LAB PRACTICAL RECORD

On

Introduction to Database Managment System

ETCA851A

K.R MANGLAM UNIVERSITY

Sohna Road, Haryana

AUG 2021 – DEC 2021

Submitted To:

MS. ASHA SOHAL

Submitted By:

NAME: Devendra Kumar

Roll no: 2101560018

MCA, 1st Sem

Courier-Management- System

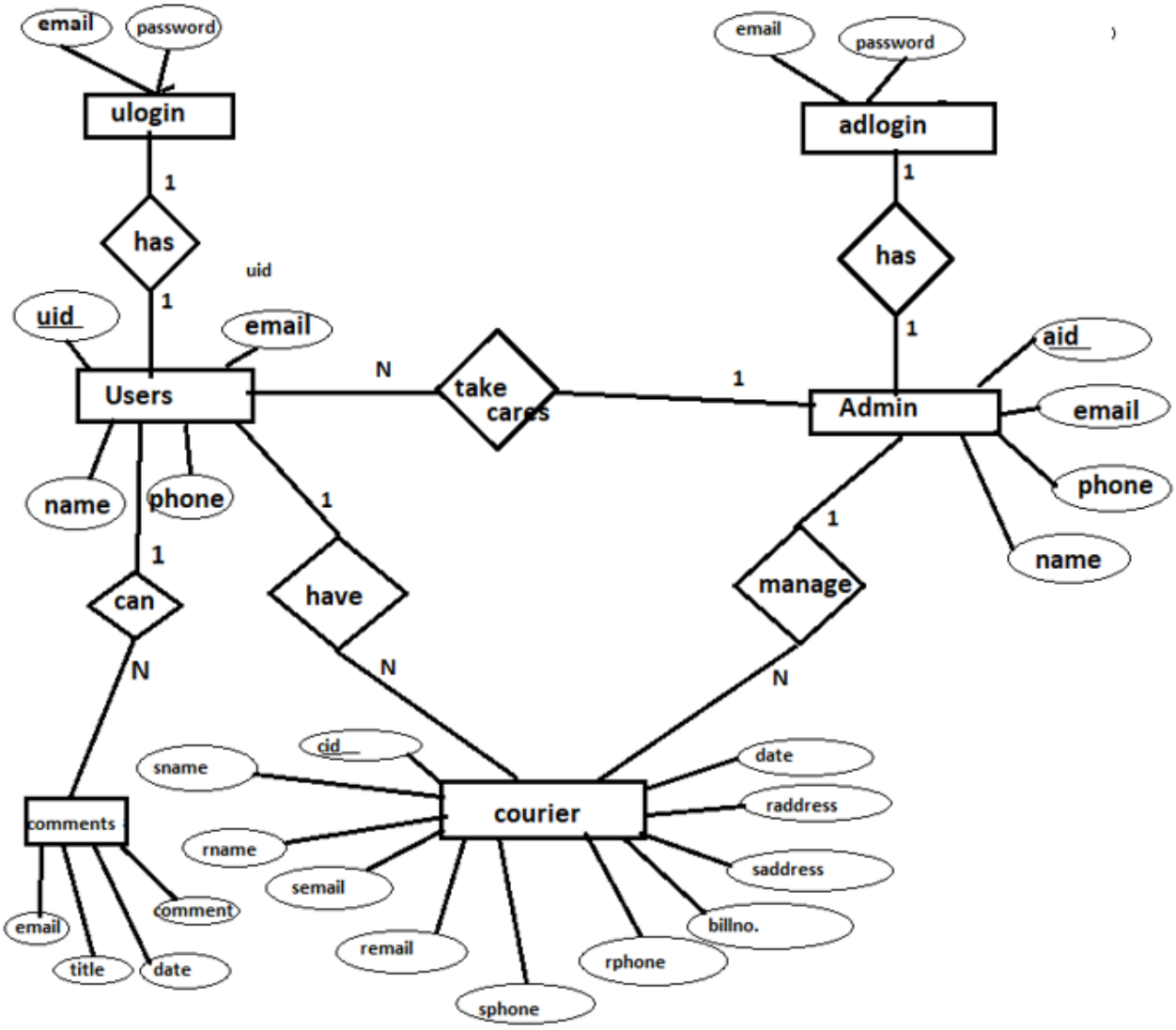
Introduction: -

This Courier Management System Project will have different modules. The login section will have login facility for the admin and for the user who will operate this system. While taking orders from its customers, it will take all the details of its customers who is placing the orders and all the details for the recipient such as its address, name, mobile number. During billing process system will generate a tracking id for their products. Through this tracking id, customers or its recipient will able to track their products from any location using internet. It will provide status of the product after placing orders within I minute.

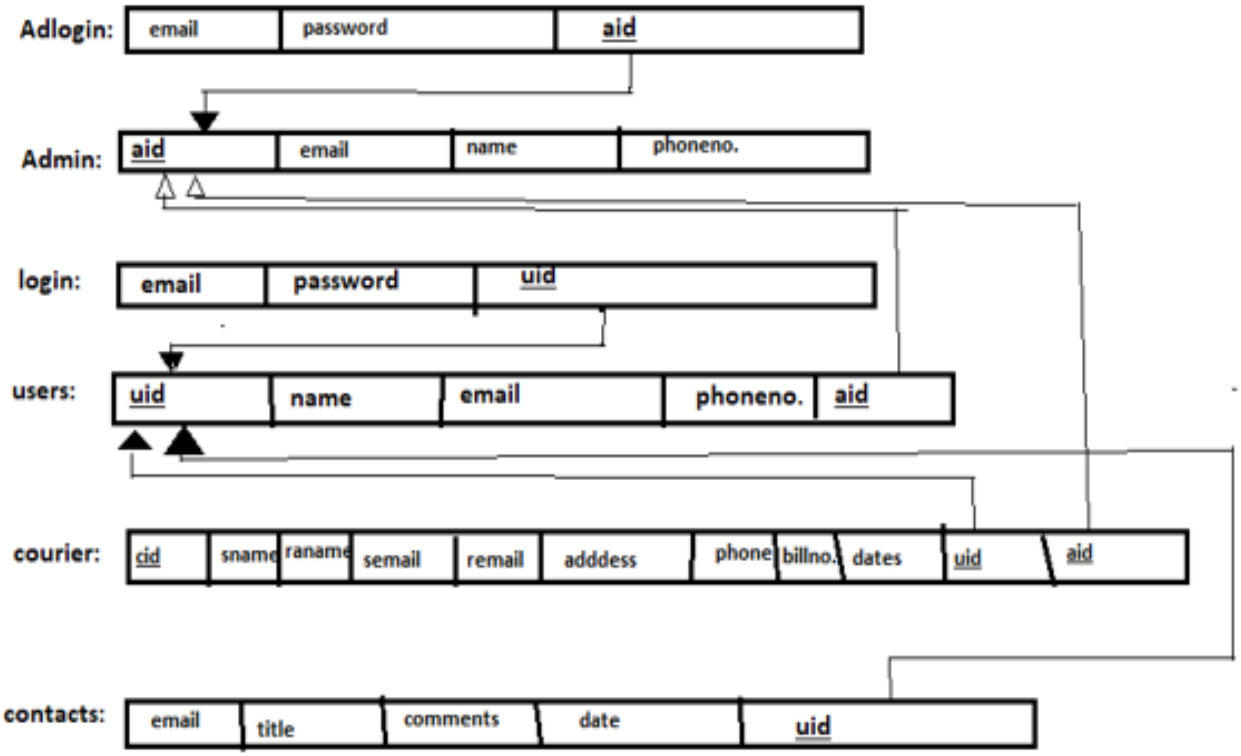
The courier service is one of the solutions of these problems. It is used to send some things to any person in the world within time. The courier company has number of branches, which are spread over the country or the world. So that when person wants to send things then he has to contact at nearest courier service branch. The courier company creates the schedule & gives internal/external services. The courier service work as destination office or source office.

- **We used DBMS in Courier management system**
- **First, we need to identify the entities, attribute and relationship**
- **We show entities, attributes and relationship by ER diagram.**
- **SQL is a language which is used for applying queries. Example we apply queries for insertion, deletion or updating etc.**

ER Diagram


email
password
ulogin
1
has
1
uid
email
Users
N
take
cares
1
Admin
email
password
adlogin
1
has
1
aid
email
phone
name
uid
name
phone
1
can
N
1
have
N
1
manage
N
comments
email
title
date
comment
cid
sname
rname
semail
remail
sphone
courier
rphone
billno.
saddress
raddress
date

Database Schema



Command For Creation This Database And Show All: -

Create Database: -

The CREATE DATABASE statement is used to create a new SQL database.

Syntax: -

CREATE DATABASE *databasename*;

Example: -

CREATE DATABASE *courier*;

OUTPUT: -

```
MariaDB [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| courier            |
| information_schema |
| library            |
| mysql              |
| performance_schema |
| phpmyadmin         |
| test               |
+--------------------+
7 rows in set (0.001 sec)
```

Commands for Creating Tables:-

Creating a Table: -

The CREATE TABLE statement is used to create a new table in a database.The column parameters specify the names of the columns of the table. The datatype parameter specifies the type of data the column can hold (e.g. varchar, integer, date, etc.).

Syntax: -

CREATE TABLE *table_name* (
column1 datatype,
column2 datatype,
column3 datatype,
....
);

Example: -

```
CREATE TABLE  admin  (
  a_id  int(11) NOT NULL,
  email varchar(50) NOT NULL,
  name varchar(50) DEFAULT NULL,
  pnumber int(14) DEFAULT NULL
```

OUTPUT: -

```
MariaDB [courier]> DESC admin;
+---------+-------------+------+-----+---------+----------------+
| Field   | Type        | Null | Key | Default | Extra          |
+---------+-------------+------+-----+---------+----------------+
| a_id    | int(11)     | NO   | PRI | NULL    | auto_increment |
| email   | varchar(50) | NO   | UNI | NULL    |                |
| name    | varchar(50) | YES  |     | NULL    |                |
| pnumber | int(14)     | YES  |     | NULL    |                |
+---------+-------------+------+-----+---------+----------------+
4 rows in set (0.009 sec)
```

Commands For Selecting a Table: -

Selecting a table from a database: -

The following SQL statement selects all the columns from the "courier" table.

Command: -

SELECT * FROM *users*;

OUTPUT: -

```
MariaDB [courier]> SELECT * FROM admin;
+------+------------------+--------+---------+
| a_id | email            | name   | pnumber |
+------+------------------+--------+---------+
|    1 | admin1@gmail.com | Admin1 |   12345 |
|    2 | admin2@gmail.com | Admin2 |   12345 |
+------+------------------+--------+---------+
2 rows in set (0.001 sec)
```

Command to Insert into a table: -

Insert into a table: -

The INSERT INTO statement is used to insert new records in a table.

Syntax:-

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

Eaxmple: -

INSERT INTO *usres* (a_id, email, name,pnumber) VALUES

(1, 'admin1@gmail.com ', 'Admin1', '12345'),

(2, 'admin2@gmail.com ', 'Admin2', '12345'),

(3, 'devendrakumar34188@gmail.com ', 'Dev', '735155');

OUTPUT: -

```
MariaDB [courier]> INSERT INTO admin
    -> values
    -> (3,'devendrakumar34188@gmail.com','Dev',735155);
Query OK, 1 row affected (0.015 sec)

MariaDB [courier]> SELECT * FROM admin;
+------+-------------------------------+--------+---------+
| a_id | email                         | name   | pnumber |
+------+-------------------------------+--------+---------+
|    1 | admin1@gmail.com              | Admin1 |   12345 |
|    2 | admin2@gmail.com              | Admin2 |   12345 |
|    3 | devendrakumar34188@gmail.com | Dev    |  735155 |
+------+-------------------------------+--------+---------+
3 rows in set (0.001 sec)
```

Queries for our Database: -

Show tables: -

Here, we first se that how much tables we have in our database.

Command: -

SHOW *tables*;

OUTPUT: -

```
MariaDB [courier]> SHOW TABLES;
+------------------+
| Tables_in_courier |
+------------------+
| adlogin          |
| admin            |
| contacts         |
| courier          |
| login            |
| users            |
+------------------+
6 rows in set (0.001 sec)
```

Selecting a perticular column from a table: -

The following SQL statement selects all (including the duplicates) values from the "email" column in the "admin" table.

Command: -

SELECT *email* FROM *admin*;

OUTPUT: -

```
MariaDB [courier]> SELECT email FROM admin;
+-------------------------------+
| email                         |
+-------------------------------+
| admin1@gmail.com              |
| admin2@gmail.com              |
| devendrakumar34188@gmail.com  |
+-------------------------------+
3 rows in set (0.001 sec)
```

Selecting rows according to the condition : -

The WHERE clause is used to filter records. It is used to extract only those records that fulfill a specified condition.

Command: -

SELECT * FROM admin WHERE a_id = 1;

OUTPUT: -

```
MariaDB [courier]> SELECT * FROM admin WHERE a_id = 1;
+------+------------------+--------+---------+
| a_id | email            | name   | pnumber |
+------+------------------+--------+---------+
|    1 | admin1@gmail.com | Admin1 |   12345 |
+------+------------------+--------+---------+
1 row in set (0.008 sec)
```

Update a particular column in a table: -

The UPDATE statement is used to modify the existing records in a table. Be careful when updating records. If you omit the WHERE clause, ALL records will be updated. First we select the admin to see all the columns.

Command: -

SELECT * FROM *admin*;

OUTPUT: -

```
MariaDB [courier]> SELECT * FROM admin;
+------+-------------------------------+--------+---------+
| a_id | email                         | name   | pnumber |
+------+-------------------------------+--------+---------+
|    1 | admin1@gmail.com              | Admin1 |   12345 |
|    2 | admin2@gmail.com              | Admin2 |   12345 |
|    3 | devendrakumar34188@gmail.com | Dev    |  735155 |
+------+-------------------------------+--------+---------+
3 rows in set (0.001 sec)
```

Then we use UPDATE command to update the first row with a_id = 1.

Command: -

UPDATE *admin,*

SET pnumber = `863089`, name = ‘Devendra,

WHERE a_id = 1;

OUTPUT: -

```
MariaDB [courier]> UPDATE admin
    -> SET pnumber = '863089',
    -> name ='Devendra'
    -> WHERE a_id =1;
Query OK, 1 row affected (0.005 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

After the UPDATE Command the table will look like: -

```
MariaDB [courier]> SELECT * FROM admin;
+------+-------------------------------+----------+---------+
| a_id | email                         | name     | pnumber |
+------+-------------------------------+----------+---------+
|    1 | admin1@gmail.com              | Devendra |  863089 |
|    2 | admin2@gmail.com              | Admin2   |   12345 |
|    3 | devendrakumar34188@gmail.com | Dev      |  735155 |
+------+-------------------------------+----------+---------+
3 rows in set (0.000 sec)
```

Alter a perticular column in a table: -

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.

The ALTER TABLE statement is also used to add and drop various constraints on an existing table.

First, we SELECT the table which we want to ALTER from the database: -

Command: -

SELECT * FROM *users*;

OUTPUT: -

```
MariaDB [courier]> SELECT * FROM users;
+------+-------------------------------+---------------+------------+
| u_id | email                         | name          | pnumber    |
+------+-------------------------------+---------------+------------+
|    1 | premkumar1215225@gmail.com | DevendraKumar |      56665 |
|    4 | love@gmail.com                | LOVE RAJ      | 2147483647 |
+------+-------------------------------+---------------+------------+
2 rows in set (0.003 sec)
```

Now we apply the ALTER command in the table: -

Command: -

ALTER TABLE *users*

ADD Address (15);

OUTPUT: -

```
MariaDB [courier]> ALTER TABLE users
    -> ADD Address varchar(15);
Query OK, 0 rows affected (0.029 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

After Adding a column, the flight table will look like: -

```
MariaDB [courier]> SELECT * FROM users;
+------+-----------------------------+---------------+------------+---------+
| u_id | email                       | name          | pnumber    | Address |
+------+-----------------------------+---------------+------------+---------+
|    1 | premkumar1215225@gmail.com | DevendraKumar |      56665 | NULL    |
|    4 | love@gmail.com              | LOVE RAJ      | 2147483647 | NULL    |
+------+-----------------------------+---------------+------------+---------+
2 rows in set (0.001 sec)
```

Deleting a column from a table: -

To delete a column in a table, use the following syntax (notice that some database systems don't allow deleting a column):

Syntax: -

ALTER TABLE *table_name*
DROP COLUMN *column_name*;

First, we will SELECT the flight table to see the rows and columns.

Command: -

SELECT * FROM *users*;

OUTPUT: -

```
MariaDB [courier]> SELECT * FROM users;
+------+----------------------------+---------------+------------+---------+
| u_id | email                      | name          | pnumber    | Address |
+------+----------------------------+---------------+------------+---------+
|    1 | premkumar1215225@gmail.com | DevendraKumar |      56665 | NULL    |
|    4 | love@gmail.com             | LOVE RAJ      | 2147483647 | NULL    |
+------+----------------------------+---------------+------------+---------+
2 rows in set (0.001 sec)
```

Now we use ALTER command and the DROP to delete the Address column.

Command: -

ALTER TABLE *users*

DROP COLUMN Address;

OUTPUT: -

```
MariaDB [courier]> ALTER TABLE users
    -> DROP COLUMN Address;
Query OK, 0 rows affected (0.018 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

After deleting the TERMINAL column from flight table: -

```
MariaDB [courier]> SELECT * FROM users;
+------+---------------------------+---------------+------------+
| u_id | email                     | name          | pnumber    |
+------+---------------------------+---------------+------------+
|    1 | premkumar1215225@gmail.com | DevendraKumar |      56665 |
|    4 | love@gmail.com            | LOVE RAJ      | 2147483647 |
+------+---------------------------+---------------+------------+
2 rows in set (0.001 sec)
```

Now let's see all the tables in the airlines Database: -

```
MariaDB [courier]> use courier;
Database changed
MariaDB [courier]> SHOW TABLES;
+--------------------+
| Tables_in_courier  |
+--------------------+
| adlogin            |
| admin              |
| contacts           |
| courier            |
| login              |
| users              |
+--------------------+
6 rows in set (0.002 sec)
```

Project Operation: -

- This operation shows the list of those attributes that we wish to appear in the result. Rest of the attributes are eliminated from the table.
- It is denoted by ∏.

Notation: ∏ A1, A2, An (r)

Where:

A1, **A2**, **A3** is used as an attribute name of relation **r**.

INPUT: -

∏ email,name,pnumber(users)

OUTPUT: -

```
MariaDB [courier]> SELECT email,name,pnumber FROM users;
+---------------------------+---------------+------------+
| email                     | name          | pnumber    |
+---------------------------+---------------+------------+
| premkumar1215225@gmail.com | DevendraKumar |      56665 |
| love@gmail.com            | LOVE RAJ      | 2147483647 |
+---------------------------+---------------+------------+
2 rows in set (0.001 sec)
```