NATIONAL INSTITUTE OF TECHNOLOGY, WARANGAL

Department of Computer Science and Engineering

DBMS PROJECT -1

PARTICIPANTS DETAILS:

1.

ANSHU KUMAR

21CSB0A05

2.

UJJWAL KUMAR SINGH

21CSB0B61

TABLE OF CONTENTS:

1. Problem Statement

 2. Key Points and Assumptions

 3. ER Diagram

4. Relational Schema after Normalization

5. Table Structure

6. Functional Dependencies and Primary Keys

7. Normalization

8. SQL code for tables creation and insertion

9. Some Queries

PROBLEM STATEMENT -

Create a database management system that can handle **Online Mobile Shopping** transactions with ease and efficiency. The system should be able to store customer information, product details and payment information securely.

KEY POINTS AND ASSUMPTIONS –

Here are the key points and assumptions for each table:

Table: CUSTOMERS1

Key points: Stores customer information.

Assumptions: Each customer has a unique C_ID. The C_ID is used as the primary key to identify each customer.

Table: PAYMENT

Key points: Stores payment information.

Assumptions: Each payment has a unique PAYMENT_ID. The PAYMENT_ID is used as the primary key to identify each payment. Each payment is associated with an ADMIN_ID, which is a foreign key referencing the ADMINISTRATION table.

Table: MAKE_PAYMENT

Key points: Represents the relationship between customers and payments.

Assumptions: The combination of C_ID and PAYMENT_ID uniquely identifies each record in the MAKE_PAYMENT table. Both C_ID and PAYMENT_ID are foreign keys referencing the CUSTOMERS1 and PAYMENT tables, respectively.

Table: CART

Key points: Stores cart information.

Assumptions: Each cart has a unique CART_ID. The CART_ID is used as the primary key to identify each cart. Each cart is associated with a CUSTOMER_ID and a REGISTRATION_ID, both of which are foreign keys referencing the CUSTOMERS1 and REGISTRATION1 tables, respectively.

Table: ADMINISTRATION

Key points: Stores administration information.

Assumptions: Each administrator has a unique ADMIN_ID. The ADMIN_ID is used as the primary key to identify each administrator.

Table: CAMERA

Key points: Stores camera information.

Assumptions: Each camera has a unique CAMERA_ID. The CAMERA_ID is used as the primary key to identify each camera.

Table: REGISTRATION1

Key points: Represents the registration information for a phone.

Assumptions: Each registration has a unique REGISTRATION_ID. The REGISTRATION_ID is used as the primary key to identify each registration. Each registration is associated with a MODEL_NO

and a CUSTOMER_ID, both of which are foreign keys referencing the PHONE and CUSTOMERS1 tables, respectively.

Table: PROCESSOR

Key points: Stores processor information.

Assumptions: Each processor has a unique PROCESSOR_ID. The PROCESSOR_ID is used as the primary key to identify each processor.

Table: PHONE

Key points: Stores phone information.

Assumptions: Each phone has a unique MODEL_NO. The MODEL_NO is used as the primary key to identify each phone. Each phone is associated with a PROCESSOR_ID, a CAMERA_ID, and an ADMIN_ID, all of which are foreign keys referencing the PROCESSOR, CAMERA, and ADMINISTRATION tables, respectively.

Note: The assumptions mentioned above are based on the table structures provided. Actual assumptions and specifications may vary based on the specific requirements and design considerations of the system.

ER DIAGRAM –
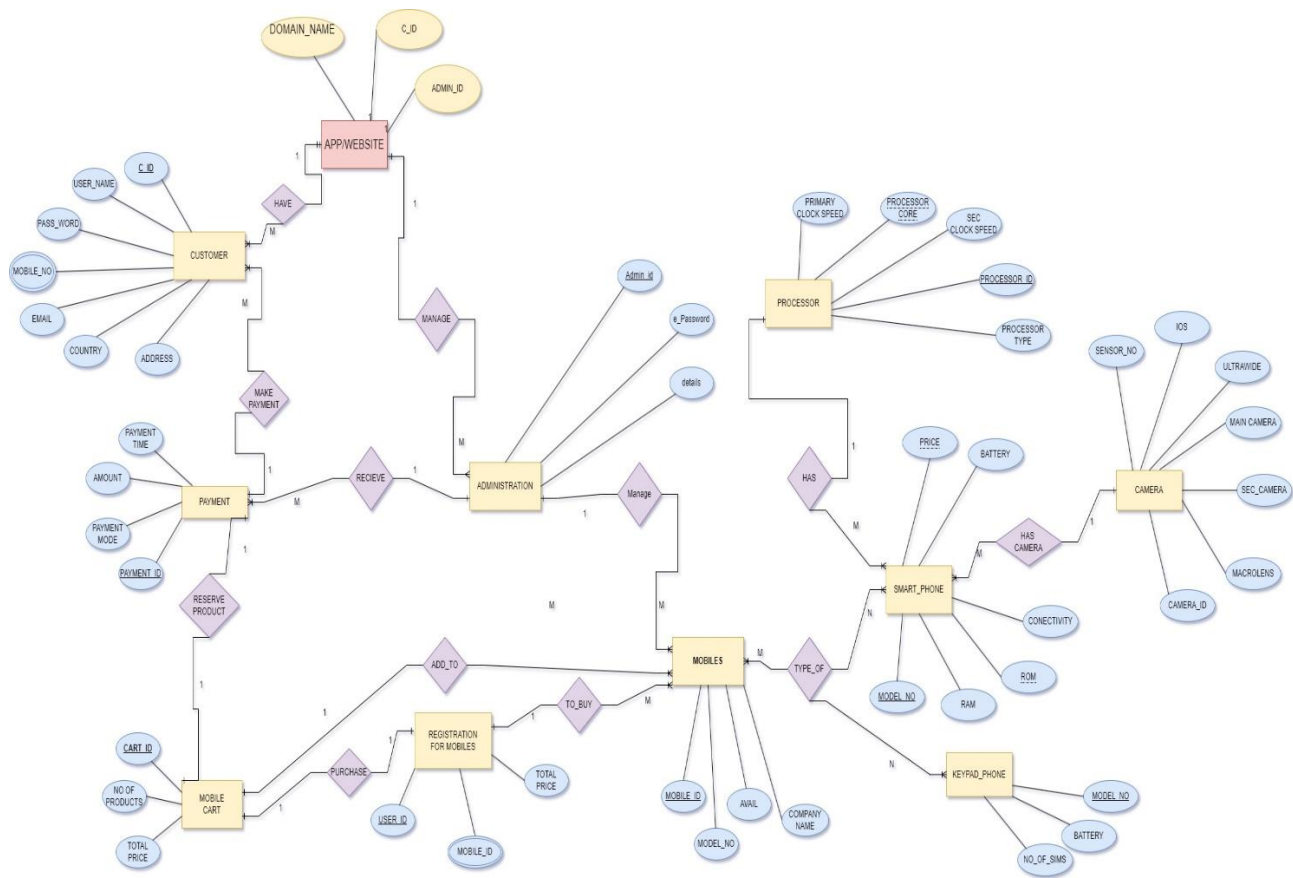
NORMALIZATION_STEP

To normalize the given tables, FOLLOW through the steps:

Step 1: Identify the functional dependencies and primary keys

Table: CUSTOMERS1

- Primary Key: C_ID

Table: PAYMENT

- Primary Key: PAYMENT_ID

- Foreign Key: ADMIN_ID (references ADMINISTRATION table)

Table: MAKE_PAYMENT

- Primary Key: C_ID, PAYMENT_ID

- Foreign Keys: C_ID (references CUSTOMERS1 table), PAYMENT_ID (references PAYMENT table)

Table: CART

- Primary Key: CART_ID

- Foreign Keys: CUSTOMER_ID (references CUSTOMERS1 table), REGISTRATION_ID (references REGISTRATION1 table)

Table: ADMINISTRATION

- Primary Key: ADMIN_ID

Table: CAMERA

- Primary Key: CAMERA_ID

Table: REGISTRATION1

- Primary Key: REGISTRATION_ID

- Foreign Keys: CUSTOMER_ID (references CUSTOMERS1 table), MODEL_NO (references PHONE table)

Table: PROCESSOR

- Primary Key: PROCESSOR_ID

Table: PHONE

- Primary Key: MODEL_NO

- Foreign Keys: PROCESSOR_ID (references PROCESSOR table), CAMERA_ID (references CAMERA table), ADMIN_ID (references ADMINISTRATION table)

Step 2: Eliminate redundant data

•   Review the tables and ensure that there are no repeating groups or arrays within them.

Step 3: First Normal Form (1NF)

•   All tables already satisfy the requirements of 1NF, as they have a primary key that uniquely identifies each row.

Step 4: Second Normal Form (2NF)

•   No partial dependencies exist, so no changes are needed.

Step 5: Third Normal Form (3NF)

•   No transitive dependencies exist, so no changes are needed.

Step 6: Additional Normalization (Higher Normal Forms)

•   The tables appear to be in Third Normal Form, and no further normalization is required based on the given information.

Step 7: Define relationships using foreign keys

•   Foreign keys have already been defined based on the relationships between the tables.

The given tables are already normalized based on the information provided

ENTITY RELATIONSHIP –

AFTER NORMALIZATION
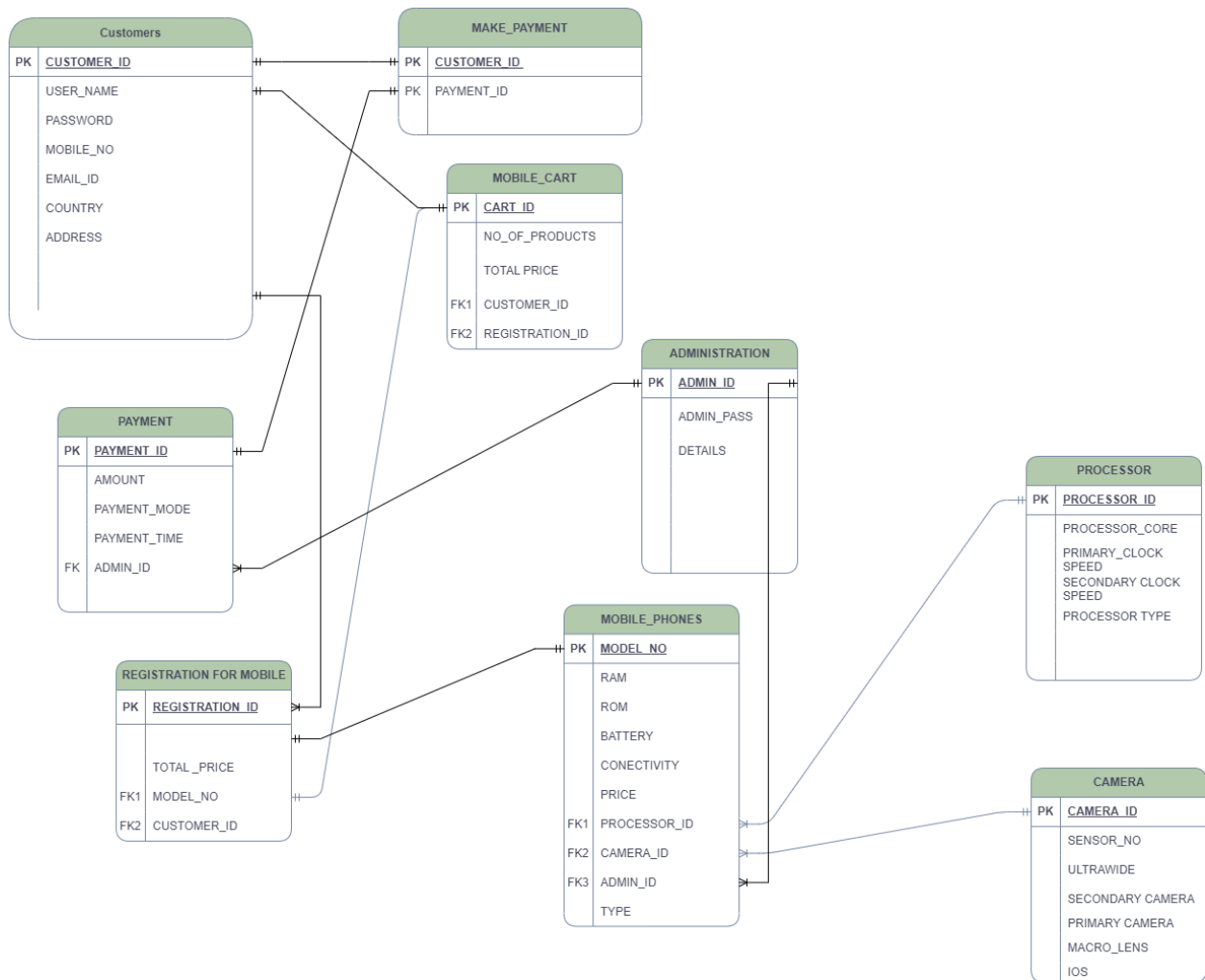
FUNCTIONAL DEPENDENCIES:-

Table: CUSTOMERS1

C_ID -> USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS

Table: PAYMENT

PAYMENT_ID -> AMOUNT, PAYMENT_MODE, PAYMENT_TIME

ADMIN_ID -> No specific functional dependency mentioned

Table: MAKE_PAYMENT

(C_ID, PAYMENT_ID) -> No specific functional dependency mentioned

Table: CART

CART_ID -> NO_OF_PRODUCTS, TOTAL_PRICE, CUSTOMER_ID, REGISTRATION_ID

Table: ADMINISTRATION

ADMIN_ID -> PASSWORD, DETAILS

Table: CAMERA

CAMERA_ID -> MACROLENS, MAINCAMERA, ULTRAWIDE, IOS, SENSOR_NO

Table: REGISTRATION1

REGISTRATION_ID -> MODEL_NO, TOTAL_PRICE, CUSTOMER_ID

Table: PROCESSOR

PROCESSOR_ID -> PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS

Table: PHONE

MODEL_NO -> RAM, ROM, CONECTIVITY, BATTERY, PRICE, PROCESSOR_ID, CAMERA_ID, ADMIN_ID

Note: The functional dependencies mentioned above are based on the provided table structure and assumptions about the relationships between the columns. It's always recommended to analyze the actual data and business requirements to determine the accurate functional dependencies.

TABLE STRUCTURE

1. Customer:

```
Name         Null?    Type
---------  --------  -------------
C_ID        NOT NULL VARCHAR2(10)
USER_NAME            VARCHAR2(20)
PASS                 VARCHAR2(10)
MOBILE_NO            VARCHAR2(15)
EMAIL                VARCHAR2(40)
COUNTRY              VARCHAR2(15)
ADDRESS              VARCHAR2(100)
```

2. PROCESSOR:

```
--------------------  --------  ------------
PROCESSOR_ID          NOT NULL VARCHAR2(10)
PROCESSOR_TYPE                 VARCHAR2(10)
PRIMARY_CLOCK_SPEED             NUMBER(38)
SEC_CLOCK_SPEED                 NUMBER(38)
NO_OF_CORE                      NUMBER(38)
NO_OF_THREADS                   NUMBER(38)
```

3. PHONE

```
-------------  --------  -------------
MODEL_NO       NOT NULL VARCHAR2(15)
RAM                     NUMBER(38)
ROM                     NUMBER(38)
CONECTIVITY             VARCHAR2(10)
BATTERY                 NUMBER(38)
PRICE                   NUMBER(38)
PROCESSOR_ID            VARCHAR2(20)
CAMERA_ID               VARCHAR2(20)
ADMIN_ID                VARCHAR2(10)
TYPE                    VARCHAR2(4)
```

4   RESGISTRATION

```
Name              Null?    Type
---------------- -------- -------------
REGISTRATION_ID NOT NULL VARCHAR2(10)
MODEL_NO                 VARCHAR2(10)
TOTAL_PRICE             NUMBER(38)
CUSTOMER_ID             VARCHAR2(10)
```

5. CART

```
---------------- -------- -------------
CART_ID         NOT NULL VARCHAR2(10)
NO_OF_PRODUCTS          NUMBER(38)
TOTAL_PRICE            NUMBER(38)
CUSTOMER_ID           VARCHAR2(10)
REGISTRATION_ID       VARCHAR2(10)
```

6. PAYMENT

```
PAYMENT_ID    NOT NULL VARCHAR2(15)
AMOUNT                NUMBER(38)
PAYMENT_MODE         VARCHAR2(10)
PAYMENT_TIME         VARCHAR2(10)
ADMIN_ID            VARCHAR2(10)
```

7. MAKE_PAYMENT

```
---------- -------- -------------
C_ID       NOT NULL VARCHAR2(10)
PAYMENT_ID NOT NULL VARCHAR2(15)
```

8. ADMINISTRATION

```
-------- -------- --------------
ADMIN_ID NOT NULL VARCHAR2(10)
PASSWORD         VARCHAR2(20)
DETAILS          VARCHAR2(200)
```

9. CAMERA

```
---------- -------- -------------
CAMERA_ID  NOT NULL VARCHAR2(10)
MACROLENS          NUMBER(38)
MAINCAMERA         NUMBER(38)
ULTRAWIDE          NUMBER(38)
IOS                VARCHAR2(3)
SENSOR_NO          VARCHAR2(10)
```

SQL CODE FOR TABLE CREATION AND INSERTION

```sql
CREATE TABLE CUSTOMERS1(
    C_ID VARCHAR(10) NOT NULL,
    USER_NAME VARCHAR(20),
    PASS VARCHAR(10),
    MOBILE_NO VARCHAR(15),
    EMAIL VARCHAR(40),
    COUNTRY VARCHAR(15),
    ADDRESS VARCHAR(100),
    PRIMARY KEY(C_ID)
);
CREATE TABLE PAYMENT(
    PAYMENT_ID VARCHAR(15),
    AMOUNT INT,
    PAYMENT_MODE VARCHAR(10),
    PAYMENT_TIME VARCHAR(10),
    PRIMARY KEY(PAYMENT_ID),
    ADMIN_ID VARCHAR(10),
    FOREIGN KEY (ADMIN_ID) REFERENCES ADMINISTRATION
);
CREATE TABLE MAKE_PAYMENT(
    C_ID VARCHAR(10),
    PAYMENT_ID VARCHAR(15),
    PRIMARY KEY (C_ID,PAYMENT_ID)
);
DROP TABLE MAKE_PAYMENT;
CREATE TABLE CART(
    CART_ID VARCHAR(10),
```

```sql
   NO_OF_PRODUCTS INT,

   TOTAL_PRICE INT,

   PRIMARY KEY(CART_ID),

   CUSTOMER_ID VARCHAR(10),

   REGISTRATION_ID VARCHAR(10),

   FOREIGN KEY (REGISTRATION_ID) REFERENCES REGISTRATION1,

   FOREIGN KEY (CUSTOMER_ID) REFERENCES CUSTOMERS1
);
CREATE TABLE ADMINISTRATION(

   ADMIN_ID VARCHAR(10) NOT NULL,

   PASSWORD VARCHAR(20),

   DETAILS VARCHAR(200),

   PRIMARY KEY(ADMIN_ID)
);
CREATE TABLE CAMERA(

   CAMERA_ID VARCHAR(10) NOT NULL,

   MACROLENS INTEGER,

   MAINCAMERA INTEGER,

   ULTRAWIDE INTEGER,

   IOS VARCHAR(3),

   SENSOR_NO VARCHAR(10),

   PRIMARY KEY(CAMERA_ID)
);
CREATE TABLE REGISTRATION1(

REGISTRATION_ID VARCHAR(10),

MODEL_NO VARCHAR(10),

TOTAL_PRICE INT,

CUSTOMER_ID VARCHAR(10),

FOREIGN KEY (CUSTOMER_ID) REFERENCES CUSTOMERS1,

FOREIGN KEY (MODEL_NO) REFERENCES PHONE,

PRIMARY KEY (REGISTRATION_ID)
```

```sql
);




CREATE TABLE PROCESSOR(

    PROCESSOR_ID VARCHAR(10) NOT NULL,

    PROCESSOR_TYPE VARCHAR(10),

    PRIMARY_CLOCK_SPEED INT,

    SEC_CLOCK_SPEED INT,

    NO_OF_CORE INT,

    NO_OF_THREADS INT,

    PRIMARY KEY (PROCESSOR_ID)

);

CREATE TABLE PHONE(

    MODEL_NO VARCHAR(15) NOT NULL,

    RAM INT,

    ROM INT,

    CONECTIVITY VARCHAR(10),

    BATTERY INT,

    PRICE INT,

    PROCESSOR_ID VARCHAR(20),

    CAMERA_ID VARCHAR(20),

    ADMIN_ID VARCHAR(10),

    PRIMARY KEY(MODEL_NO),

    TYPE VARCHAR(4),

    FOREIGN KEY(PROCESSOR_ID) REFERENCES PROCESSOR,

    FOREIGN KEY(CAMERA_ID) REFERENCES CAMERA,

    FOREIGN KEY(ADMIN_ID) REFERENCES ADMINISTRATION

);
```

CAMERA INSERTION:

INSERT INTO CAMERA VALUES ('e9X2R7', 16, 8, 4, 'No', 'S78901');

INSERT INTO CAMERA (CAMERA_ID, MACROLENS, MAINCAMERA, ULTRAWIDE, IOS, SENSOR_NO)

VALUES ('K5Y8U2', 12, 12, 8, 'Yes', 'R23456');

INSERT INTO CAMERA (CAMERA_ID, MACROLENS, MAINCAMERA, ULTRAWIDE, IOS, SENSOR_NO)

VALUES ('N3A9Z0', 20, 16, 8, 'No', 'A34567');

INSERT INTO CAMERA (CAMERA_ID, MACROLENS, MAINCAMERA, ULTRAWIDE, IOS, SENSOR_NO)

VALUES ('P4Q6X8', 12, 12, 0, 'Yes', 'C45678');

INSERT INTO CAMERA (CAMERA_ID, MACROLENS, MAINCAMERA, ULTRAWIDE, IOS, SENSOR_NO)

VALUES ('R8Y2N6', 18, 10, 6, 'Yes', 'D56789');

INSERT INTO CAMERA (CAMERA_ID, MACROLENS, MAINCAMERA, ULTRAWIDE, IOS, SENSOR_NO)

VALUES ('S1B5Q9', 14, 12, 10, 'No', 'E67890');

INSERT INTO CAMERA (CAMERA_ID, MACROLENS, MAINCAMERA, ULTRAWIDE, IOS, SENSOR_NO)

VALUES ('T3X7C1', 20, 16, 8, 'Yes', 'F12345');

INSERT INTO CAMERA (CAMERA_ID, MACROLENS, MAINCAMERA, ULTRAWIDE, IOS, SENSOR_NO)

VALUES ('U4Z8D2', 12, 12, 0, 'No', 'G23456');

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P3', 'TypeA', 3200, 2800, 6, 12);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P4', 'TypeB', 2800, 2400, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P22', 'TypeC', 3000, 2500, 8, 16);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P23', 'TypeA', 3400, 3000, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P42', 'TypeB', 3200, 2800, 6, 12);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P43', 'TypeC', 2800, 2400, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P21', 'TypeA', 3000, 2500, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P22', 'TypeB', 3200, 2800, 6, 12);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P23', 'TypeC', 2800, 2400, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P40', 'TypeD', 3400, 3000, 8, 16);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P41', 'TypeA', 2800, 2400, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P58', 'TypeE', 3200, 2800, 6, 12);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P59', 'TypeF', 3000, 2500, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P24', 'TypeD', 3400, 3000, 8, 16);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P25', 'TypeE', 3200, 2800, 6, 12);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P26', 'TypeF', 3000, 2500, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P44', 'TypeG', 3600, 3200, 6, 12);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P45', 'TypeH', 3200, 2800, 4, 8);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P62', 'TypeI', 3000, 2500, 6, 12);

INSERT INTO PROCESSOR (PROCESSOR_ID, PROCESSOR_TYPE, PRIMARY_CLOCK_SPEED, SEC_CLOCK_SPEED, NO_OF_CORE, NO_OF_THREADS)

VALUES ('P63', 'TypeJ', 2800, 2400, 4, 8);

INSERTING CUSTOMER DATA

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES

('C1', 'JohnDoe', 'pass123', '1234567890', 'john.doe@example.com', 'USA', '123 Main St');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES

('C2', 'JaneSmith', 'abc456', '9876543210', 'jansmith@example.com', 'Canada', '456 Elm St');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES

('C3', 'MikeJohnson', 'mikepass', '5555555555', 'mins@example.com', 'UK', '789 Oak Ave');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES

('C4', 'EmilyBrown', 'pass789', '1112223333', 'em.bwn@example.com', 'Australia', '321 Pine Rd');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES

('C5', 'DavidWilson', 'davidpass', '4444444444', 'dad.wilson@example.com', 'Germany', '654 Cedar Ln');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES

('C6', 'SarahTaylor', 'pass123', '9999999999', 'sar.taylor@example.com', 'France', '987 Maple Ave');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C7', 'RobertLee', 'robertpass', '6666666666', 'rort.lee@example.com', 'China', '852 Willow Rd');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C8', 'JenniferClark', 'pass789', '3333333333', 'jennr.clark@example.com', 'Brazil', '159 Oak St');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C9', 'DanielMiller', 'danpass', '7777777777', 'danl.miller@example.com', 'Mexico', '753 Pine Ave');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C10', 'LauraDavis', 'pass123', '2222222222', 'laa.davis@example.com', 'Spain', '369 Cedar Ln');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C11', 'MichaelWang', 'mikepass', '8888888888', 'micl.wang@example.com', 'Japan', '741 Elm St');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C12', 'JessicaLiu', 'pass789', '4445556666', 'jesa.liu@example.com', 'South Korea', '963 Willow Rd');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C13', 'RyanNguyen', 'ryanpass', '9998887777', 'rn.nguyen@example.com', 'Vietnam', '258 Oak St');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C14', 'EmilyChen', 'pass123', '1119993333', 'emy.chen@example.com', 'Thailand', '654 Pine Rd');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C15', 'ChristopherKim', 'chrispass', '7776665555', 'chrpher.kim@example.com', 'Philippines', '987 Cedar Ln');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C16', 'MichelleGupta', 'pass789', '3332221111', 'mile.gupta@example.com', 'India', '321 Maple Ave');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C17', 'MatthewKhan', 'mattpass', '6663339999', 'matw.khan@example.com', 'Pakistan', '741 Willow Rd');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C18', 'SamanthaAli', 'pass123', '2225558888', 'saha.ali@example.com', 'Egypt', '963 Oak St');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C19', 'KevinZhang', 'pass789', '8885552222', 'ken.zhang@example.com', 'Russia', '852 Pine Rd');

INSERT INTO CUSTOMERS1 (C_ID, USER_NAME, PASS, MOBILE_NO, EMAIL, COUNTRY, ADDRESS) VALUES ('C20', 'AmandaSanchez', 'amandapass', '1114447777', 'aa.sanchez@example.com', 'Argentina', '159 Cedar')

INSERT INTO PHONE VALUES

('model1', 4, 64, '5G', 4000, 20000, 'P26', 'K5Y8U2', 'admin1','A');

INSERT INTO PHONE VALUES

('model2', 6, 128, '5G', 4500, 25000, 'P24', 'D9X2R7', 'admin2','B');

INSERT INTO PHONE VALUES

('model3', 8, 256, '5G', 5000, 30000, 'P26', 'K5Y8U2', 'admin3','C');

INSERT INTO PHONE VALUES

('model4', 4, 64, '4G', 4000, 15000, 'P24', 'S1B5Q9', 'admin4','A');

INSERT INTO PHONE VALUES

('model5', 6, 128, '4G', 4500, 20000, 'P44', 'D9X2R7', 'admin2','C');

INSERT INTO PHONE VALUES

('model6', 8, 256, '4G', 5000, 25000, 'P45', 'K5Y8U2', 'admin3','A');

INSERT INTO PHONE VALUES

('model7', 4, 64, '3G', 4000, 10000, 'P63', 'S1B5Q9', 'admin1','C');

INSERT INTO PHONE VALUES

('model8', 6, 128, '3G', 4500, 15000,'P44','D9X2R7','admin2','D');

INSERT INTO PHONE VALUES

('model9',8 ,256 ,'3G' ,5000 ,20000 ,'P62' ,'K5Y8U2' ,'admin4','E');

INSERT INTO PHONE VALUES

('model10' ,4 ,64 ,'2G' ,4000 ,5000 ,'P26' ,'S1B5Q9' ,'admin1','A');

INSERT INTO PHONE VALUES

('model11' ,6 ,128 ,'2G' ,4500 ,7500 ,'P44' ,'D9X2R7' ,'admin1','A');

INSERT INTO PHONE VALUES

('model12' ,8 ,256 ,'2G' ,5000 ,10000 ,'P63' ,'e9X2R7' ,'admin2','D');

INSERT INTO PHONE VALUES

('model13' ,4 ,64 ,'5G' ,4000 ,20000 ,'P45' ,'K5Y8U2' ,'admin3','A');

INSERT INTO PHONE VALUES

('model14' ,6 ,128 ,'5G' ,4500 ,25000 ,'P24' ,'D9X2R7' ,'admin4','A');

INSERT INTO PHONE VALUES

('model15' ,8 ,256 ,'5G' ,5000 ,30000 ,'P25' ,'camera15' ,'admin5','B');

INSERT INTO PHONE VALUES

('model16' ,4 ,64 ,'4G' ,4000 ,15000 ,'P3' ,'D9X2R7','admin1','D');

INSERT INTO PHONE VALUES

('model17' ,6 ,128,'4G' ,4500 ,20000,'P4','K5Y8U2','admin1','A');

INSERT INTO PHONE VALUES

('model18' ,8 ,256,'4G' ,5000 ,25000,'P22','R8Y2N6','admin3','C');

INSERT INTO PHONE VALUES

('model19' ,4 ,64,'3G' ,4000 ,10000,'P23','T3X7C1','admin1','A');

INSERT INTO PHONE VALUES

('model20' ,6 ,128,'3G' ,4500 ,15000,'P21','D9X2R7','admin2','B');

INSERTING THE VALUE IN ADMIN

INSERT INTO ADMINISTRATION (ADMIN_ID, PASSWORD, DETAILS) VALUES

('admin1', 'password123', 'Administrator 1 details');

INSERT INTO ADMINISTRATION (ADMIN_ID, PASSWORD, DETAILS) VALUES

('admin2', 'pass456', 'Administrator 2 details');

INSERT INTO ADMINISTRATION (ADMIN_ID, PASSWORD, DETAILS) VALUES

('admin3', 'adminpass', 'Administrator 3 details');

INSERT INTO ADMINISTRATION (ADMIN_ID, PASSWORD, DETAILS) VALUES

('admin4', 'securepass', 'Administrator 4 details');

INSERT INTO ADMINISTRATION (ADMIN_ID, PASSWORD, DETAILS) VALUES

('admin5', '123456', 'Administrator 5 details');

INSERTING THE PAYMENT TABLE

INSERT INTO PAYMENT VALUES

('Payment1', 1000, 'Credit', '12:30 PM', 'admin1');

INSERT INTO PAYMENT VALUES

('Payment2', 500, 'Debit', '02:45 PM', 'admin2');

INSERT INTO PAYMENT VALUES

('Payment3', 750, 'Cash', '09:15 AM', 'admin3');

INSERT INTO PAYMENT VALUES

('Payment4', 2000, 'Credit', '04:20 PM', 'admin4');

INSERT INTO PAYMENT VALUES

('Payment5', 1500, 'Debit', '08:00 AM', 'admin5');

INSERT INTO PAYMENT VALUES

('Payment6', 1200, 'Cash', '11:45 AM', 'admin1');

INSERT INTO PAYMENT VALUES

('Payment7', 900, 'Credit', '06:10 PM', 'admin2');

INSERT INTO PAYMENT VALUES

('Payment8', 800, 'Debit', '01:30 PM', 'admin3');

INSERT INTO PAYMENT VALUES

('Payment9', 2500, 'Cash', '10:55 AM', 'admin4');

INSERT INTO PAYMENT VALUES

('Payment10', 1800, 'Credit', '03:15 PM', 'admin5');

MAKE_PAYMENT_TABLE

--INSERT INTO MAKE_PAYMENT VALUES

--('C1', 'Payment10');

--INSERT INTO MAKE_PAYMENT (C_ID, PAYMENT_ID) VALUES

--('C10', 'Payment4');

--INSERT INTO MAKE_PAYMENT (C_ID, PAYMENT_ID) VALUES

--('C13', 'Payment3');

--INSERT INTO MAKE_PAYMENT (C_ID, PAYMENT_ID) VALUES

--('C17', 'Payment10');

INSERTING THE VALUES IN REGISTRATION:

INSERT INTO REGISTRATION1 VALUES

('Reg1', 'model5', 500, 'C1');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg2', 'model2', 800, 'C2');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg3', 'model3', 700, 'C3');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg4', 'model4', 600, 'C4');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg5', 'model5', 900, 'C5');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg6', 'model6', 750, 'C6');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg7', 'model7', 550, 'C1');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg8', 'model8', 850, 'C3');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg9', 'model9', 680, 'C9');

INSERT INTO REGISTRATION1 (REGISTRATION_ID, MODEL_NO, TOTAL_PRICE, CUSTOMER_ID) VALUES ('Reg10', 'model10', 520, 'C10');

TABLE DATA: (QUERY)

1.

SELECT* FROM PROCESSOR;

```
PROCESSOR_  PROCESSOR_  PRIMARY_CLOCK_SPEED SEC_CLOCK_SPEED NO_OF_CORE NO_OF_THREADS
----------  ----------  ------------------- --------------- ---------- -------------
P3          TypeA                      3200            2800          6            12
P4          TypeB                      2800            2400          4             8
P22         TypeC                      3000            2500          8            16
P23         TypeA                      3400            3000          4             8
P42         TypeB                      3200            2800          6            12
P43         TypeC                      2800            2400          4             8
P21         TypeA                      3000            2500          4             8
P40         TypeD                      3400            3000          8            16
P41         TypeA                      2800            2400          4             8
P58         TypeE                      3200            2800          6            12
P59         TypeF                      3000            2500          4             8

PROCESSOR_  PROCESSOR_  PRIMARY_CLOCK_SPEED SEC_CLOCK_SPEED NO_OF_CORE NO_OF_THREADS
----------  ----------  ------------------- --------------- ---------- -------------
P24         TypeD                      3400            3000          8            16
P25         TypeE                      3200            2800          6            12
P26         TypeF                      3000            2500          4             8
P44         TypeG                      3600            3200          6            12
P45         TypeH                      3200            2800          4             8


P45         TypeH                      3200            2800          4             8
P62         TypeI                      3000            2500          6            12
P63         TypeJ                      2800            2400          4             8

18 rows selected.
```

2.

SELECT* FROM CAMERA;

```
CAMERA_ID    MACROLENS MAINCAMERA   ULTRAWIDE IOS SENSOR_NO
---------- ----------- ----------- ----------- --- ----------
D9X2R7              16           8           4 No  S78901
K5Y8U2              12          12           8 Yes R23456
N3A9Z0              20          16           8 No  A34567
P4Q6X8              12          12           0 Yes C45678
R8Y2N6              18          10           6 Yes D56789
S1B5Q9              14          12          10 No  E67890
T3X7C1              20          16           8 Yes F12345
U4Z8D2              12          12           0 No  G23456
e9X2R7              16           8           4 No  S78901

9 rows selected.
```
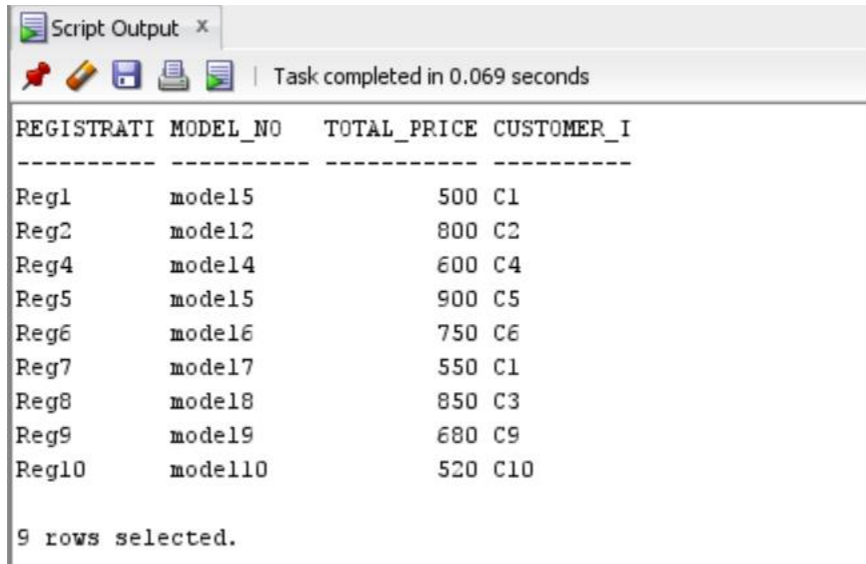
3. SELECT* FROM PHONE;

Script Output ×

Task completed in 0.057 seconds

MODEL_NO	RAM	ROM	CONECTIVIT	BATTERY	PRICE	PROCESSOR_ID	CAMERA_ID	ADMIN_ID	TYPE
model2	6	128	5G	4500	25000	P24	D9X2R7	admin2	B
model4	4	64	4G	4000	15000	P24	S1B5Q9	admin4	A
model1	4	64	5G	4000	20000	P26	K5Y8U2	admin1	A
model5	6	128	4G	4500	20000	P44	D9X2R7	admin2	C
model6	8	256	4G	5000	25000	P45	K5Y8U2	admin3	A
model7	4	64	3G	4000	10000	P63	S1B5Q9	admin1	C
model8	6	128	3G	4500	15000	P44	D9X2R7	admin2	D
model9	8	256	3G	5000	20000	P62	K5Y8U2	admin4	E
model10	4	64	2G	4000	5000	P26	S1B5Q9	admin1	A
model11	6	128	2G	4500	7500	P44	D9X2R7	admin1	A
model12	8	256	2G	5000	10000	P63	e9X2R7	admin2	D

MODEL_NO	RAM	ROM	CONECTIVIT	BATTERY	PRICE	PROCESSOR_ID	CAMERA_ID	ADMIN_ID	TYPE
model13	4	64	5G	4000	20000	P45	K5Y8U2	admin3	A
model14	6	128	5G	4500	25000	P24	D9X2R7	admin4	A
model16	4	64	4G	4000	15000	P3	D9X2R7	admin1	D
model17	6	128	4G	4500	20000	P4	K5Y8U2	admin1	A
model18	8	256	4G	5000	25000	P22	R8Y2N6	admin3	C

4. **SELECT* FROM REGISTRATION1;**

```
Script Output  x

Task completed in 0.069 seconds

REGISTRATI MODEL_NO    TOTAL_PRICE CUSTOMER_I
---------- ----------  ----------- ----------
Reg1       model5              500 C1
Reg2       model2              800 C2
Reg4       model4              600 C4
Reg5       model5              900 C5
Reg6       model6              750 C6
Reg7       model7              550 C1
Reg8       model8              850 C3
Reg9       model9              680 C9
Reg10      model10             520 C10

9 rows selected.
```

5. **SELECT* FROM CART;**

```
CART_ID    NO_OF_PRODUCTS TOTAL_PRICE CUSTOMER_I REGISTRATI
---------- -------------- ----------- ---------- ----------
Cart1                   3         150 C1         Reg1
Cart2                   2         100 C2         Reg2
Cart4                   1          50 C4         Reg4
Cart5                   4         200 C5         Reg5
Cart6                   2         100 C6         Reg6
Cart7                   3         150 C7         Reg7

6 rows selected.
```

6. **SELECT* FROM PAYMENT;**

```
Script Output  ×

📌 ✏ 💾 🖨 ▣ | Task completed in 0.034 seconds

PAYMENT_ID          AMOUNT PAYMENT_MO PAYMENT_TI ADMIN_ID
--------------- ---------- ---------- ---------- ----------
Payment1          | 1000 Credit     12:30 PM    admin1
Payment2             500 Debit      02:45 PM    admin2
Payment3             750 Cash       09:15 AM    admin3
Payment4            2000 Credit     04:20 PM    admin4
Payment5            1500 Debit      08:00 AM    admin5
Payment6            1200 Cash       11:45 AM    admin1
Payment7             900 Credit     06:10 PM    admin2
Payment8             800 Debit      01:30 PM    admin3
Payment9            2500 Cash       10:55 AM    admin4
Payment10           1800 Credit     03:15 PM    admin5
```

7. **SELECT* FROM MAKE_PAYMENT;**

```
Script Output  ×

📌 ✏ 💾 🖨 ▣ | Task completed in 0.059 seconds

C_ID       PAYMENT_ID
---------- ---------------
C1         Payment10
C10        Payment4
C13        Payment3
C17        Payment10
```

8. **SELECT* FROM ADMINISTRATION;**

```
ADMIN_ID   PASSWORD            DETAILS
---------- ------------------- -----------------------------------
admin1     password123         Administrator 1 details
admin2     pass456             Administrator 2 details
admin3     adminpass           Administrator 3 details
admin4     securepass          Administrator 4 details
admin5     123456              Administrator 5 details
```

9. SELECT* FROM customers1;

C_ID	USER_NAME	PASS	MOBILE_NO	EMAIL	COUNTRY	ADDRESS
C3	MikeJohnson	mikepass	5555555555	mins@example.com	UK	789 Oak Ave
C4	EmilyBrown	pass789	1112223333	em.bwn@example.com	Australia	321 Pine Rd
C5	DavidWilson	davidpass	4444444444	dad.wilson@example.com	Germany	654 Cedar Ln
C6	SarahTaylor	pass123	9999999999	sar.taylor@example.com	France	987 Maple Ave
C7	RobertLee	robertpass	6666666666	rort.lee@example.com	China	852 Willow Rd
C8	JenniferClark	pass789	3333333333	jennr.clark@example.com	Brazil	159 Oak St
C9	DanielMiller	danpass	7777777777	danl.miller@example.com	Mexico	753 Pine Ave
C10	LauraDavis	pass123	2222222222	laa.davis@example.com	Spain	369 Cedar Ln
C11	MichaelWang	mikepass	8888888888	micl.wang@example.com	Japan	741 Elm St
C12	JessicaLiu	pass789	4445556666	jesa.liu@example.com	South Korea	963 Willow Rd
C13	RyanNguyen	ryanpass	9998887777	rn.nguyen@example.com	Vietnam	258 Oak St

C_ID	USER_NAME	PASS	MOBILE_NO	EMAIL	COUNTRY	ADDRESS
C14	EmilyChen	pass123	1119993333	emy.chen@example.com	Thailand	654 Pine Rd
C15	ChristopherKim	chrispass	7776665555	chrpher.kim@example.com	Philippines	987 Cedar Ln
C16	MichelleGupta	pass789	3332221111	mile.gupta@example.com	India	321 Maple Ave
C17	MatthewKhan	mattpass	6663339999	matw.khan@example.com	Pakistan	741 Willow Rd
C19	KevinZhang	pass789	8885552222	ken.zhang@example.com	Russia	852 Pine Rd
C20	AmandaSanchez	amandapass	1114447777	aa.sanchez@example.com	Argentina	159 Cedar
C1	JohnDoe	pass123	1234567890	john.doe@example.com	USA	123 Main St
C2	JaneSmith	abc456	9876543210	jansmith@example.com	Canada	456 Elm St

20 rows selected.