

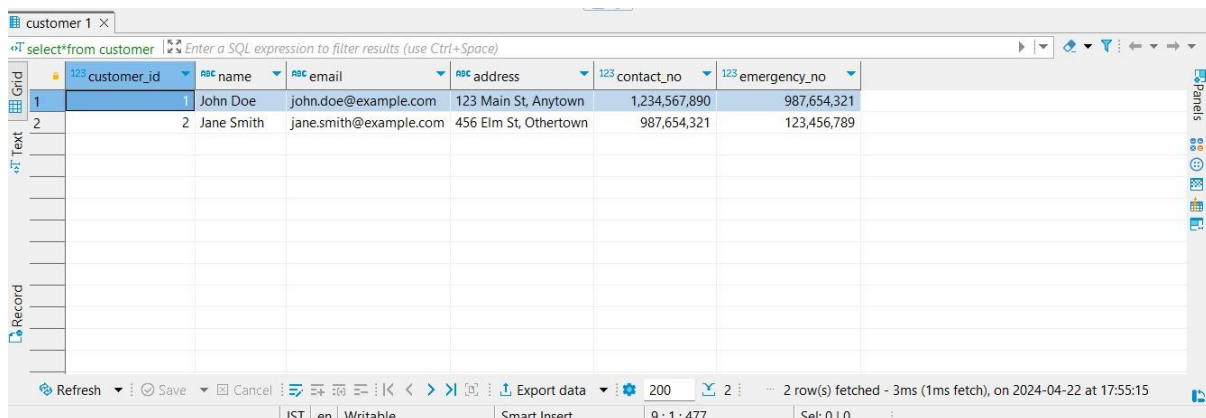
RED TAXI

- **DATABASE:**

```
create database redtaxi;  
USE redtaxi;
```

- **CUSTOMER TABLE:**

```
create table customer(customer_id int auto_increment primary key,  
name varchar(255),  
email varchar(255),  
address varchar(255),  
contact_no int,  
emergency_no int);  
INSERT INTO customer (name, email, address, contact_no, emergency_no)  
VALUES  
( 'John Doe', 'john.doe@example.com', '123 Main St, Anytown',  
1234567890, 987654321),  
( 'Jane Smith', 'jane.smith@example.com', '456 Elm St, Othertown',  
987654321, 123456789);  
select*from customer;
```



The screenshot shows a database management interface with a table named 'customer'. The table has 6 columns: customer_id, name, email, address, contact_no, and emergency_no. There are 2 rows of data. The interface includes a search bar at the top, a toolbar with various icons, and a status bar at the bottom indicating '2 row(s) fetched - 3ms (1ms fetch), on 2024-04-22 at 17:55:15'.

customer_id	name	email	address	contact_no	emergency_no
1	John Doe	john.doe@example.com	123 Main St, Anytown	1,234,567,890	987,654,321
2	Jane Smith	jane.smith@example.com	456 Elm St, Othertown	987,654,321	123,456,789

- **VECHICAL TABLE:**

```
create table vehicle (vehicle_id int auto_increment primary key,  
register_no int,  
model varchar(255),  
capacity int,  
is_automatic boolean comment 'True if the vehicle has automatic  
transmission, false if it has manual transmission');  
INSERT INTO vehicle (register_no, model, capacity, is_automatic)  
VALUES  
(12345, 'Toyota Camry', 5, TRUE),  
(67890, 'Honda Civic', 4, FALSE),  
(54321, 'Ford F-150', 3, TRUE),  
(98765, 'Chevrolet Silverado', 6, FALSE);  
select*from vehicle;
```

vehicle 1 x

select*from vehicle Enter a SQL expression to filter results (use Ctrl+Space)

	123 vehicle_id	123 register_no	ABC model	123 capacity	123 is_automatic
1	1	12,345	Toyota Camry	5	1
2	2	67,890	Honda Civic	4	0
3	3	54,321	Ford F-150	3	1
4	4	98,765	Chevrolet Silverado	6	0

Refresh Save Cancel Export data 200 4 4 row(s) fetched - 2ms (1ms fetch), on 2024-04-22 at 18:24:40

IST en Writable Smart Insert 16 : 1 : 964 Sel: 0 | 0

- DRIVER TABLE:**

```
create table driver (
  driver_id int auto_increment primary key,
  name varchar(255),
  gender enum('Male', 'Female', 'Other'),
  license_no int,
  license_exp_date date,
  is_status boolean comment 'True if the driver has current experience,
  false if the driver doesn't have experience for the past 3 months',
  vehicle_id int,
  foreign key (vehicle_id) references vehicle(vehicle_id)
);
insert into driver (name, gender, license_no, license_exp_date,
is_status, vehicle_id) values
('John Doe', 'Male', 123456, '2024-12-31', TRUE, 1),
('Jane Smith', 'Female', 654321, '2023-10-15', FALSE, 2),
('Alex Kim', 'Other', 987654, '2025-06-20', TRUE, 3);
select*from driver;
```

driver 1 x

select*from driver Enter a SQL expression to filter results (use Ctrl+Space)

	123 driver_id	ABC name	ABC gender	123 license_no	license_exp_date	123 is_status	123 vehicle_id
1	1	John Doe	Male	123,456	2024-12-31	1	1
2	2	Jane Smith	Female	654,321	2023-10-15	0	2
3	3	Alex Kim	Other	987,654	2025-06-20	1	3

Refresh Save Cancel Export data 200 3 3 row(s) fetched - 1ms, on 2024-04-22 at 18:55:51

IST en Writable Smart Insert 30 : 1 : 1663 Sel: 0 | 0

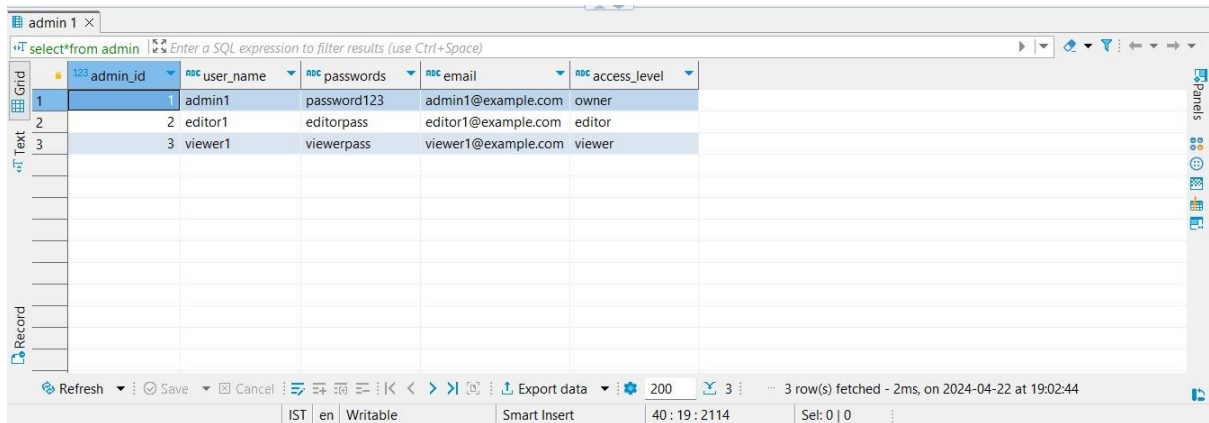
- ADMIN TABLE:**

```
create table admin(
  admin_id int auto_increment primary key,
  user_name varchar(255),
  passwords varchar(255),
  email varchar(255),
```

```

access_level enum('owner','editor','viewer'));
insert into admin (user_name, passwords, email, access_level) values
('admin1', 'password123', 'admin1@example.com', 'owner'),
('editor1', 'editorpass', 'editor1@example.com', 'editor'),
('viewer1', 'viewerpass', 'viewer1@example.com', 'viewer');
select*from admin;

```



admin_id	user_name	passwords	email	access_level
1	admin1	password123	admin1@example.com	owner
2	editor1	editorpass	editor1@example.com	editor
3	viewer1	viewerpass	viewer1@example.com	viewer

• TRIP TABLE:

```

create table trip (
trip_id int auto_increment primary key,
customer_id INT,
foreign key (customer_id) references customer(customer_id),
vehicle_id INT,
foreign key (vehicle_id) references vehicle(vehicle_id),
driver_id INT,
foreign key (driver_id) references driver(driver_id),
starting_location varchar(255),
ending_location varchar(255),
starting_time time,
ending_time time,
distance int,
trip_date date
);
insert into trip (customer_id, vehicle_id, driver_id, starting_location,
ending_location, starting_time, ending_time, distance, trip_date) values
(1, 1, 1, 'New York', 'Boston', '08:00:00', '12:00:00', 200, '2024-04-22'),
(1, 2, 2, 'Los Angeles', 'San Francisco', '10:00:00', '14:00:00', 400,
'2024-04-23'),
(1, 3, 3, 'Chicago', 'Detroit', '09:00:00', '13:00:00', 300, '2024-04-24');
select*from trip;

```

trip 1 x

select*from trip

	123 trip_id	123 customer_id	123 vehicle_id	123 driver_id	noc starting_location	noc ending_location	starting_time	ending_time	123 distance
1	13	1	1	1	New York	Boston	08:00:00	12:00:00	20
2	14	1	2	2	Los Angeles	San Francisco	10:00:00	14:00:00	40
3	15	1	3	3	Chicago	Detroit	09:00:00	13:00:00	30

Refresh Save Cancel Export data 200 3 3 row(s) fetched - 4ms (1ms fetch), on 2024-04-22 at 21:21:58

IST en Writable Smart Insert 69: 1 [17] Sel: 17 | 1

• PAYMENT TABLE:

```
create table payment(
payment_id int auto_increment primary key,
trip_id int,
foreign key (trip_id) references trip(trip_id),
amount int,
payment_method varchar(255),
transaction_id varchar(255),
total_amount bigint
);
insert into payment (trip_id, amount, payment_method, transaction_id,
total_amount) values
(13, 50, 'Credit Card', 'ABC123', 60),
(14, 60, 'PayPal', 'DEF456', 70),
(15, 70, 'Cash', 'GHI789', 80);
select*from payment;
```

payment 1 x

select*from payment

	123 payment_id	123 trip_id	123 amount	noc payment_method	noc transaction_id	123 total_amount	123 additional_charges
1	4	13	50	Credit Card	ABC123	60	[NULL]
2	5	14	60	PayPal	DEF456	70	[NULL]
3	6	15	70	Cash	GHI789	80	[NULL]

Refresh Save Cancel Export data 200 3 3 row(s) fetched - 1ms, on 2024-04-22 at 21:24:04

IST en Writable Smart Insert 83: 1 [20] Sel: 20 | 1

• FEEDBACK TABLE:

```
create table feedback(
feedback_id int auto_increment primary key,
trip_id int,
foreign key (trip_id) references trip(trip_id),
rating int,
comments varchar(255)
);
```

```
insert into feedback (trip_id, rating, comments) values
(13, 4, 'Great service overall. The driver was friendly and punctual. '),
(14, 3, 'Average experience. Vehicle cleanliness could be improved. '),
(15, 5, 'Excellent trip! Everything went smoothly. ');
select * from feedback;
```

feedback 1 ×

select*from feedback Enter a SQL expression to filter results (use Ctrl+Space)

	123 feedback_id	123 trip_id	123 rating	asc comments
1	1	13	4	Great service overall. The driver was friendly and punctual.
2	2	14	3	Average experience. Vehicle cleanliness could be improved.
3	3	15	5	Excellent trip! Everything went smoothly.

Refresh Save Cancel Export data 200 3 3 row(s) fetched - 1ms, on 2024-04-22 at 21:25:30