, email address, identity number, salary, date of joining, job role and job status. Employees are referred to by their employee id number. When any customer visits in the headquarter or the Skyline web application to buy or rent flat or property, a customer ID number is generated and name, date of birth, age, address, phone number, email address are also stored. Skyline manages many properties and stores the property details containing unique flat code, location, type, status and area. A customer or more than one customer can buy one or many properties. While transaction process, transaction id and status are stored of a customer. In order to run the company with a proper flow by overcoming financial obstacles, Skyline, as an evolving company to disrupt the real estate market, takes a lot of investment from many shareholders and loan from more than one banks. The details such as name, amount of investment, unique code, equity amount and multiple phone numbers of a shareholder are being stored in the database. While taking loans from bank, both the bank details and the loan information are taken which includes bank code, name, address, branch, loan amount, loan type, loan date as well as loan maturity date. To keep track company growth and insights, having financial record is mandatory. So, it also includes yearly financial record providing profit, revenue, cash burn, gross margin, asset, expenses details which will be beneficial to have a more efficient approach and financial success.

ER Diagram



Normalization



GIVE (B_CODE, B_NAME, STREET_NO, APPAR_NO, BRANCH, L_TYPE, L_AMOUNT, L_DATE, LM_DATE)

1NF: STREET_NO, APPAR_NO

No multivalued attribute

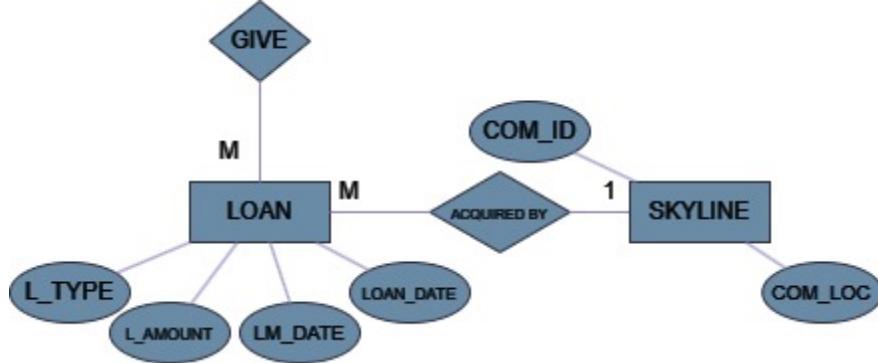
2NF: B_CODE, B_NAME, STREET_NO, APPAR_NO, BRANCH B_CODE, L_TYPE, L_AMOUNT, L_DATE, LM_DATE

3NF: No transitive dependency

B_CODE, B_NAME, STREET_NO, APPAR_NO, BRANCH B_CODE, L_TYPE, L_AMOUNT, L_DATE, LM_DATE

Table: B_CODE, B_NAME, STREET_NO, APPAR_NO, BRANCH B_CODE, L_TYPE, L_AMOUNT, L_DATE, LM_DATE

2.



ACQUIRED BY (L_TYPE, L_AMOUNT, L_DATE, LM_DATE, COM_ID, COM_LOC)

1NF: No multivalued attribute

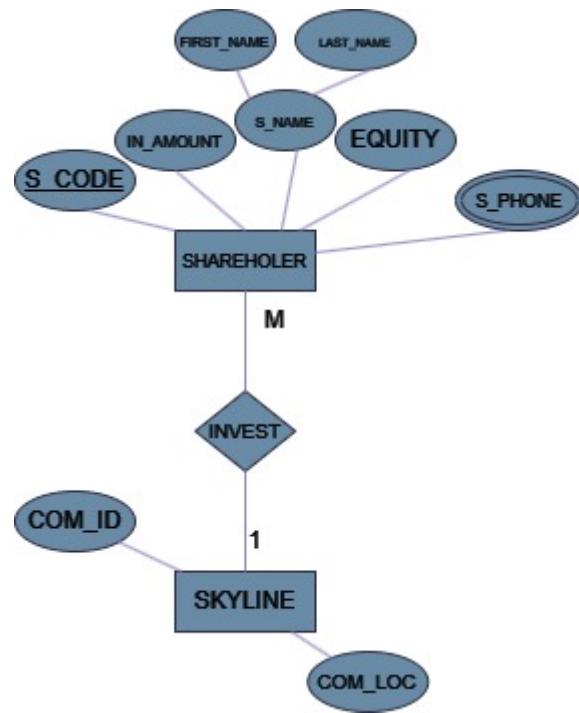
2NF: L_TYPE, L_AMOUNT, L_DATE, LM_DATE, , COM_ID ,COM_LOC

3NF: No transitive dependency

L_TYPE, L_AMOUNT, L_DATE, LM_DATE, COM_ID, COM_LOC

Table: L_TYPE, L_AMOUNT, L_DATE, LM_DATE, COM_ID, COM_LOC

3.



INVEST (S_CODE, FIRST_NAME, LAST_NAME, IN_AMOUNT, EQUITY, COM_ID, COM_LOC)

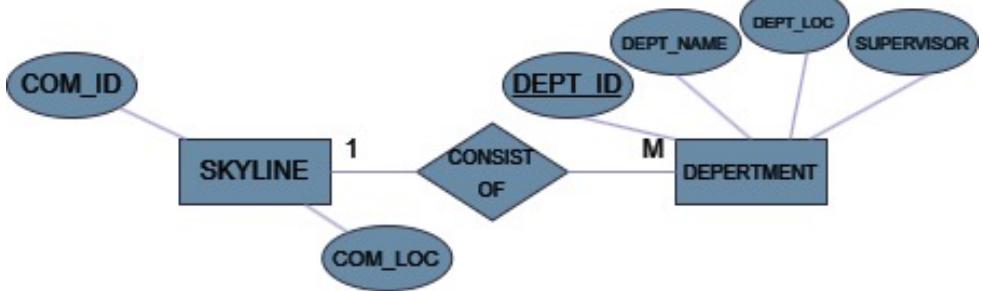
1NF: FIRST_NAME, LAST_NAME No multivalued attribute

2NF: S_CODE, FIRST_NAME, LAST_NAME, IN_AMOUNT, EQUITY, S_CODE, S_NAME, S_PHONE ,COM_ID, COM_LOC

3NF: No transitive dependency S_CODE, FIRST_NAME, LAST_NAME, IN_AMOUNT, EQUITY
S_CODE, S_NAME, S_PHONE ,COM_ID, COM_LOC

Table: S_CODE, FIRST_NAME, LAST_NAME, IN_AMOUNT, EQUITY S_CODE, S_NAME, S_PHONE , COM_ID, COM_LOC

4.



CONSIST OF (COM_ID, Com_Loc, DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR)

1NF: No multivalued attribute

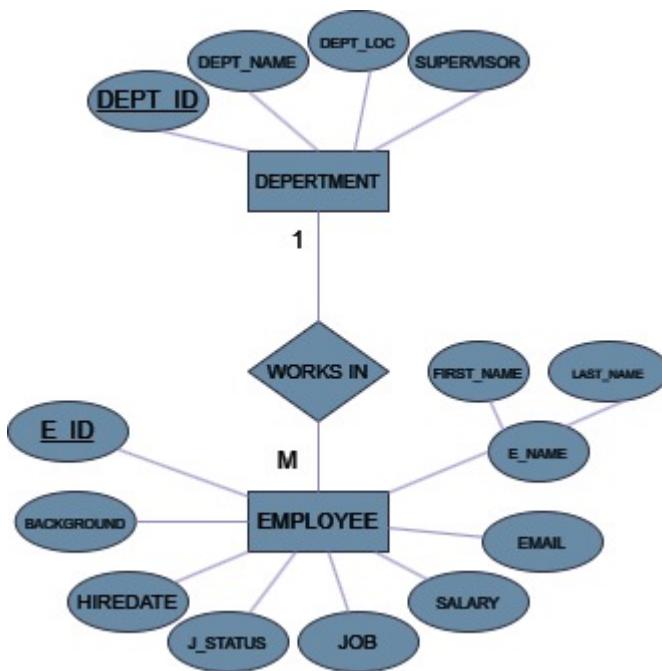
2NF: COM_ID, COM_LOC, DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR

3NF: No transitive dependency

COM_ID, COM_LOC, DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR

Table: COM_ID, COM_LOC, DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR

5.



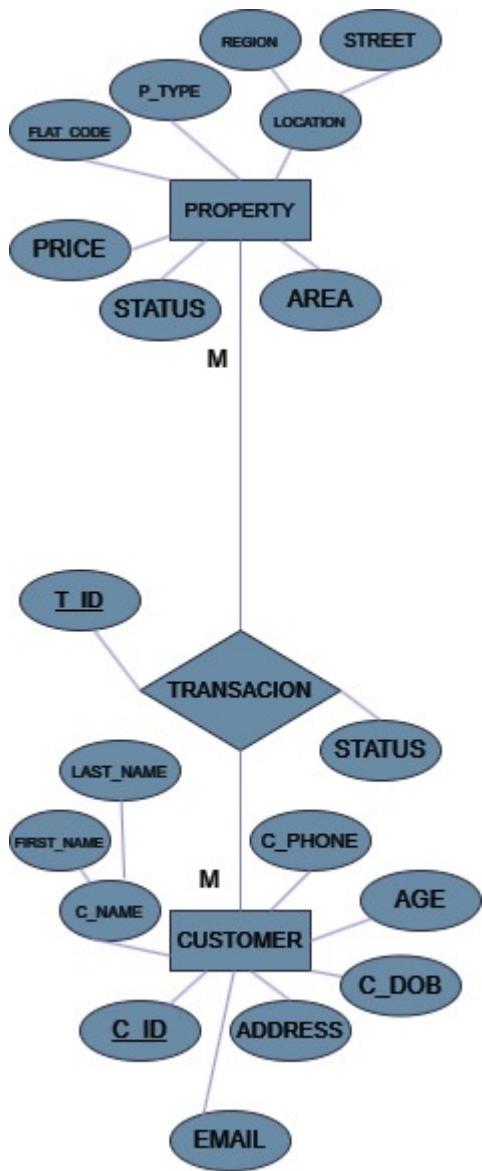
WORKS IN (DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR, E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID)

1NF: FIRST_NAME, LAST_NAME No multivalued attribute

2NF: DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID

3NF: No transitive dependency DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID

6.



TRANSACTION (C_ID, FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, C_DOB, FLAT_CODE, P_TYPE, REGION, STREET_NO, AREA, PRICE, STATUS)

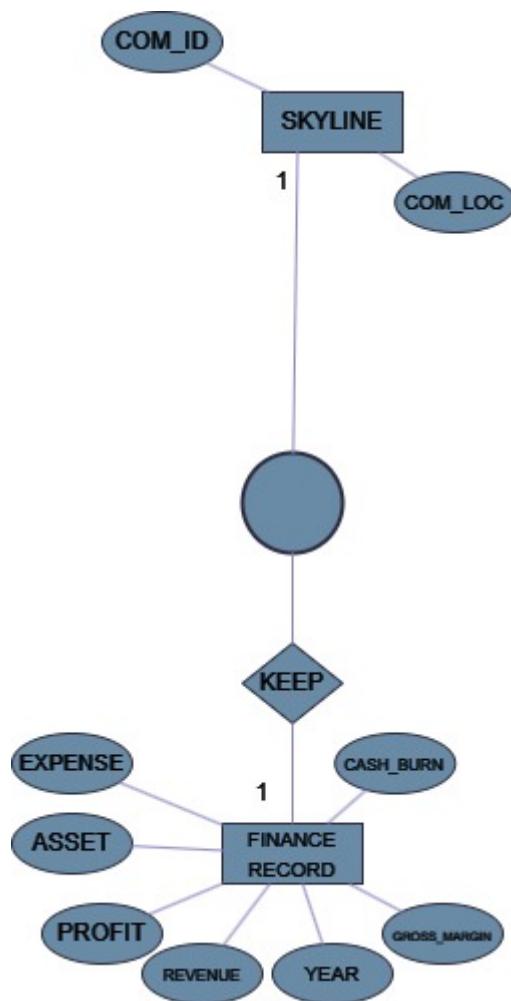
1NF: FIRST_NAME, LAST_NAME

No multivalued attribute

2NF: C_ID, FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, C_DOB, FLAT_CODE, P_TYPE, REGION, STREET_NO, AREA, PRICE, STATUS, C_ID, C_NAME, C_PHONE

3NF: No transitive dependency C_ID, FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, C_DOB, FLAT_CODE, P_TYPE, REGION, STREET_NO, AREA, PRICE, STATUS, C_ID, C_NAME, C_PHONE

Table: C_ID, FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, C_DOB, Flat_Code, P_Type, Region, Street_NO, Area, Price, Status C_ID, C_NAME, C_Phone



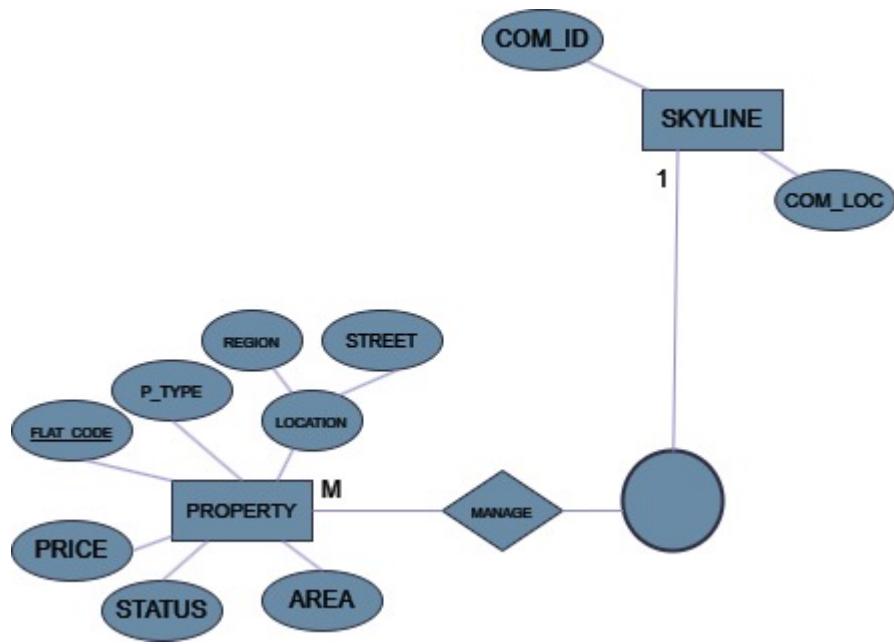
KEEP (Com_ID, Com_Loc, YEAR, REVENUE, PROFIT, GROSS_MARGIN, ASSET, EXPENSES, CASH_BURN)

1NF: No multivalued attribute

2NF: Com_ID, Com_Loc REVENUE, PROFIT, GROSS_MARGIN, ASSET, EXPENSES, CASH_BURN

3NF: Com_ID, Com_Loc REVENUE, PROFIT, GROSS_MARGIN, ASSET, EXPENSES, CASH_BURN
Table: Com_ID, Com_Loc REVENUE, PROFIT, GROSS_MARGIN, ASSET, EXPENSES, CASH_BURN

8.



MANAGE (Com_ID, Com_Loc, Price, P_type, Region, Street_No, Flat_Code, Status, Area)

1NF: Region, Street_No No multivalued attributes

2NF: Com_ID, Com_Loc Price, P_type, Region, Street_No, Flat_Code, Status, Area

3NF: Com_ID, Com_Loc Price, P_type, Region, Street_No, Flat_Code, Status, Area
Table: Com_ID, Com_Loc Price, P_type, Region, Street_No, Flat_Code, Status, Area

Finalization

List Of Tables

- I. SKYLINE
- II. BANK
- III. LOAN
- IV. SHAREHOLDER
- V. DEPARTMENT
- VI. EMPLOYEE
- VII. FINANCE RECORD
- VIII. PROPERTY
- IX. CUSTOMER

Inserted Values in the tables

❖ DEPARTMENT TABLE

DEPT_ID	DEPT_NAME	DEPT_LOC	SUPERVISOR
401	IT	6th floor	Harrison
402	Marketing	4th floor	Hemsworth
403	Management	7th floor	Michel
404	Finance	3rd floor	Georgia
405	HR	2nd floor	Mendes

```
CREATE SEQUENCE dept_seq START WITH 401 INCREMENT BY 1 MAXVALUE 999 NOCYCLE  
NOCACHE;
```

```
CREATE TABLE DEPARTMENT(DEPT_ID NUMBER PRIMARY KEY, DEPT_NAME varchar(200), DEPT_LOC  
varchar(200), SUPERVISOR varchar(500));
```

```
INSERT into DEPARTMENT (DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR) VALUES  
(dept_seq.NEXTVAL, 'IT', '6th floor', 'Harrison');
```

```
INSERT into DEPARTMENT (DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR) VALUES  
(dept_seq.NEXTVAL, 'Marketing', '4th floor', 'Hemsworth');
```

```
INSERT into DEPARTMENT (DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR) VALUES  
(dept_seq.NEXTVAL, 'Management', '7th floor', 'Michel');
```

```
INSERT into DEPARTMENT (DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR) VALUES  
(dept_seq.NEXTVAL, 'Finance', '3rd floor', 'Georgia');
```

```
INSERT into DEPARTMENT (DEPT_ID, DEPT_NAME, DEPT_LOC, SUPERVISOR) VALUES  
(dept_seq.NEXTVAL, 'HR', '2nd floor', 'Mendes');
```

❖ EMPLOYEE TABLE

E_ID	FIRST_NAME	LAST_NAME	EMAIL	JOB	J_STATUS	HIREDATE	BACKGROUND	SALARY	D_ID
1001	Emon	Alhera	alhera@gmail.com	Data Scientist	Full-Time	22-11-2020	BSc. in CSE	47980	401
1002	Rifat	Talukdar	rifat@gmail.com	Web Developer	Full-Time	12-02-2021	BSc. in CSE	46550	401
1003	Jabin	Mitu	mitu@gmail.com	Graphics Designer	Part-Time	11-02-2023	BA. in English	30590	402
1004	Mardia	Mumu	mumu@gmail.com	Finance Manager	Full-Time	31-12-2022	BBA. in Finance	31500	404
1005	Michel	Wayne	wayne@gmail.com	Managing Head	Full-Time	17-09-2020	MBA. in Management	51650	403
1006	Harrison	Styles	harry@gmail.com	IT Head	Full-Time	21-09-2019	MSc. in Data Analytics	63950	401
1007	Hemsworth	Payne	payne@gmail.com	Marketing Head	Full-Time	22-10-2021	MSc. in Psychology	56950	402
1008	Siam	Ahmed	siam@gmail.com	HR Assistant Manager	Part-Time	23-04-2023	BBA. General	24860	405
1009	Georgia	Mallik	mallik@gmail.com	Head of Finance	Full-Time	12-12-2021	MBA. in Finance	65860	404
1010	Mendes	Shawn	mendes@gmail.com	Head of HR	Full-Time	13-05-2022	MBA. in HR Management	57750	405

```
CREATE SEQUENCE emp_seq START WITH 1001 INCREMENT BY 1 MAXVALUE 999999 NOCYCLE
NOCACHE;
```

```
CREATE TABLE EMPLOYEE(E_ID NUMBER PRIMARY KEY NOT NULL, FIRST_NAME varchar(200),
LAST_NAME varchar(500), EMAIL varchar(200), Job varchar(50), J_STATUS varchar(200), HIREDATE
varchar(200), BACKGROUND varchar(200), SALARY varchar(200), D_ID int);
```

```
ALTER TABLE EMPLOYEE ADD FOREIGN KEY (D_ID) REFERENCES DEPARTMENT(DEPT_ID);
```

```
INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE,
BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Emon', 'Alhera', 'alhera@gmail.com',
'Data Scientist', 'Full-Time', '22-11-2020', 'BSc. in CSE', '47980', 401);
```

```
INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, HIREDATE, BACKGROUND,
SALARY, D_ID) VALUES (emp_seq.NEXTVAL, J_STATUS, 'Rifat', 'Talukdar', 'rifat@gmail.com', 'Web
Developer', 'Full-Time', '12-02-2021', 'BSc. in CSE', '46550', 401);
```

```
INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE,
BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Jabin', 'Mitu', 'mitu@gmail.com',
'Graphics Designer', 'Part-Time', '11-02-2023', 'BA. in English', '30590', 402);
```

```
INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Mardia', 'Mumu', 'mumu@gmail.com', 'Finance Manager', 'Full-Time', '31-12-2022', 'BBA. in Finance', '31500', 404);

INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Michel', 'Wayne', 'wayne@gmail.com', 'Managing Head', 'Full-Time', '17-09-2020', 'MBA. in Management', '51650', 403);

INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Harrison', 'Styles', 'harry@gmail.com', 'IT Head', 'Full-Time', '21-09-2019', 'MSc. in Data Analytics', '63950', 401);

INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Hemsworth', 'Payne', 'payne@gmail.com', 'Marketing Head', 'Full-Time', '22-10-2021', 'MSc. in Psychology', '56950', 402);

INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Siam', 'Ahmed', 'siam@gmail.com', 'HR Assistant Manager', 'Part-Time', '23-04-2023', 'BBA. General', '24860', 405);

INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Georgia', 'Mallik', 'mallik@gmail.com', 'Head of Finance', 'Full-Time', '12-12-2021', 'MBA. in Finance', '65860', 404);

INSERT into EMPLOYEE (E_ID, FIRST_NAME, LAST_NAME, EMAIL, Job, J_STATUS, HIREDATE, BACKGROUND, SALARY, D_ID) VALUES (emp_seq.NEXTVAL, 'Mendes', 'Shawn', 'mendes@gmail.com', 'Head of HR', 'Full-Time', '13-05-2022', 'MBA. in HR Management', '57750', 405);
```

❖ BANK TABLE

B_CODE	B_NAME	STREET_NO	APPAR_NO	BRANCH
2462513	AB_Bank_Limited	Block_A Bashundhara	Plot_No. 7	Dhaka
6124532	Dhaka_Bank_Limited	Ward_4 Kotwali	Holding 90	Faridpur
5237131	Dutch-Bangla Bank	Sheikh_Mujib_Road	Holding 153	Chattogram
8462829	Dutch-Bangla Bank	Road_8 Dhanmondi_R/A	House_500-A/1	Dhaka

```
CREATE TABLE BANK (B_CODE NUMBER PRIMARY KEY NOT NULL, B_NAME  
varchar(200), Street_NO varchar(500), APPAR_NO varchar(200), BRANCH varchar(200));
```

```
INSERT into BANK (B_CODE, B_NAME, Street_NO, APPAR_NO, BRANCH) VALUES  
('2462513', 'AB_Bank_Limited', 'Block_A Bashundhara', 'Plot_No. 7', 'Dhaka');  
  
INSERT into BANK (B_CODE, B_NAME, Street_NO, APPAR_NO, BRANCH) VALUES  
('6124532', 'Dhaka_Bank_Limited', 'Ward_4 Kotwali', 'Holding 90', 'Faridpur');  
  
INSERT into BANK (B_CODE, B_NAME, Street_NO, APPAR_NO, BRANCH) VALUES  
('8462829', 'Dutch-Bangla Bank', 'Road_8 Dhanmondi_R/A', 'House_500-A/1', 'Dhaka');  
  
INSERT into BANK (B_CODE, B_NAME, Street_NO, APPAR_NO, BRANCH) VALUES  
('5237131', 'Dutch-Bangla Bank', 'Sheikh_Mujib_Road', 'Holding 153', 'Chattogram');
```

❖ LOAN TABLE

B_CODE	L_TYPE	L_AMOUNT	L_DATE	LM_DATE
2462513	Term Loan	5400000	12-03-2020	13-03-2025
6124532	Working Capital Loan	4000000	23-11-2022	22-11-2023
8462829	Term Loan	1150000	06-07-2019	13-09-2023
5237131	Letter of Credit	2550000	12-03-2020	13-03-2025

```
CREATE TABLE LOAN (B_CODE NUMBER NOT NULL, L_TYPE varchar(200), L_Amount  
varchar(200), L_DATE varchar(200), LM_DATE varchar(200));
```

```
ALTER TABLE LOAN ADD FOREIGN KEY (B_Code) REFERENCES BANK(B_CODE);
```

```
INSERT into LOAN (B_CODE, L_TYPE, L_Amount, L_DATE, LM_DATE) VALUES  
('2462513', 'Term Loan', '5400000', '12-03-2020', '13-03-2025');  
  
INSERT into LOAN (B_CODE, L_TYPE, L_Amount, L_DATE, LM_DATE) VALUES  
('6124532', 'Working Capital Loan', '4000000', '23-11-2022', '22-11-2023');  
  
INSERT into LOAN (B_CODE, L_TYPE, L_Amount, L_DATE, LM_DATE) VALUES  
('8462829', 'Term Loan', '1150000', '06-07-2019', '13-09-2023');  
  
INSERT into LOAN (B_CODE, L_TYPE, L_Amount, L_DATE, LM_DATE) VALUES  
('5237131', 'Letter of Credit', '2550000', '12-03-2020', '13-03-2025');
```

❖ SKYLINE TABLE

COM_ID	COM_LOC
ACME12345	Dhaka

```
CREATE TABLE SKYLINE (Com_ID CHAR(20) PRIMARY KEY NOT  
NULL, Com_Loc VARCHAR(200));
```

```
INSERT into SKYLINE (Com_ID, Com_Loc) VALUES  
('ACME12345', 'Dhaka');
```

❖ SHAREHOLDER TABLE

S_CODE	FIRST_NAME	LAST_NAME	IN_AMOUNT	EQUITY
5759	Akbar	Ali	3500000	6.5%
6173	Zenin	Jarir	2000000	4%
8562	Alif	Adnan	5000000	8%

```
CREATE TABLE SHAREHOLDER (S_CODE NUMBER PRIMARY KEY NOT NULL, FIRST_NAME  
varchar(200), LAST_NAME varchar(200), in_Amount varchar(200), EQUITY  
varchar(200));
```

```
INSERT into SHAREHOLDER (S_CODE, FIRST_NAME, LAST_NAME, in_Amount, EQUITY)  
VALUES ('5759', 'Akbar', 'Ali', '3500000', '6.5%');
```

```
INSERT into SHAREHOLDER (S_CODE, FIRST_NAME, LAST_NAME, in_Amount, EQUITY)  
VALUES ('6173', 'Zenin', 'Jarir', '2000000', '4%');
```

```
INSERT into SHAREHOLDER (S_CODE, FIRST_NAME, LAST_NAME, in_Amount, EQUITY)  
VALUES ('8562', 'Alif', 'Adnan', '5000000', '8%');
```

→ SHR_PHONE

S_CODE	S_NAME	S_PHONE
5759	Akbar_Ali	01983463543
5759	Akbar_Ali	01765436789
6173	Zenin_Jarir	017765483921
6173	Zenin_Jarir	01867545678
8562	Alif_Adnan	01564567893
8562	Alif_Adnan	018189293545

```
CREATE TABLE SHR_Phone (S_CODE NUMBER NOT NULL, S_NAME varchar(200), S_Phone  
varchar(200));
```

```
ALTER TABLE SHR_Phone  
ADD FOREIGN KEY (S_Code) REFERENCES BANK(S_CODE);
```

```
INSERT into SHR_Phone (S_Code, S_NAME, S_Phone) VALUES ('5759', 'Akbar_Ali',  
'01983463543');
```

```
INSERT into SHR_Phone (S_Code, S_NAME, S_Phone) VALUES ('5759', 'Akbar_Ali',  
'01765436789');
```

```
INSERT into SHR_Phone (S_Code, S_NAME, S_Phone) VALUES ('6173',  
'Zenin_Jarir', '017765483921');
```

```
INSERT into SHR_Phone (S_Code, S_NAME, S_Phone) VALUES ('6173',  
'Zenin_Jarir', '01867545678');
```

```
INSERT into SHR_Phone (S_Code, S_NAME, S_Phone) VALUES ('8562',  
'Alif_Adnan', '01564567893');
```

```
INSERT into SHR_Phone (S_Code, S_NAME, S_Phone) VALUES ('8562',  
'Alif_Adnan', '018189293545');
```

❖ FINANCIAL_RECORD TABLE

YEAR	REVENUE	PROFIT	GROSS_MARGIN	ASSET	EXPENSES	CASH_BURN
2019	1,500,000	400,000	35%	5,000,000	1,100,000	200,000
2020	2,000,000	600,000	40%	6,000,000	1,400,000	250,000
2021	2,500,000	800,000	45%	7,500,000	1,700,000	300,000
2022	3,000,000	1,000,000	50%	9,000,000	2,000,000	350,000

```
CREATE TABLE FINANCIAL_RECORD (YEAR NUMBER NOT NULL, REVENUE varchar(200),  
PROFIT varchar(200), GROSS_MARGIN varchar(200), ASSET varchar(200), EXPENSES  
varchar(200), CASH_BURN varchar(200));
```

```
INSERT into FINANCIAL_RECORD (YEAR, REVENUE, PROFIT, GROSS_MARGIN, ASSET,  
EXPENSES, CASH_BURN) VALUES ('2019', '1,500,000', '400,000', '35%',  
'5,000,000', '1,100,000', '200,000');
```

```
INSERT into FINANCIAL_RECORD (YEAR, REVENUE, PROFIT, GROSS_MARGIN, ASSET,  
EXPENSES, CASH_BURN) VALUES ('2020', '2,000,000', '600,000', '40%',  
'6,000,000', '1,400,000', '250,000');
```

```
INSERT into FINANCIAL_RECORD (YEAR, REVENUE, PROFIT, GROSS_MARGIN, ASSET,  
EXPENSES, CASH_BURN) VALUES ('2021', '2,500,000', '800,000', '45%',  
'7,500,000', '1,700,000', '300,000');
```

```
INSERT into FINANCIAL_RECORD (YEAR, REVENUE, PROFIT, GROSS_MARGIN, ASSET,  
EXPENSES, CASH_BURN) VALUES ('2022', '3,000,000', '1,000,000', '50%',  
'9,000,000', '2,000,000', '350,000');
```

❖ CUSTOMER TABLE

C_ID	FIRST_NAME	LAST_NAME	EMAIL	ADDRESS	C_DOB	AGE
6234	Khalid	Kashmiri	kashmiri@gmail.com	Chittagong	1985-06-07	37
7562	Abdul	Aladin	aladin@gmail.com	Dhaka	1981-07-23	41
1672	Mustakim	Rahman	rahman@gmail.com	Sylhet	1979-04-03	44
9803	Ahsam	Haque	ahsam@gmail.com	Comilla	1975-09-12	47

```
CREATE TABLE CUSTOMER (C_ID INT PRIMARY KEY NOT NULL, FIRST_NAME VARCHAR(200),  
LAST_NAME VARCHAR(200), EMAIL VARCHAR(200), ADDRESS VARCHAR(200), C_DOB  
VARCHAR(200), AGE INT);
```

```
INSERT into CUSTOMER (C_ID, FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, C_DOB,  
AGE) VALUES ('7562', 'Abdul', 'Aladin', 'aladin@gmail.com', 'Dhaka', '1981-  
07-23', 41);
```

```
INSERT into CUSTOMER (C_ID, FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, C_DOB,  
AGE) VALUES ('1672', 'Mustakim', 'Rahman', 'rahman@gmail.com', 'Sylhet',  
'1979-04-03', 44);
```

```
INSERT into CUSTOMER (C_ID, FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, C_DOB,  
AGE) VALUES ('6234', 'Khalid', 'Kashmiri', 'kashmiri@gmail.com',  
'Chittagong', '1985-06-07', 37);
```

```
INSERT into CUSTOMER (C_ID, FIRST_NAME, LAST_NAME, EMAIL, ADDRESS, C_DOB,  
AGE) VALUES ('9803', 'Ahsam', 'Haque', 'ahsam@gmail.com', 'Comilla', '1975-  
09-12', 47);
```

→ CMR_PHONE

C_ID	C_NAME	C_PHONE
1672	Mustakim_Rahman	015342678991
6234	Khalid_Kashmiri	01372635478
9803	Ahsam_Haque	018189293545
7562	Abdul_Aladin	01765673892
7562	Abdul_Aladin	01456372819
6234	Khalid_Kashmiri	01867483923

```
CREATE TABLE CMR_Phone (C_ID NUMBER NOT NULL, C_NAME varchar(200), C_Phone  
varchar(200));
```

```
ALTER TABLE CMR_Phone  
ADD FOREIGN KEY (C_ID) REFERENCES CUSTOMER(C_ID);
```

```
INSERT into CMR_Phone (C_ID, C_NAME, C_Phone) VALUES ('1672',  
'Mustakim_Rahman', '015342678991');
```

```
INSERT into CMR_Phone (C_ID, C_NAME, C_Phone) VALUES (6234,  
'Khalid_Kashmiri', '01372635478');
```

```
INSERT into CMR_Phone (C_ID, C_NAME, C_Phone) VALUES (Ahsam_Haque',  
'018189293545');
```

```
INSERT into CMR_Phone (C_ID, C_NAME, C_Phone) VALUES (7562',  
'Abdul_Aladin', '01765673892');
```

```
INSERT into CMR_Phone (C_ID, C_NAME, C_Phone) VALUES (7562',  
'Abdul_Aladin', '01456372819');
```

```
INSERT into SHR_Phone (S_Code, S_NAME, S_Phone) VALUES (6234',  
'Khalid_Kashmiri', '01867483923');
```

❖ TRANSACTION TABLE

T_ID	STATUS	C_ID
903	Purchased	6234
904	Processing	9803
901	Pending	7562
902	Purchased	1672

```
CREATE TABLE TRANSACTION (T_ID NUMBER PRIMARY KEY NOT NULL, Status  
varchar(200), C_ID INT);
```

```
ALTER TABLE TRANSACTION  
ADD FOREIGN KEY (C_ID) REFERENCES CUSTOMER(C_ID);
```

```
INSERT into TRANSACTION (T_ID, Status, C_ID) VALUES ('903', 'Purchased',  
'6234');
```

```
INSERT into TRANSACTION (T_ID, Status, C_ID) VALUES ('904', 'Processing',  
'9803');
```

```
INSERT into TRANSACTION (T_ID, Status, C_ID) VALUES ('901', 'Pending',  
'7562');
```

```
INSERT into TRANSACTION (T_ID, Status, C_ID) VALUES ('902', 'Purchased',  
'1672');
```

❖ PROPERTY TABLE

FLAT_CODE	P_TYPE	REGION	STREET_NO	AREA	PRICE	STATUS
9010	Buy	Baridhara	B-Block	2100 sq-ft	1.05CR	Ready
9011	Rent	Badda	Link_Road	1755 sq-ft	95Lakhs	Ready
9012	Rent	Motijheel	217/51 Road	1877 sq-ft	80Lakhs	On Construction
9013	Buy	Mirpur	DOHS C-Block	1650 sq-ft	75Lakhs	Ready
9014	Buy	Bashundhara	Road-2 A-Block	2450 sq-ft	1.25CR	Ready

```
CREATE TABLE PROPERTY (Flat_Code INT PRIMARY KEY NOT NULL, P_Type
VARCHAR(200), Region VARCHAR(200), Street_NO VARCHAR(200), Area
VARCHAR(200), Price VARCHAR(200), Status VARCHAR(200));
```

```
INSERT into PROPERTY (Flat_Code, P_Type, Region, Street_NO, Area, Price,
Status) VALUES ('9010', 'Buy', 'Baridhara', 'B-Block', '2100 sq-ft',
'1.05CR', 'Ready');
```

```
INSERT into PROPERTY (Flat_Code, P_Type, Region, Street_NO, Area, Price,
Status) VALUES ('9011', 'Rent', 'Badda', 'Link_Road', '1755 sq-ft',
'95Lakhs', 'Ready');
```

```
INSERT into PROPERTY (Flat_Code, P_Type, Region, Street_NO, Area, Price,
Status) VALUES ('9012', 'Rent', 'Motijheel', '217/51 Road', '1877 sq-ft',
'80Lakhs', 'On Construction');
```

```
INSERT into PROPERTY (Flat_Code, P_Type, Region, Street_NO, Area, Price,
Status) VALUES ('9013', 'Buy', 'Mirpur', 'DOHS C-Block', '1650 sq-ft',
'75Lakhs', 'Ready');
```

```
INSERT into PROPERTY (Flat_Code, P_Type, Region, Street_NO, Area, Price,
Status) VALUES ('9014', 'Buy', 'Bashundhara', 'Road-2 A-Block', '2450 sq-
ft', '1.25CR', 'Ready');
```

Query Test in DB

Query Simple:

1) Display all the records in the department table?

select * from DEPARTMENT

```
SELECT * FROM department;
```

Results Explain Describe Saved SQL History

DEPT_ID	DEPT_NAME	DEPT_LOC	SUPERVISOR
401	IT	6th floor	Harrison
402	IT	6th floor	Harrison
403	Marketing	4th floor	Hemsworth
404	IT	6th floor	Harrison
405	Marketing	4th floor	Hemsworth
406	Management	7th floor	Michel
407	Finance	3rd floor	Georgia
408	HR	2nd floor	Mendes
409	HR	2nd floor	Mendes
410	Finance	3rd floor	Georgia

2) Display all the records in the employee table?

select * from EMPLOYEE

E_ID	FIRST_NAME	LAST_NAME	EMAIL	JOB	J_STATUS	HIREDATE	BACKGROUND	SALARY	D_ID
1001	Emon	Alhera	alhera@gmail.com	Data Scientist	Full-Time	22-11-2020	BSc. in CSE	47980	401
1002	Jabin	Mitu	mitu@gmail.com	Graphics Designer	Part-Time	11-02-2023	BA. in English	30590	402
1003	Mardia	Mumu	mumu@gmail.com	Finance Manager	Full-Time	31-12-2022	BBA. in Finance	31500	404
1004	Michel	Wayne	wayne@gmail.com	Managing Head	Full-Time	17-09-2020	MBA. in Management	51650	403
1005	Michel	Wayne	wayne@gmail.com	Managing Head	Full-Time	17-09-2020	MBA. in Management	51650	403
1006	Harrison	Styles	harry@gmail.com	IT Head	Full-Time	21-09-2019	MSc. in Data Analytics	63950	401
1007	Harrison	Styles	harry@gmail.com	IT Head	Full-Time	21-09-2019	MSc. in Data Analytics	63950	401
1008	Hemsworth	Payne	payne@gmail.com	Marketing Head	Full-Time	22-10-2021	MSc. in Psychology	56950	402
1009	Hemsworth	Payne	payne@gmail.com	Marketing Head	Full-Time	22-10-2021	MSc. in Psychology	56950	402
1010	Siam	Ahmed	siam@gmail.com	HR Assistant Manager	Part-Time	23-04-2023	BBA. General	24860	405

3) Display email, job, j_status, hiredate in employee table?

select job,email, j_status, hiredate from EMPLOYEE

```
SELECT email, job, j_status, hiredate  
FROM employee;
```

Results Explain Describe Saved SQL History

EMAIL	JOB	J_STATUS	HIREDATE
alhera@gmail.com	Data Scientist	Full-Time	22-11-2020
mitu@gmail.com	Graphics Designer	Part-Time	11-02-2023
mumu@gmail.com	Finance Manager	Full-Time	31-12-2022
wayne@gmail.com	Managing Head	Full-Time	17-09-2020
wayne@gmail.com	Managing Head	Full-Time	17-09-2020
harry@gmail.com	IT Head	Full-Time	21-09-2019
harry@gmail.com	IT Head	Full-Time	21-09-2019
payne@gmail.com	Marketing Head	Full-Time	22-10-2021
payne@gmail.com	Marketing Head	Full-Time	22-10-2021
siam@gmail.com	HR Assistant Manager	Part-Time	23-04-2023

4) Display all the records in bank table?

select * from BANK

```
SELECT * FROM bank;
```

Results Explain Describe Saved SQL History

B_CODE	B_NAME	STREET_NO	APPAR_NO	BRANCH
2462513	AB_Bank_Limited	Block_A Bashundhara	Plot_No. 7	Dhaka
6124532	Dhaka_Bank_Limited	Ward_4 Kotwali	Holding 90	Faridpur
8462829	Dutch-Bangla Bank	Road_8 Dhanmondi_R/A	House_500-A/1	Dhaka
5237131	Dutch-Bangla Bank	Sheikh_Mujib_Road	Holding 153	Chattogram

5) Display all the records in loan table?

select * from LOAN

```
SELECT * FROM loan;
```

Results Explain Describe Saved SQL History

B_CODE	L_TYPE	L_AMOUNT	L_DATE	LM_DATE
2462513	Term Loan	5400000	12-03-2020	13-03-2025
2462513	Term Loan	5400000	12-03-2020	13-03-2025
6124532	Working Capital Loan	4000000	23-11-2022	22-11-2023
6124532	Working Capital Loan	4000000	23-11-2022	22-11-2023
8462829	Term Loan	1150000	06-07-2019	13-09-2023
8462829	Term Loan	1150000	06-07-2019	13-09-2023
5237131	Letter of Credit	2550000	12-03-2020	13-03-2025
5237131	Letter of Credit	2550000	12-03-2020	13-03-2025
5237131	Letter of Credit	2550000	12-03-2020	13-03-2025
5237131	Letter of Credit	2550000	12-03-2020	13-03-2025

6) Display S_CODE, IN_AMOUNT, EQUITY in SHAREHOLDER table?

select s_code, in_amount, equity from SHAREHOLDER

```
SELECT S_CODE, IN_AMOUNT, EQUITY  
FROM SHAREHOLDER;
```

Results Explain Describe Saved SQL

S_CODE	IN_AMOUNT	EQUITY
5759	3500000	6.5%
6173	2000000	4%
8562	5000000	8%

7) Display REVENUE, PROFIT, GROSS_MARGIN in FINANCIAL_RECORD table?

select revenue, profit, gross_margin from FINANCIAL_RECORD

```
SELECT REVENUE, PROFIT, GROSS_MARGIN  
FROM FINANCIAL_RECORD;
```

Results Explain Describe Saved SQL History

REVENUE	PROFIT	GROSS_MARGIN
1,500,000	400,000	35%
1,500,000	400,000	35%
2,000,000	600,000	40%
2,500,000	800,000	45%
3,000,000	1,000,000	50%

Query Complex:

8) Display email, address, age in the customer table where the first letter of first_name is 'A'?

select email, address, age from customer where first_name like 'A%'

EMAIL	ADDRESS	AGE
aladin@gmail.com	Dhaka	41
ahsam@gmail.com	Comilla	47

9) Display email in the employee table where the last letter of first_name is 'a'?

select email from employee where first_name like '%a'

EMAIL
mallik@gmail.com
mumu@gmail.com

10) Display all employees details who join before 11/02/2023?

select * from EMPLOYEE where HIREDATE<='11/02/2023'

E_ID	FIRST_NAME	LAST_NAME	EMAIL	JOB	J_STATUS	HIREDATE	BACKGROUND	SALARY	D_ID
1020	Jabin	Mitu	mitu@gmail.com	Graphics Designer	Part-Time	11-02-2023	BA. in English	30590	402

11) Display E_ID in the employee table group by E_ID?

select E_ID from employee group by E_ID

E_ID
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010

Sub-Query:

12) Display all the email in the customer table where age is not equal max age?

```
SELECT email FROM customer WHERE age != (SELECT MAX(age) FROM customer)
```

EMAIL
aladin@gmail.com
kashmiri@gmail.com

Joining:

14) Display all the B_CODE that are in LOAN and BANK TABLE?

```
SELECT * FROM BANK JOIN LOAN ON BANK.B_CODE= LOAN.B_CODE
```

B_CODE	B_NAME	STREET_NO	APPAR_NO	BRANCH	B_CODE	L_TYPE	L_AMOUNT	L_DATE	LM_DATE
2462513	AB_Bank_Limited	Block_A_Bashundhara	Plot_No. 7	Dhaka	2462513	Term Loan	5400000	12-03-2020	13-03-2025
8462829	Dutch-Bangla Bank	Road_8_Dhanmondi_R/A	House_500-A/1	Dhaka	8462829	Term Loan	1150000	06-07-2019	13-09-2023
5237131	Dutch-Bangla Bank	Sheikh_Mujib_Road	Holding 153	Chattogram	5237131	Letter of Credit	2550000	12-03-2020	13-03-2025

15) Display all the C_ID that are in CUSTOMER and CMR_PHONE TABLE?

```
SELECT * FROM CUSTOMER JOIN CMR_PHONE ON CUSTOMER.C_ID=CMR_PHONE.C_ID
```

C_ID	FIRST_NAME	LAST_NAME	EMAIL	ADDRESS	C_DOB	AGE	C_ID	C_NAME	C_PHONE
7562	Abdul	Aladin	aladin@gmail.com	Dhaka	1981-07-23	41	7562	Abdul_Aladin	01765673892
7562	Abdul	Aladin	aladin@gmail.com	Dhaka	1981-07-23	41	7562	Abdul_Aladin	01456372819
7562	Abdul	Aladin	aladin@gmail.com	Dhaka	1981-07-23	41	7562	Abdul_Aladin	01765673892
1672	Mustakim	Rahman	rahman@gmail.com	Sylhet	1979-04-03	44	1672	Mustakim_Rahman	015342678991
6234	Khalid	Kashmiri	kashmiri@gmail.com	Chittagong	1985-06-07	37	6234	Khalid_Kashmiri	01372635478
7562	Abdul	Aladin	aladin@gmail.com	Dhaka	1981-07-23	41	7562	Abdul_Aladin	01456372819

Views:

16) Create a view called employees_bio based on the job, background, salary where department id are 401 and 402.

create view employees_bio as select job, background, salary from employee where D_ID=401 or D_ID=402

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEES_BIO	JOB	Varchar2	50	-	-	-	✓	-	-
	BACKGROUND	Varchar2	200	-	-	-	✓	-	-
	SALARY	Varchar2	200	-	-	-	✓	-	-
1 - 3									

JOB	BACKGROUND	SALARY
Data Scientist	BSc. in CSE	47980
Web Developer	BSc. in CSE	46550
Graphics Designer	BA. in English	30590
IT Head	MSc. in Data Analytics	63950
Marketing Head	MSc. in Psychology	56950

17) Create a view called customer_bio based on the email, address and date of birth where c_id are 6234 and 7562.

create view customer_bio as select email, address, c_dob from customer where C_ID=6234 or C_ID= 103

Table	Column	Data Type	Length	Precision	Scale	Primary Key	Nullable	Default	Comment
EMPLOYEES_BIO	JOB	Varchar2	50	-	-	-	✓	-	-
	BACKGROUND	Varchar2	200	-	-	-	✓	-	-
	SALARY	Varchar2	200	-	-	-	✓	-	-
1 - 3									

EMAIL	ADDRESS	C_DOB
kashmiri@gmail.com	Chittagong	1985-06-07

Conclusion

The "Skyline - Real Estate Management System" project is envisioned to bring a paradigm shift in the real estate industry by developing a comprehensive system that simplifies the management and operation of real estate properties. The system focuses on various aspects such as property listing, transaction processing, customer service, payment handling, and data analytics, aiming to enhance the overall experience for both property owners and clients. The project incorporates an extensive database schema consisting of various entities and relationships to model a real estate management system. Here's a summary of the key components, Entities and Relationships: Property, Customer, Bank, Loan, Shareholders etc . The interrelation of Property Owners listing Properties, Properties being managed by Agents, Transactions made by Agents to Clients, and Clients acquiring Properties forms the core of this system. Database Normalization: The database design underwent the normalization process (1NF, 2NF, 3NF) to minimize redundancies and ensure data integrity, decomposing data into structured tables with reduced anomalies. ER Diagram: The ER Diagram visually represents the relationships between various entities, their attributes, and the cardinalities, providing a clear overview of the system's data architecture. Finalization: Tables were finalized with well-defined attributes for each entity, ensuring efficient storage and retrieval of data, such as Property Owners, Properties, Agents, and Clients. Looking ahead, the "Skyline - Real Estate Management System" project has the potential to revolutionize the real estate industry by providing a platform that is not only efficient and user-friendly but also scalable and adaptable to future market trends. The system's robust architecture and comprehensive database schema make it well-equipped to handle the dynamic nature of the real estate market, paving the way for advanced features such as predictive analytics, AI-driven property recommendations, and virtual property tours. The significance of a well-structured and organized database system in managing real estate operations effectively cannot be overstated. By leveraging intelligent design, normalization techniques, and a comprehensive database schema, this system aims to provide a seamless experience for managing real estate operations and enhancing the property acquisition journey for clients. The future of real estate management looks promising with the advent of systems like "Skyline".