INTERNATIONAL ISLAMIC UNIVERSITY CHITTAGONG

DEPARTMRNT OF COMPUTER SCIENCE & ENGINEERING

Project on : Courier Management System

Course code : CSE-2424

Course title : Database Management Systems Lab

Submitted by-

Name : Nafisa Nawar

ID : C221222

Semester : 4th

Section : 4AF

Date of Allotment: 02/11/2023

Date of submission: 17/11/2023

Submitted to-

Dr.Mohammad Aman Ullah

Associate Professor

Department of CSE, IIUC

COURIER MANAGEMENT SYSTEM

- **What is courier?**
- A courier is a person or organization that delivers a message, package or letter from one place or person to another place or person.
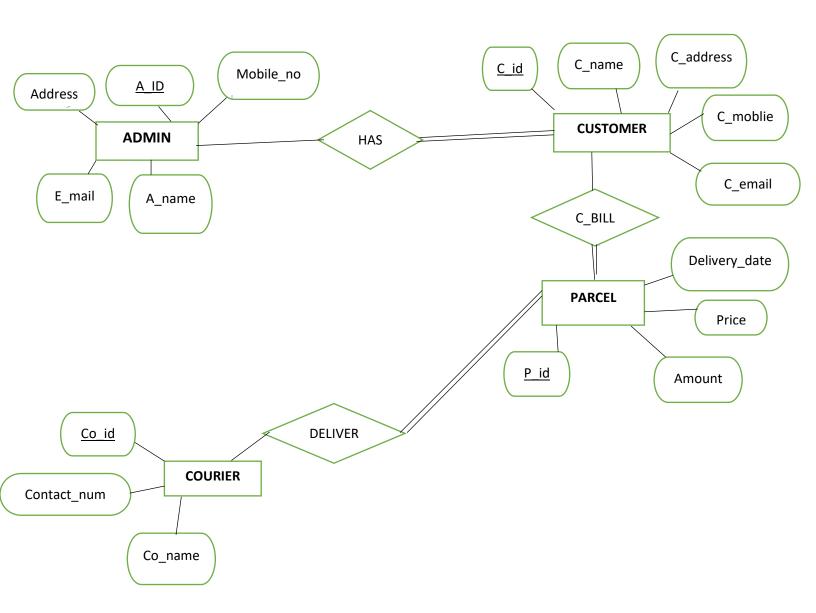-

- **What does mean courier management system?**
- A courier delivery management system, or CMS, is business software that simplifies courier management and routing.
- Courier management refers to the organization and coordination of last-mile -delivery operations. It involves scheduling deliveries, assigning couriers, managing operations, optimizing routes, and controlling fuel and expense costs. A courier management system, also known as courier software, manages the transportation of shipments from the warehouse to the customer. It automates the many operations involved in delivery businesses, such as real-time tracking.
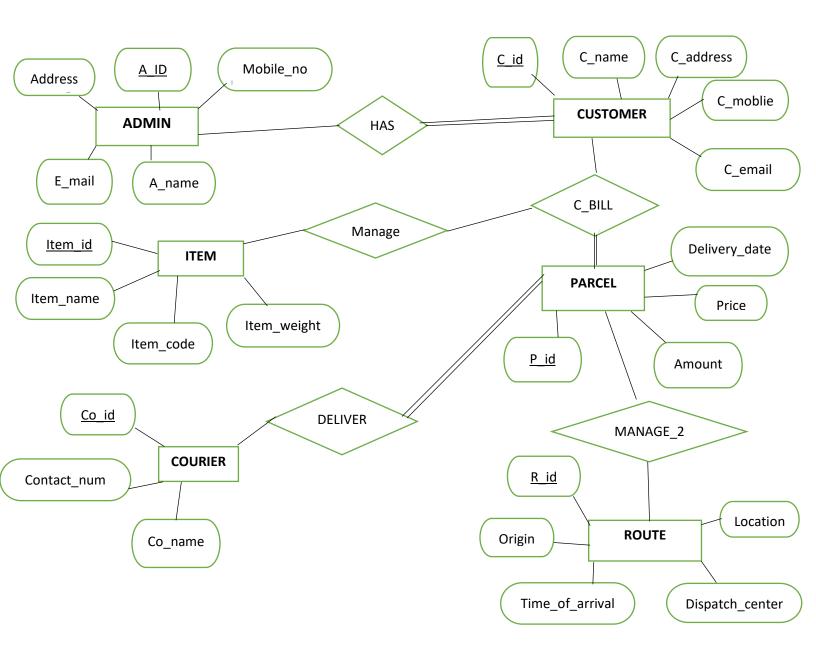
- **Advantage of courier management system:**
- They help transport businesses manage the dispatch of packages. With the rise of next-day and same-day delivery demands, a reliable courier management system is a must-have
- There are three main objectives of courier management: Deliver products to customers. Support the couriers delivering those products. Ensure that customers have a great buying experience, so they come back for more.

CONCEPTUAL ER-DIAGRAM

ADMIN
- Address
- A_ID
- Mobile_no
- E_mail
- A_name

CUSTOMER
- C_id
- C_name
- C_address
- C_moblie
- C_email

PARCEL
- Delivery_date
- Price
- P_id
- Amount

COURIER
- Co_id
- Contact_num
- Co_name

HAS

C_BILL

DELIVER

NORMALIZED ER-DIAGRAM

DATABASE SCHEMA

ADMIN (A_ID, A_NAME, MOBILE_NO, EMAIL,ADDRESS)

COURIER (CO_ID, CONTACT_NUM, CO_NAME)

CUSTOMER
(C_ID,C_NAME,C_EMAIL,C_MOBILE_NO,C_ADDRESS)

ITEM (ITEM_ID,ITEM_NAME,ITEM_CODE,ITEM_WEIGHT)

PARCEL (P_ID,AMOUNT,PRICE,DELIVERY_DATE)

ROUTE
(R_ID,ORIGIN,TIME_OF_ARRIVAL,LOCATION,DISPATCH_CENTER
)

HAS(A_ID,C_ID)

C_BILL(C_ID,P_ID)

MANAGE(C_ID,P_ID,ITEM_ID)

MANAGE_2(P_ID,R_ID)

DELIVER(P_ID,CO_ID)

DDL STATEMENT

➢ ADMIN TABLE

ADMIN

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Add Column | Modify Column | Rename Column | Drop Column | Rename | Copy | Drop | Truncate | Create Lookup

Column Name	Data Type	Nullable	Default	Primary Key
A_ID	VARCHAR2(200)	No	-	1
A_NAME	VARCHAR2(100)	No	-	-
MOBILE_NO	VARCHAR2(15)	No	-	-
EMAIL	VARCHAR2(100)	Yes	-	-
ADDRESS	VARCHAR2(400)	Yes	-	-
				1 - 5

SQL

CREATE TABLE "ADMIN"

   ("A_ID" VARCHAR2(200) NOT NULL ENABLE,

      "A_NAME" VARCHAR2(100) NOT NULL ENABLE,

      "MOBILE_NO" VARCHAR2(15) NOT NULL ENABLE,

      "EMAIL" VARCHAR2(100),

      "ADDRESS" VARCHAR2(400),

       CONSTRAINT "ADMIN_PK" PRIMARY KEY ("A_ID") ENABLE

)

/

➢ COURIER TABLE

COURIER

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Add Column | Modify Column | Rename Column | Drop Column | Rename | Copy | Drop | Truncate | Create Lookup Table

Column Name	Data Type	Nullable	Default	Primary Key
CO_ID	VARCHAR2(400)	No	-	-
CONTACT_NUM	VARCHAR2(400)	No	-	-
CO_NAME	VARCHAR2(400)	No	-	-

1 - 3

SQL

CREATE TABLE "COURIER"

    ("CO_ID" VARCHAR2(400) NOT NULL ENABLE,

         "CONTACT_NUM" VARCHAR2(400) NOT NULL ENABLE,

         "CO_NAME" VARCHAR2(400) NOT NULL ENABLE

)

/

➢ CUSTOMER TABLE

CUSTOMER

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Add Column | Modify Column | Rename Column | Drop Column | Rename | Copy | Drop | Truncate | Create Lookup Table

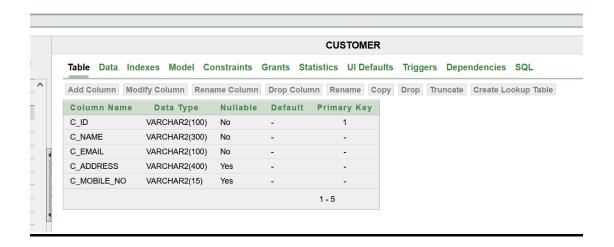| Column Name	Data Type	Nullable	Default	Primary Key
C_ID	VARCHAR2(100)	No	-	1
C_NAME	VARCHAR2(300)	No	-	-
C_EMAIL	VARCHAR2(100)	No	-	-
C_ADDRESS	VARCHAR2(400)	Yes	-	-
C_MOBILE_NO	VARCHAR2(15)	Yes	-	-

1 - 5

SQL

CREATE TABLE "CUSTOMER"

    ("C_ID" VARCHAR2(100) NOT NULL ENABLE,

          "C_NAME" VARCHAR2(300) NOT NULL ENABLE,

          "C_EMAIL" VARCHAR2(100) NOT NULL ENABLE,

          "C_ADDRESS" VARCHAR2(400),

          "C_MOBILE_NO" VARCHAR2(15),

           CONSTRAINT "CUSTOMER_PK" PRIMARY KEY ("C_ID") ENABLE

)

/

➢ ITEM TABLE

ITEM

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Add Column | Modify Column | Rename Column | Drop Column | Rename | Copy | Drop | Truncate | Create Lookup Table

Column Name	Data Type	Nullable	Default	Primary Key
ITEM_ID	VARCHAR2(400)	No	-	1
ITEM_NAME	VARCHAR2(400)	No	-	-
ITEM_CODE	NUMBER(5,0)	No	-	-
ITEM_WEIGHT	VARCHAR2(40)	No	-	-
			1 - 4	

SQL

CREATE TABLE "ITEM"

    ("ITEM_ID" VARCHAR2(400) NOT NULL ENABLE,

          "ITEM_NAME" VARCHAR2(400) NOT NULL ENABLE,

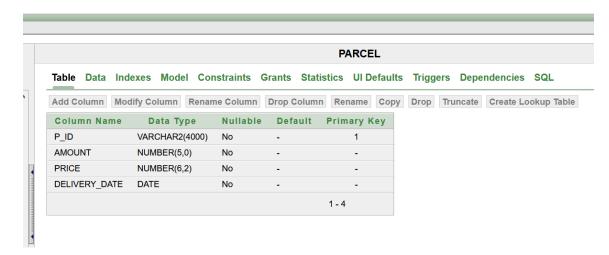   "ITEM_CODE" NUMBER(5,0) NOT NULL ENABLE,

          "ITEM_WEIGHT" VARCHAR2(40) NOT NULL ENABLE,

           CONSTRAINT "ITEM_PK" PRIMARY KEY ("ITEM_ID") ENABLE,

           CONSTRAINT "ITEM_CK1" CHECK (Item_Weight<'50kg') ENABLE

)

/

➢ PARCEL TABLE

PARCEL				
Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL				
Add Column Modify Column Rename Column Drop Column Rename Copy Drop Truncate Create Lookup Table				

Column Name	Data Type	Nullable	Default	Primary Key
P_ID	VARCHAR2(4000)	No	-	1
AMOUNT	NUMBER(5,0)	No	-	-
PRICE	NUMBER(6,2)	No	-	-
DELIVERY_DATE	DATE	No	-	-
			1 - 4	

SQL

CREATE TABLE "PARCEL"

   ("P_ID" VARCHAR2(4000) NOT NULL ENABLE,
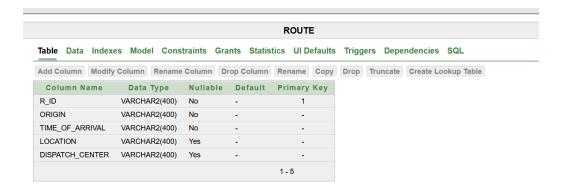
       "AMOUNT" NUMBER(5,0) NOT NULL ENABLE,

       "PRICE" NUMBER(6,2) NOT NULL ENABLE,

       "DELIVERY_DATE" DATE NOT NULL ENABLE,

       CONSTRAINT "PARCEL_PK" PRIMARY KEY ("P_ID") ENABLE

)

/

➢ ROUTE TABLE

ROUTE				
Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL				
Add Column Modify Column Rename Column Drop Column Rename Copy Drop Truncate Create Lookup Table				

Column Name	Data Type	Nullable	Default	Primary Key
R_ID	VARCHAR2(400)	No	-	1
ORIGIN	VARCHAR2(400)	No	-	-
TIME_OF_ARRIVAL	VARCHAR2(400)	No	-	-
LOCATION	VARCHAR2(400)	Yes	-	-
DISPATCH_CENTER	VARCHAR2(400)	Yes	-	-
			1 - 5	

SQL

```
CREATE TABLE "ROUTE"
  (     "R_ID" VARCHAR2(400) NOT NULL ENABLE,
        "ORIGIN" VARCHAR2(400) NOT NULL ENABLE,
        "TIME_OF_ARRIVAL" VARCHAR2(400) NOT NULL ENABLE,
        "LOCATION" VARCHAR2(400),
        "DISPATCH_CENTER" VARCHAR2(400),
         CONSTRAINT "ROUTE_PK" PRIMARY KEY ("R_ID") ENABLE
   )
/
```

➢ C_BILL

SQL

```
CREATE TABLE "C_BILL"
  (     "C_ID" VARCHAR2(400) NOT NULL ENABLE,
        "P_ID" VARCHAR2(400) NOT NULL ENABLE,
         CONSTRAINT "C_BILL_FK" FOREIGN KEY ("C_ID")
          REFERENCES  "CUSTOMER" ("C_ID") ENABLE,
         CONSTRAINT "C_BILL_FK2" FOREIGN KEY ("P_ID")
          REFERENCES  "PARCEL" ("P_ID") ENABLE
   )
/
```

➢ HAS

SQL

```
CREATE TABLE "HAS"
  (     "A_ID" VARCHAR2(400) NOT NULL ENABLE,
```

```
        "C_ID" VARCHAR2(400) NOT NULL ENABLE,
        CONSTRAINT "HAS_FK" FOREIGN KEY ("A_ID")
         REFERENCES  "ADMIN" ("A_ID") ENABLE,
        CONSTRAINT "HAS_FK2" FOREIGN KEY ("C_ID")
          REFERENCES  "CUSTOMER" ("C_ID") ENABLE
   )
/
```

➢ MANAGE

Column Name	Data Type	Nullable	Default	Primary Key
C_ID	VARCHAR2(400)	No	-	-
P_ID	VARCHAR2(400)	No	-	-
ITEM_ID	VARCHAR2(400)	No	-	-

1 - 3

SQL

```
CREATE TABLE  "MANAGE"
   (       "C_ID" VARCHAR2(400) NOT NULL ENABLE,
           "P_ID" VARCHAR2(400) NOT NULL ENABLE,
           "ITEM_ID" VARCHAR2(400) NOT NULL ENABLE,
         CONSTRAINT "MANAGE_FK" FOREIGN KEY ("C_ID")
          REFERENCES  "CUSTOMER" ("C_ID") ENABLE,
        CONSTRAINT "MANAGE_FK2" FOREIGN KEY ("P_ID")
         REFERENCES  "PARCEL" ("P_ID") ENABLE,
        CONSTRAINT "MANAGE_FK3" FOREIGN KEY ("ITEM_ID")
         REFERENCES  "ITEM" ("ITEM_ID") ENABLE
   )
/
```

➢ DELIVER

Column Name	Data Type	Nullable	Default	Primary Key
P_ID	VARCHAR2(4000)	Yes	-	-
CO_ID	VARCHAR2(400)	Yes	-	-

1 - 2

SQL

```
CREATE TABLE  "DELIVER"
  (       "P_ID" VARCHAR2(4000),
          "CO_ID" VARCHAR2(400)
  )
/
```

➢ MANAGE_2

SQL

```
CREATE TABLE  "MANAGE_2"
  (       "P_ID" VARCHAR2(4000),
          "R_ID" VARCHAR2(400)
  )
/
```

DML STATEMENT

➢ DATA IN ADMIN TABLE

ADMIN

Table **Data** Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query Count Rows Insert Row

EDIT	A_ID	A_NAME	MOBILE_NO	EMAIL	ADDRESS
	A08	Nur	01879306340	Nur@gmail.com	Halisahar
	A09	Ayesha	01877346798	Ayesha@gmail.com	Halisahar
	A10	Isha	01755498452	Isha@gmail.com	chaktai
	A02	Jarin	01881491513	Jarin@gmail.com	East madarbari
	A03	Nafisa	01741246146	Nafisa@gmail.com	West madarbari
	A04	Nawar	01815946807	Nawar@gmail.com	West madarbari
	A05	Israt	01875747724	Israt@gmail.com	Sadarghat
	A01	Esha	01815946806	Esha@gmail.com	East madarbari
	A06	Saima	01745882499	Saima@gmail.com	Jamalkhan
	A07	Joha	01761921554	Joha@gmail.com	Jamalkhan
					row(s) 1 - 10 of 10

Download

➢ DATA IN COURIER TABLE

COURIER

Table **Data** Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query Count Rows Insert Row

EDIT	CO_ID	CONTACT_NUM	CO_NAME
	CO9	01741246145	Ishra
	CO2	01881491513	Jarin
	CO1	01815946806	Nafisa
	CO3	01815946807	Nawar
	CO4	01945248748	Mehanaz
	CO5	01741220708	Humaira
	CO6	01822597911	Sadia
	CO7	01872313886	Elahi
	CO10	01883566866	Manisha
	CO8	01994672231	Israt
		row(s) 1 - 10 of 10	

Download

➢ DATA IN CUSTOMER TABLE

CUSTOMER   Create ▼

Table **Data** Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query | Count Rows | Insert Row

EDIT	C_ID	C_NAME	C_EMAIL	C_ADDRESS	C_MOBILE_NO
✎	C1	Raisa	Raisa@gmail.com	Chittagong	01888498702
✎	C2	Zarin	Zarin@gmail.com	khulshi	01765098231
✎	C4	Ema	Ema@gmail.com	Chittagong	01834366592
✎	C5	Ria	Ria@gmail.com	Chittagong	01774567349
✎	C6	Jarin	Jarin@gmail.com	Halisahar	01884562294
✎	C7	Jeba	Jeba@gmail.com	pathorghata	01785438893
✎	C8	Humaira	Humaira@gmail.com	Agrabad	01984563545
✎	C9	Adon	Adon@gmail.com	Chaktai	01774567371
✎	C10	Israt	Israt@gmail.com	Chittagong	01877564326
✎	C3	Ina	Ina@gmail.com	jamalkhan	01734558971

row(s) 1 - 10 of 10

Download

➢ DATA IN ITEM TABLE

ITEM   Create ▼

Table **Data** Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query | Count Rows | Insert Row

EDIT	ITEM_ID	ITEM_NAME	ITEM_CODE	ITEM_WEIGHT
✎	I101	Bag	110	300g
✎	I103	Shirt	23	140g
✎	I104	Jersy	400	100g
✎	I102	Luggage	10	4000g
✎	I105	Photo frame	20	2500g
✎	I107	Keyboard	7	1500g
✎	I108	Mouse	8	100g
✎	I109	phone cover	2	20g
✎	I110	Earphone	5	13.11g
✎	I106	BOOK	6	1000g

row(s) 1 - 10 of 10

Download

➢ DATA IN PARCEL TABLE

PARCEL

Table **Data** Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query | Count Rows | Insert Row

EDIT	P_ID	AMOUNT	PRICE	DELIVERY_DATE
	P06	4	2000	04-JUN-23
	P08	4	3500	24-SEP-23
	P09	3	800	27-AUG-23
	P02	2	600	11-OCT-23
	P03	1	200	10-FEB-23
	P04	1	700	13-FEB-23
	P01	1	300	12-NOV-22
	P07	3	3000	17-JAN-23
	P05	2	500	02-FEB-23
	P10	3	1200	16-OCT-23
			row(s) 1 - 10 of 10	

Download

➢ DATA IN ROUTE TABLE

ROUTE

Table **Data** Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query | Count Rows | Insert Row

EDIT	R_ID	ORIGIN	TIME_OF_ARRIVAL	LOCATION	DISPATCH_CENTER
	R07	Chittagong	1.00pm	Sylhet	Sylhet
	R09	Chittagong	8.00pm	Chittagong	Chittagong
	R01	Chittagong	3.00pm	Chittagong	Chittagong
	R02	Dhaka	4.00pm	Chittagonng	Chittagong
	R03	Dhaka	2.00pm	Dhaka	Dhaka
	R04	Chittagong	6.00pm	Dhaka	Dhaka
	R05	Chittagong	4.00pm	Sylhet	Sylhet
	R06	Sylhet	11.00am	Chittagong	Chittagong
	R08	sylhet	12.00pm	Dhaka	Dhaka
	R10	Dhaka	10.00am	Dhaka	Dhaka
				row(s) 1 - 10 of 10	

Download

➢ DATA IN C_BILL TABLE

C_BILL

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query Count Rows Insert Row

EDIT	C_ID	P_ID
📝	C1	P01
📝	C4	P04
📝	C5	P05
📝	C9	P09
📝	C2	P02
📝	C3	P03
📝	C6	P06
📝	C7	P07
📝	C8	P08
📝	C10	P10

row(s) 1 - 10 of 10

Download

➢ DATA IN HAS TABLE

HAS   Create ▼

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query Count Rows Insert Row

EDIT	A_ID	C_ID
📝	A06	C6
📝	A08	C8
📝	A01	C1
📝	A02	C2
📝	A03	C3
📝	A04	C4
📝	A05	C5
📝	A07	C7
📝	A09	C9
📝	A10	C10

row(s) 1 - 10 of 10

Download

➢ DATA IN DELIVER TABLE

DELIVER   Create

Table Data Indexes Model Constraints Grants Statistics UI Defaults Triggers Dependencies SQL

Query Count Rows Insert Row

EDIT	P_ID	CO_ID
📝	P01	CO1
📝	P02	CO2
📝	P08	CO8
📝	P03	CO3
📝	P04	CO4
📝	P05	CO5
📝	P06	CO6
📝	P07	CO7
📝	P09	CO9
📝	P10	CO10

row(s) 1 - 10 of 10

Download

➢ DATA IN MANAGE TABLE

➢ DATA IN MANAGE_2 TABLE

SEARCHING DATA FROM INDIVIDUAL TABLE:

QUERY 1 :

Find all information of Admin.
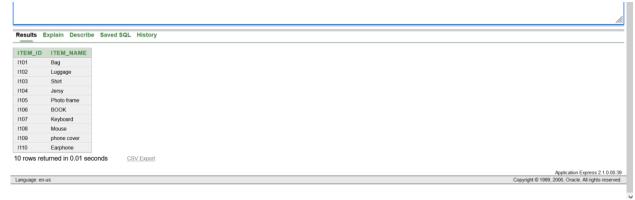
SQL:

SELECT *

FROM ADMIN

A_ID	A_NAME	MOBILE_NO	EMAIL	ADDRESS
A08	Nur	01879306340	Nur@gmail.com	Halisahar
A09	Ayesha	01877346798	Ayesha@gmail.com	Halisahar
A10	Isha	01755498452	Isha@gmail.com	chaktai
A02	Jarin	01881491513	Jarin@gmail.com	East madarbari
A03	Nafisa	01741246146	Nafisa@gmail.com	West madarbari
A04	Nawar	01815946807	Nawar@gmail.com	West madarbari
A05	Israt	01875747724	Israt@gmail.com	Sadarghat
A01	Esha	01815946806	Esha@gmail.com	East madarbari
A06	Saima	01745882499	Saima@gmail.com	Jamalkhan
A07	Joha	01761921554	Joha@gmail.com	Jamalkhan

10 rows returned in 0.01 seconds   CSV Export

QUERY 2:

Show ITEM_ID increasing order from item.

SQL:

SELECT ITEM_ID,ITEM_NAME

FROM ITEM

ORDER BY ITEM_ID;

ITEM_ID	ITEM_NAME
I101	Bag
I102	Luggage
I103	Shirt
I104	Jersy
I105	Photo frame
I106	BOOK
I107	Keyboard
I108	Mouse
I109	phone cover
I110	Earphone

10 rows returned in 0.01 seconds   CSV Export

QUERY 3:

Find those customer who live in Bangladesh.

SQL:

SELECT*

FROM CUSTOMER

WHERE C_ADDRESS='Chittagong';

C_ID	C_NAME	C_EMAIL	C_ADDRESS	C_MOBILE_NO
C1	Raisa	Raisa@gmail.com	Chittagong	01888498702
C5	Ria	Ria@gmail.com	Chittagong	01774567349
C10	Israt	Israt@gmail.com	Chittagong	01877564326

3 rows returned in 0.01 seconds   CSV Export

QUERY 4:
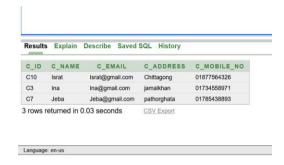
Find that customer whose id is C3, C7, C10.

SQL:

SELECT *

FROM CUSTOMER

WHERE C_ID IN ('C7', 'C3', 'C10') ;

C_ID	C_NAME	C_EMAIL	C_ADDRESS	C_MOBILE_NO
C10	Israt	Israt@gmail.com	Chittagong	01877564326
C3	Ina	Ina@gmail.com	jamalkhan	01734558971
C7	Jeba	Jeba@gmail.com	pathorghata	01785438893

3 rows returned in 0.03 seconds   CSV Export

Language: en-us

QUERY 5:
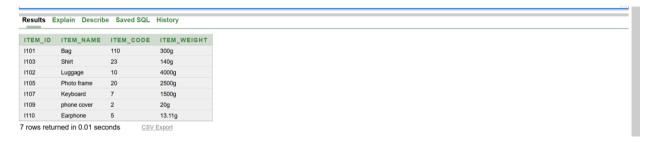
 Find that item which weight between 10g to 4000kg.

SQL:

SELECT *

FROM ITEM

WHERE ITEM_WEIGHT BETWEEN '10g' and '4000g';

QUERY 6:
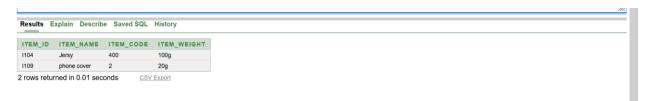
Find that name of items which strings 'er'.

SQL:

SELECT *

FROM ITEM

WHERE ITEM_NAME LIKE '%er%';

QUERY 7:
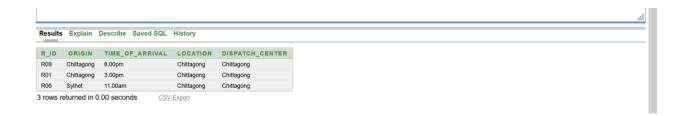
Find those parcel which location and dispatch center is in same district Chittagong.

SQL:

SELECT*

FROM ROUTE

WHERE LOCATION='Chittagong' and DISPATCH_CENTER='Chittagong';

QUERY 8:

Find Total admin from the table Admin.
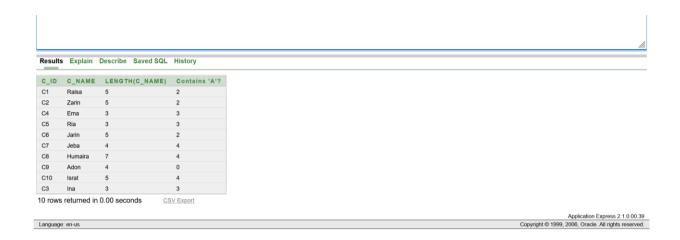
SQL:

SELECT COUNT (DISTINCT A_ID)

FROM ADMIN;

Results Explain Describe Saved SQL History

COUNT(DISTINCTA_ID)
10

1 rows returned in 0.00 seconds   CSV Export

QUERY 9:

Find the length of customer name and contains "a" from customer table.

SQL:

SELECT   C_ID, C_NAME,

LENGTH (C_NAME), INSTR (C_NAME, 'a') "Contains 'a'?"

FROM CUSTOMER

QUERY 10:
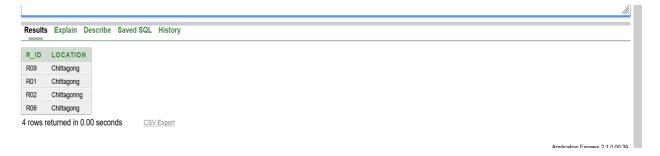
Find those parcels location which not belongs to Dhaka and Sylhet from route table.

SQL:

SELECT R_ID, LOCATION

FROM ROUTE

WHERE LOCATION NOT IN ('Dhaka', 'Sylhet');

Results Explain Describe Saved SQL History

R_ID	LOCATION
R09	Chittagong
R01	Chittagong
R02	Chittagonng
R06	Chittagong

4 rows returned in 0.00 seconds   CSV Export

Application Express 2.1.0.00.39

SEARCHING DATA FROM MULTIPLE TABLE

QUERY 1:

Show the all data of table Customer and Parcel using full outer join.

SQL:

SELECT*

From CUSTOMER NATURAL JOIN C_BILL NATURAL JOIN PARCEL

WHERE   PRICE <700;

Results Explain Describe Saved SQL History

P_ID	C_ID	C_NAME	C_EMAIL	C_ADDRESS	C_MOBILE_NO	AMOUNT	PRICE	DELIVERY_DATE
P01	C1	Raisa	Raisa@gmail.com	Chittagong	01886498702	1	300	12-NOV-22
P05	C5	Ria	Ria@gmail.com	Chittagong	01774567349	2	500	02-FEB-23
P02	C2	Zarin	Zarin@gmail.com	khulshi	01765098231	2	600	11-OCT-23
P03	C3	Ina	Ina@gmail.com	jamalkhan	01734558971	1	200	10-FEB-23

4 rows returned in 0.00 seconds   CSV Export

QUERY 2:

Show the all data of table C_BILL and Manage using full outer join.

SQL:

SELECT B.C_ID, B.P_ID, M.C_ID, M.P_ID, M.ITEM_ID

FROM C_BILL B FULL OUTER join MANAGE M

ON (B.C_ID=M.P_ID);

C_ID	P_ID	C_ID	P_ID	ITEM_ID
C6	P06	-	-	-
C3	P03	-	-	-
C5	P05	-	-	-
C4	P04	-	-	-
C1	P01	-	-	-
C10	P10	-	-	-
C7	P07	-	-	-
C9	P09	-	-	-
C2	P02	-	-	-
C8	P08	-	-	-
-	-	C9	P09	I109
-	-	C2	P02	I102
-	-	C7	P07	I107
-	-	C1	P01	I101
-	-	C4	P04	I104
-	-	C6	P06	I106
-	-	C10	P10	I110
-	-	C8	P08	I108
-	-	C3	P03	I103
-	-	C5	P05	I105

20 rows returned in 0.00 seconds   CSV Export

Application Express 2.1.0.00.39

QUERY 3:

Show the CUSTOMER who paid amount less than 3500 TK.
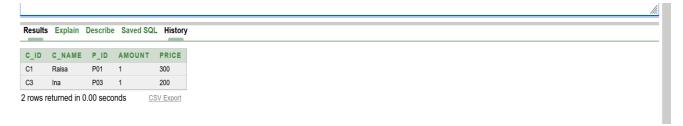
SQL:

SELECT C.C_ID, C.C_NAME, P.P_ID,P.AMOUNT, P.PRICE

FROM CUSTOMER C LEFT OUTER join PARCEL P

ON (C.C_ID=P.P_ID)

WHERE PRICE < 350;

Results Explain Describe Saved SQL History

C_ID	C_NAME	P_ID	AMOUNT	PRICE
C1	Raisa	P01	1	300
C3	Ina	P03	1	200

2 rows returned in 0.00 seconds   CSV Export

QUERY 4:

 Show customer id, parcel id & item id.

SQL:

SELECT *

FROM MANAGE

Results Explain Describe Saved SQL History

C_ID	P_ID	ITEM_ID
C1	P01	I101
C4	P04	I104
C5	P05	I105
C6	P06	I106
C7	P07	I107
C9	P09	I109
C2	P02	I102
C3	P03	I103
C8	P08	I108
C10	P10	I110

10 rows returned in 0.02 seconds   CSV Export

QUERY 5 :

Show the all data of table courier and Parcel using full outer join where price is 800.

SQL:

SELECT*

From COURIER NATURAL JOIN C_BILL NATURAL JOIN PARCEL

WHERE　　　PRICE=800;

SUB QUERY

QUERY 1 :

Give information of all admin whose admin name is same as Jarin.

SQL:
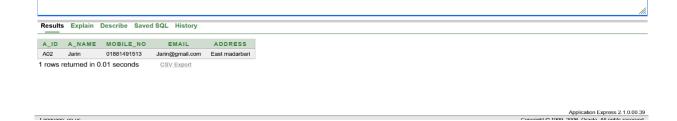
SELECT *

FROM ADMIN

WHERE A_NAME=(SELECT A_NAME

FROM ADMIN

WHERE A_NAME='Jarin')

A_ID	A_NAME	MOBILE_NO	EMAIL	ADDRESS
A02	Jarin	01881491513	Jarin@gmail.com	East madarbari

1 rows returned in 0.01 seconds   CSV Export

QUERY 2:

Show the customer details whose customer address is not 'Dhaka'.

SQL:

SELECT *

FROM CUSTOMER

WHERE C_ADDRESS NOT IN (SELECT C_ADDRESS

FROM CUSTOMER

WHERE C_ADDRESS='Dhaka')

C_ID	C_NAME	C_EMAIL	C_ADDRESS	C_MOBILE_NO
C1	Raisa	Raisa@gmail.com	Chittagong	01888498702
C2	Zarin	Zarin@gmail.com	khulshi	01765098231
C4	Ema	Ema@gmail.com	Chittagong	01834366592
C5	Ria	Ria@gmail.com	Chittagong	01774567349
C6	Jarin	Jarin@gmail.com	Halisahar	01884562294
C7	Jeba	Jeba@gmail.com	pathorghata	01785438893
C8	Humaira	Humaira@gmail.com	Agrabad	01984563545
C9	Adon	Adon@gmail.com	Chaktai	01774567371
C10	Israt	Israt@gmail.com	Chittagong	01877564326
C3	Ina	Ina@gmail.com	jamalkhan	01734558971

10 rows returned in 0.01 seconds   CSV Export

QUERY 3:

Show the courier name of courier id 'CO1'.

SQL:
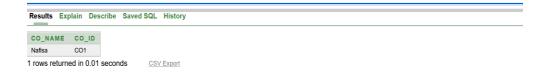
SELECT CO_NAME,CO_ID

FROM COURIER

WHERE CO_NAME =(SELECT CO_NAME

FROM COURIER

WHERE CO_ID='CO1');

CO_NAME	CO_ID
Nafisa	CO1

1 rows returned in 0.01 seconds   CSV Export

QUERY 4:

Select all the rows from the CUSTOMER table with the maximum id number.

SQL:

SELECT *

FROM CUSTOMER

WHERE C_ID=(SELECT MAX(C_ID)

FROM CUSTOMER);

QUERY 5:

Show the delivery date with their courier names.

SQL:

SELECT DELIVERY_DATE,CO_NAME

FROM PARCEL NATURAL JOIN (SELECT *

        FROM COURIER)

Results Explain Describe Saved SQL History

DELIVERY_DATE	CO_NAME
04-JUN-23	Ishra
04-JUN-23	Jarin
04-JUN-23	Nafisa
04-JUN-23	Nawar
04-JUN-23	Mehanaz
04-JUN-23	Humaira
04-JUN-23	Sadia
04-JUN-23	Elahi
04-JUN-23	Manisha
04-JUN-23	Israt

More than 10 rows available. Increase rows selector to view more rows.

10 rows returned in 0.00 seconds   CSV Export

QUERIES WITH PL/SQL:

QUERY 1:

FIND the total customer from the customer table using pl/sql.

SQL:

DECLARE n NUMBER;

BEGIN SELECT COUNT (*) INTO n FROM CUSTOMER;

dbms_output.put_line ('Total Customer:' ||n);

END;

Results Explain Describe Saved SQL History

Total Customer:10

Statement processed.

0.06 seconds

QUERY 2:

Show the details whose customer id is'C3'.

SQL:

DECLARE

A CUSTOMER.C_ID%TYPE;

B CUSTOMER.C_NAME%TYPE;

C CUSTOMER.C_ADDRESS%TYPE;

BEGIN

SELECT C_ID,C_NAME,C_ADDRESS INTO A,B,C

FROM CUSTOMER

WHERE C_ID='C3';

DBMS_OUTPUT.PUT_LINE('CUSTOMER_DETAILS:');

DBMS_OUTPUT.PUT_LINE('CUSTOMER ID:'||A);

DBMS_OUTPUT.PUT_LINE('CUSTOMER NAME:'||B);

DBMS_OUTPUT.PUT_LINE('CUSTOMER ADDRESS:'||C);

END;

```
Results  Explain  Describe  Saved SQL  History

CUSTOMER_DETAILS:
CUSTOMER ID:C3
CUSTOMER NAME:Ina
CUSTOMER ADDRESS:jamalkhan

Statement processed.
```

QUERY 3 :

Show the id info of customer named 'Jeba'.

SQL:

DECLARE

A CUSTOMER.C_ID%TYPE;

B CUSTOMER.C_ADDRESS%TYPE;

BEGIN

SELECT C_ID,C_ADDRESS INTO A,B

FROM CUSTOMER

WHERE C_NAME='Jeba';

DBMS_OUTPUT.PUT_LINE('ID_NFO:');

DBMS_OUTPUT.PUT_LINE('CUSTOMER ID:'||A);

DBMS_OUTPUT.PUT_LINE('CUSTOMER ADDRESS:'||B);

END;

```
Results  Explain  Describe  Saved SQL  History


ID_NFO:
CUSTOMER ID:C7
CUSTOMER ADDRESS:pathorghata

Statement processed.


0.02 seconds
```

QUERY 4 :

Show all customer names using cursor based record.

SQL:

DECLARE

CURSOR s_cur IS

SELECT *

FROM CUSTOMER;

s_rec s_cur%rowtype;

BEGIN

open s_cur;

LOOP

FETCH s_cur INTO s_rec;

dbms_output.put_line('CUSTOMER NAME & CUSTOMER ADDRESS:

'||s_rec.C_NAME||' '||s_rec.C_ADDRESS);

EXIT WHEN s_cur%notfound;

END LOOP;

END;

QUERY 5:

Show all admin nameusing cursor based record.

SQL:

DECLARE

A ADMIN%rowtype;

cursor C IS

SELECT *

FROM ADMIN;

BEGIN

OPEN C;

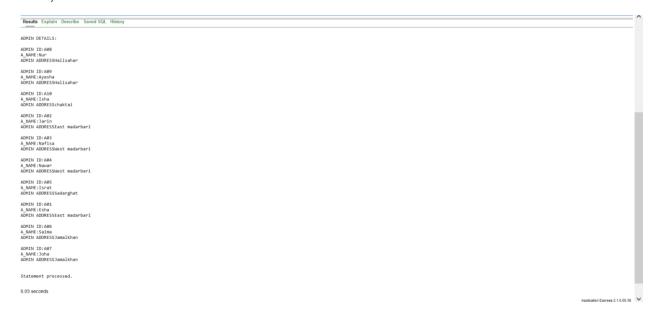dbms_output.put_line('ADMIN DETAILS:');

dbms_output.put_line('');

LOOP

FETCH C INTO A;

EXIT WHEN C%notfound;

dbms_output.put_line('ADMIN ID:'||A.A_ID);

dbms_output.put_line('A_NAME:'||A.A_NAME);

dbms_output.put_line('ADMIN ADDRESS'||A.ADDRESS);

dbms_output.put_line('');

END LOOP;

CLOSE C;

END;

```
Results  Explain  Describe  Saved SQL  History

ADMIN DETAILS:

ADMIN ID:A08
A_NAME:Nur
ADMIN ADDRESSHalisahar

ADMIN ID:A09
A_NAME:Ayesha
ADMIN ADDRESSHalisahar

ADMIN ID:A10
A_NAME:Isha
ADMIN ADDRESSchaktai

ADMIN ID:A02
A_NAME:Jarin
ADMIN ADDRESSEast madarbari

ADMIN ID:A03
A_NAME:Nafisa
ADMIN ADDRESSWest madarbari

ADMIN ID:A04
A_NAME:Nawar
ADMIN ADDRESSWest madarbari

ADMIN ID:A05
A_NAME:Israt
ADMIN ADDRESSSadarghat

ADMIN ID:A01
A_NAME:Esha
ADMIN ADDRESSEast madarbari

ADMIN ID:A06
A_NAME:Saima
ADMIN ADDRESSJamalkhan

ADMIN ID:A07
A_NAME:Joha
ADMIN ADDRESSJamalkhan


Statement processed.

0.03 seconds
                                                      Application Express 2.1.0.00.39
```

CONCLUSION:

➢ The Courier Service System project provides user interface in to the application through an authentified admin and also its a modern way of usage to a Courier Service through a computer system.

➢ The main proposed system of this web application is to remove the item details and check the current status of the user

➢ In conclusion, courier services provide several advantages, including fast and reliable delivery, secure transportation of sensitive items, and convenience.

THE END