

# RIDE SHARING APPLICATION

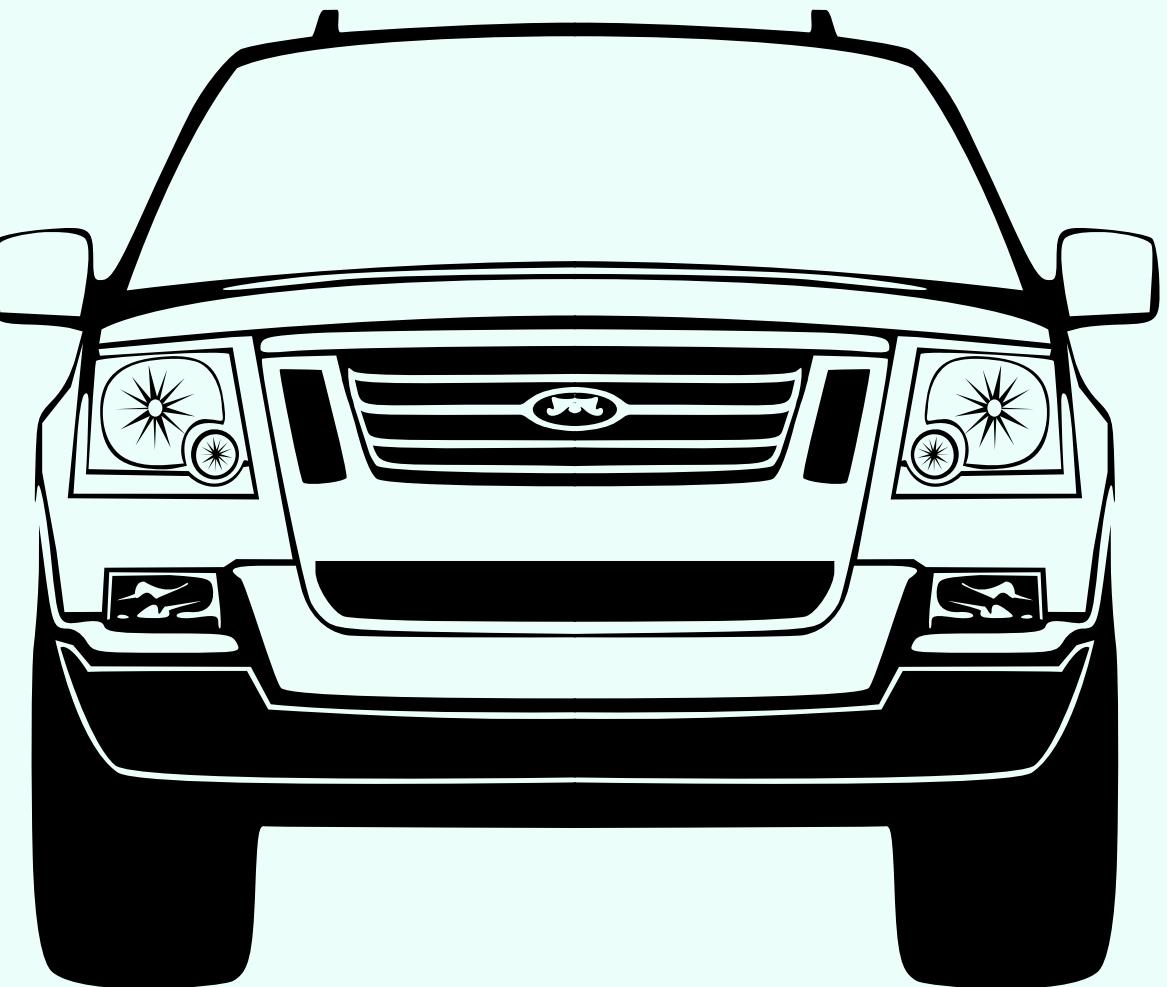
...

Simran - 191627101

Deepti - 191627080

Akarsh - 191627002

Shreyan - 191627083



NEXT →





**This Application  
facilitates ride-sharing  
by giving users a choice  
to either offer or find  
rides.**



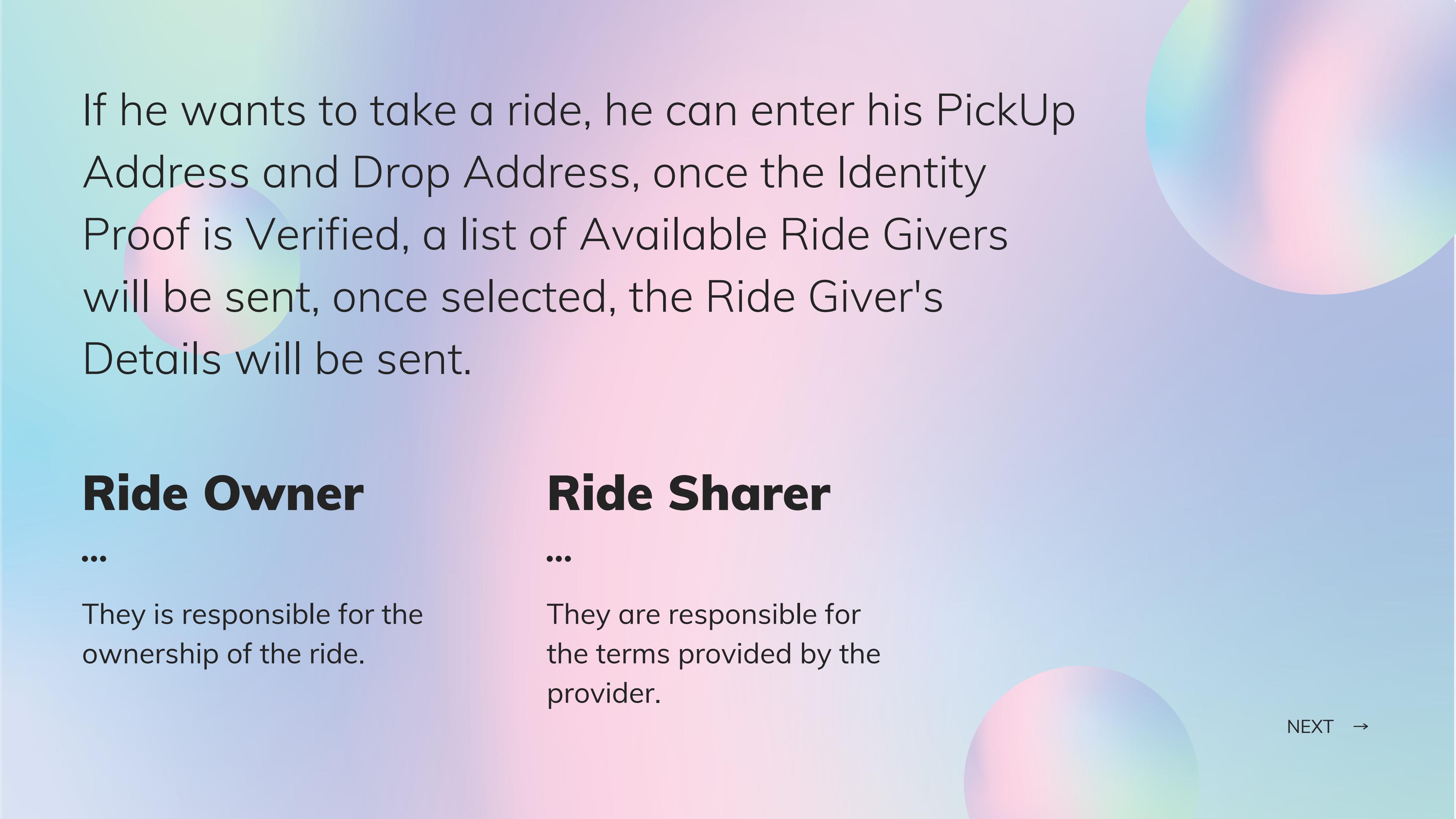
# Abstract

**In simple terms, carpooling/ride-sharing refers to sharing empty seats in a vehicle. People commuting from one point to another can share empty seats with other passengers who are travelling on the same route.**

**When the application is launched, the user gets to make an account and pick to either offer a ride or take a ride.**

**A ride taker gets to share empty seats in ride giver's vehicle and has a comfortable commute.**

NEXT →



If he wants to take a ride, he can enter his PickUp Address and Drop Address, once the Identity Proof is Verified, a list of Available Ride Givers will be sent, once selected, the Ride Giver's Details will be sent.

## Ride Owner

...

They is responsible for the ownership of the ride.

## Ride Sharer

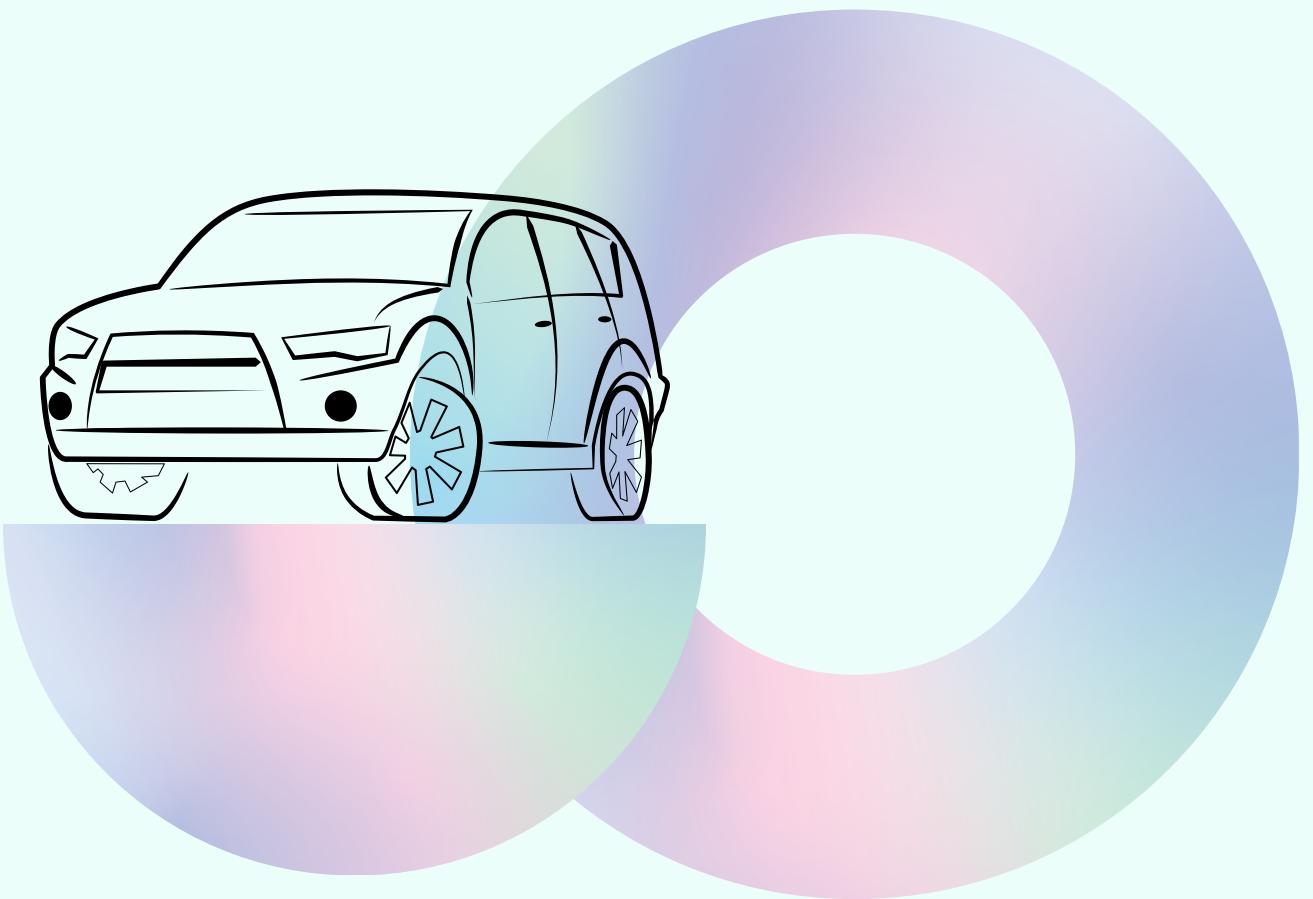
...

They are responsible for the terms provided by the provider.

NEXT →

# Objectives

- To Develop a Database System for a Ride Sharing Application that provides maximum efficiency and minimal operating complexity
- Show the process of creating the database from initial tables to inserting values to query outputs

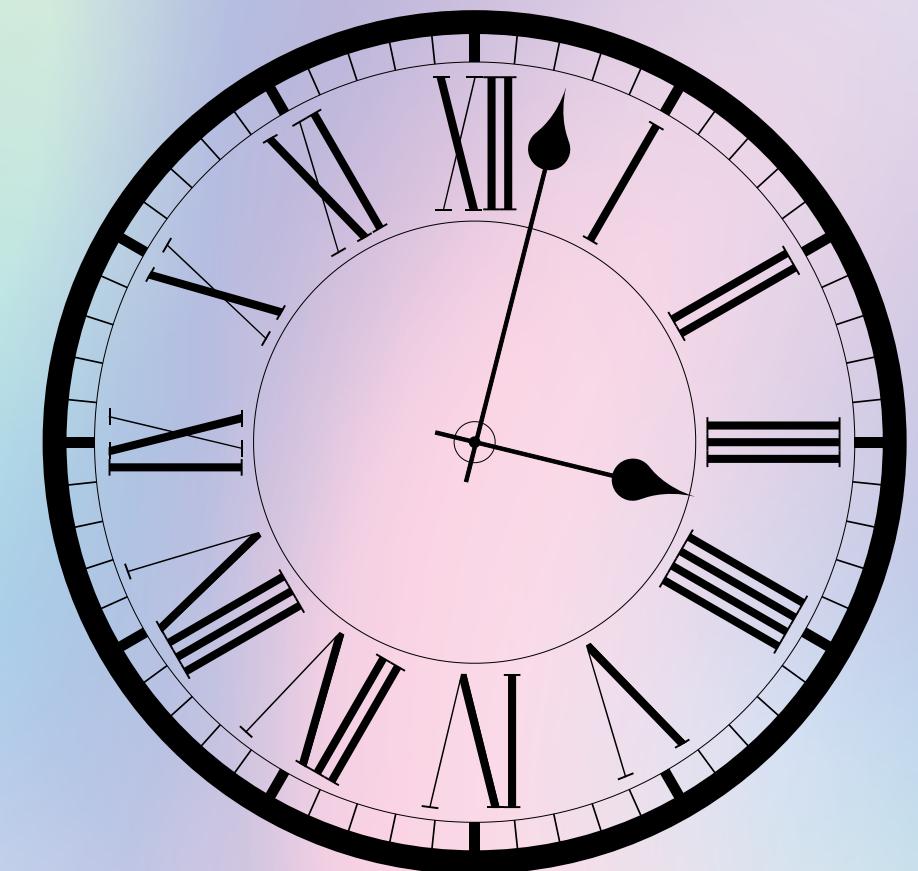


# Save Money

A ride giver reduces his/her commuting costs by sharing fuel expenses with other passengers.

...

If he wants to offer a ride, he can enter the StartAddress and EndAddress, once the Driving license is verified, he will get a list of Ride Takers to confirm, once confirmed, the Ride Taker's Details will be sent.



# Create Account

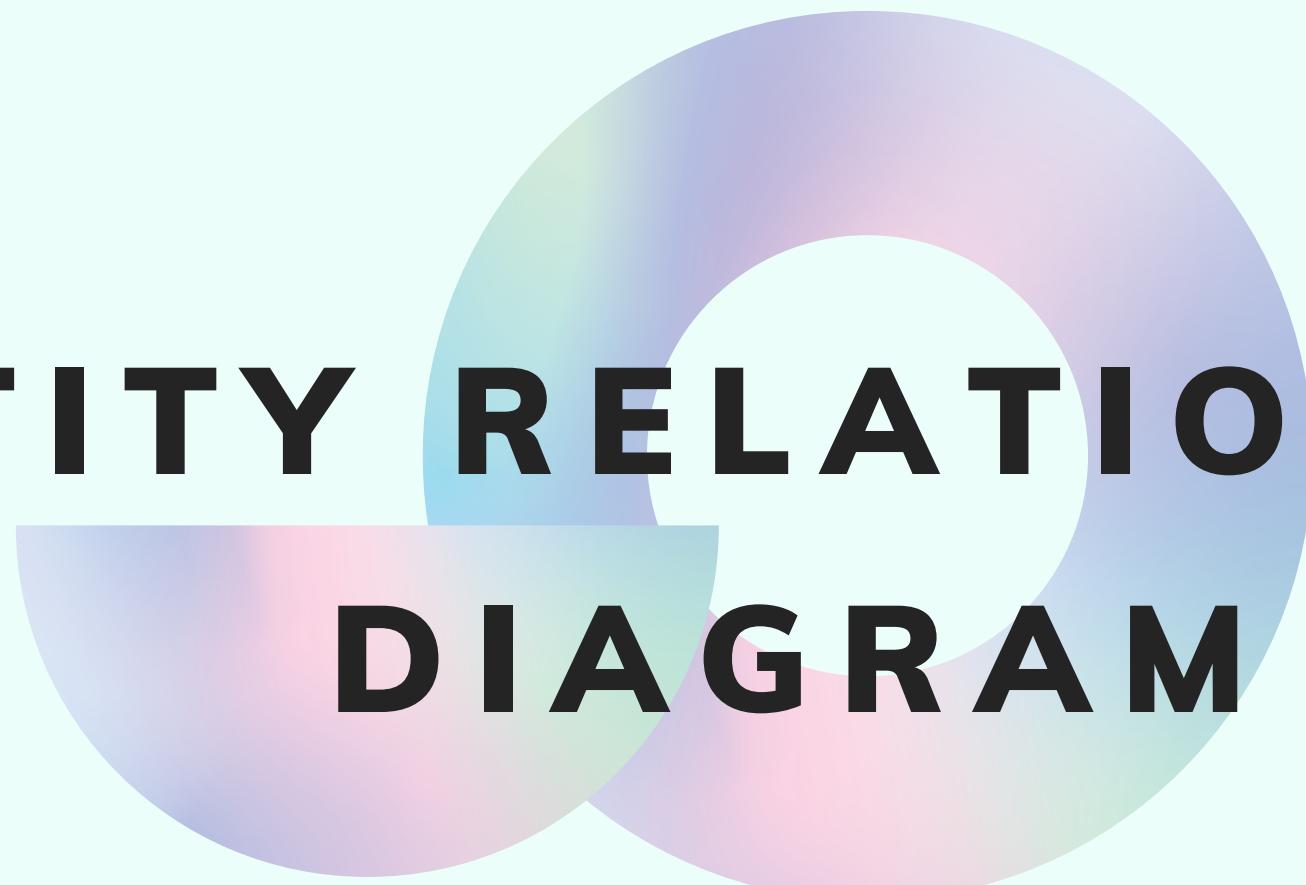


Profile: with details like Name, Picture, PhoneNo and emaillID, along with a list of Identity Proofs(Driving License, Aadhar Card etc)

- Vehicles: to view his available vehicles
- Rides: with the history of rides along with payment details
- Eco-meter: to view how much CO2 is being saved and in turn you save the world!!

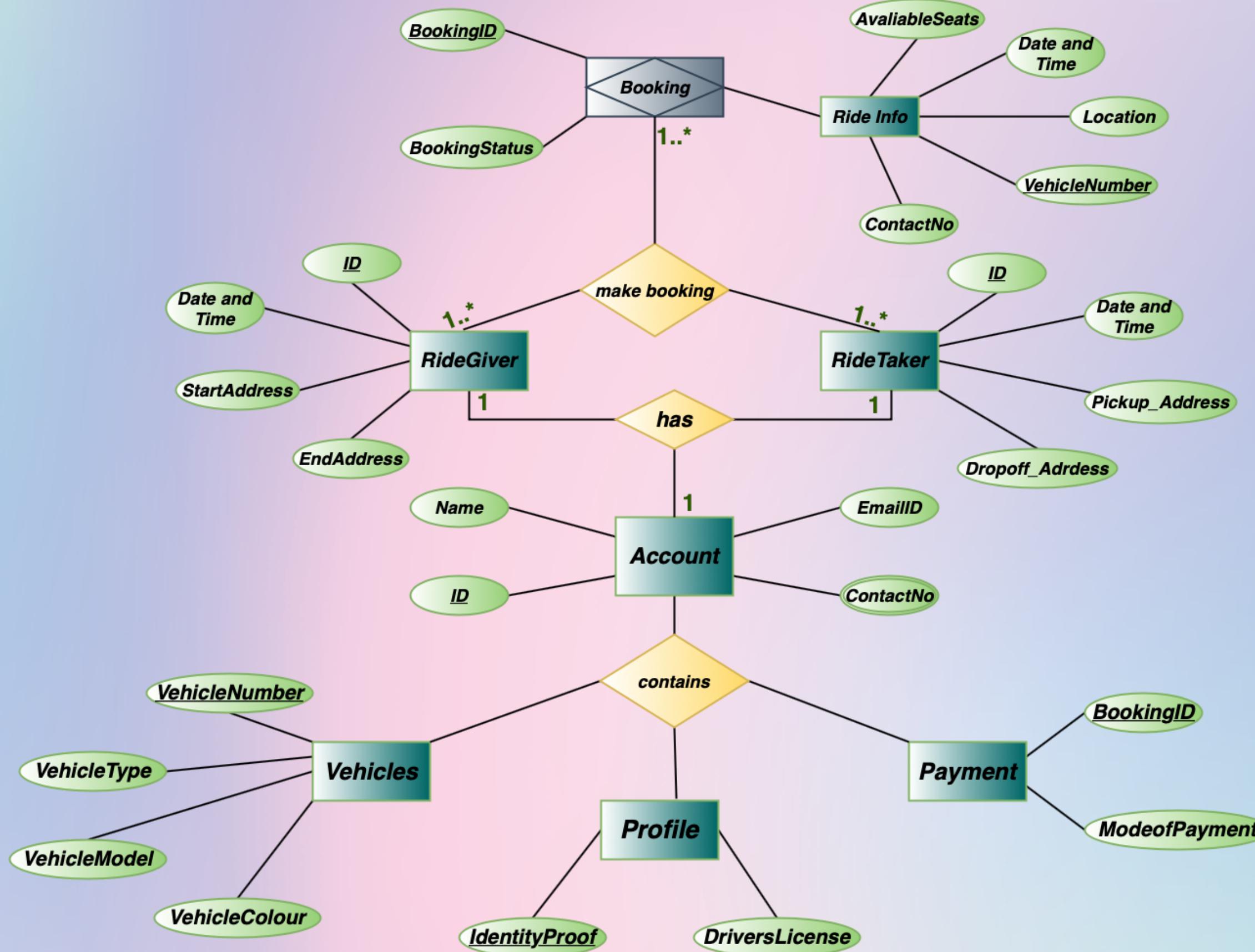
...

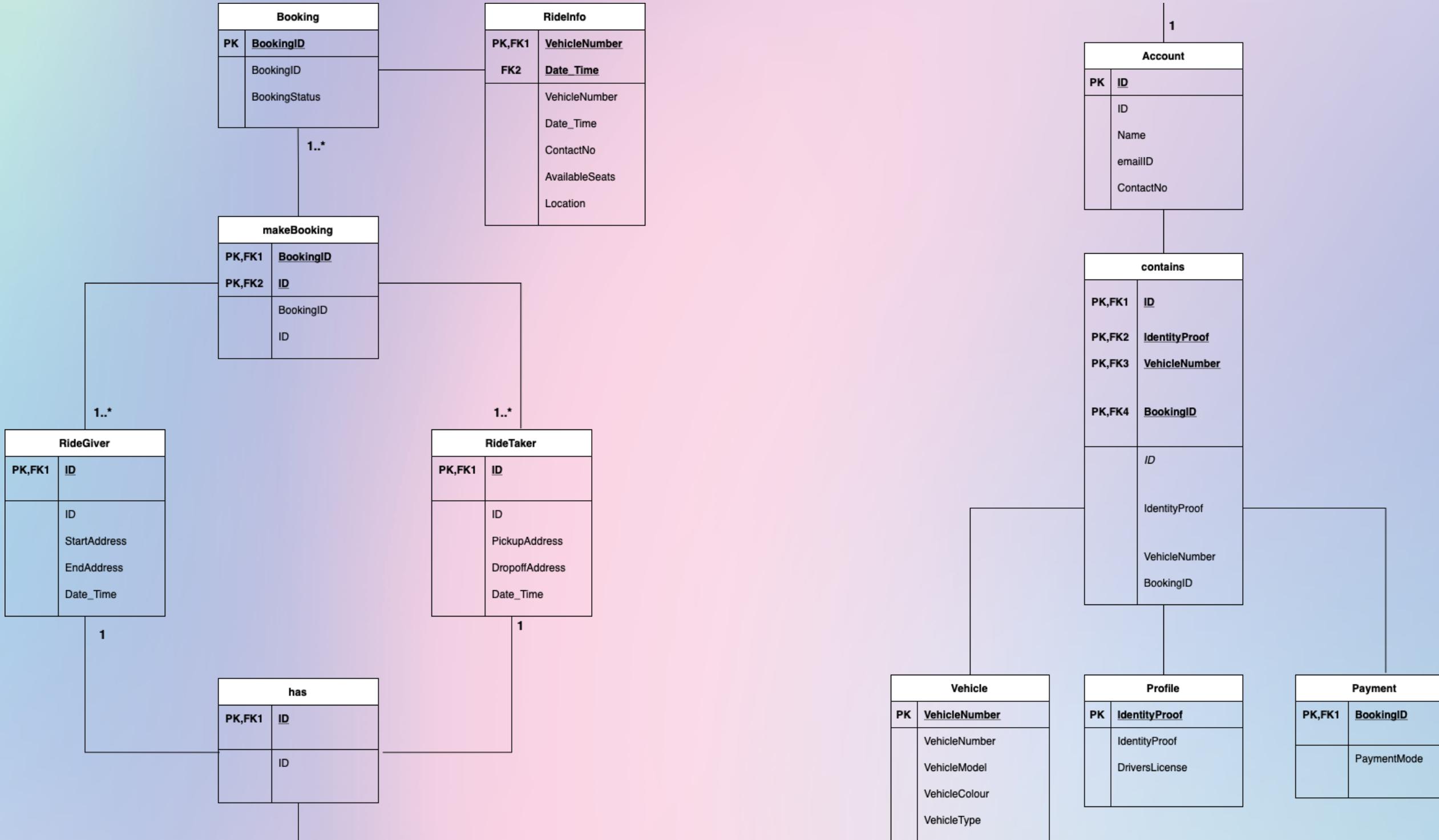
A culture of carpooling within helps you save money, reduce Carbon Footprint by reducing the number of vehicles on road, help your employees network better, get to work happier, and find parking without any stress.



# **ENTITY RELATIONSHIP DIAGRAM**

NEXT →





```
create table Account(  
Name varchar(10),  
ID number(10),  
emailID varchar(20),  
ContactNo number(10),  
primary key(ID) );  
desc Account  
drop table Account;  
insert into Account  
values('Simran',101,'simran@rider.com',9992221111);  
insert into Account  
values('Akarsh',002,'ache@rider.com',9991112222);  
insert into Account  
values('Deepti',080,'deepti@rider.com',9992223333);  
insert into Account  
values('Shreyan',083,'reddy@rider.com',9993331111);  
insert into Account  
values('Richa',051,'richa@manipal.com',9993332222);  
select * from Account;
```

# 1. ACCOUNT

NAME	ID	EMAILID	CONTACTNO
Deepti	80	deepti@rider.com	9992223333
Richa	51	richa@manipal.com	9993332222
Simran	101	simran@rider.com	9992221111
Akarsh	2	ache@rider.com	9991112222
Shreyan	83	reddy@rider.com	9993331111

[Download CSV](#)

5 rows selected.

## 2. RIDEGIVER

```
create table RideGiver(  
ID number(10),  
date_time timestamp,  
StartAddress varchar(25),  
EndAddress varchar(25),  
primary key(ID),  
Foreign key(ID) references Account  
);
```

```
Insert into RideGiver values(101,timestamp '2021-04-05 10:25:30' , 'Royal  
Embassy','Tiger Circle');
```

```
Insert into RideGiver values(80,timestamp '2021-02-21 11:30:40' , 'Manipal  
Stores',' AB5');
```

```
Insert into RideGiver values(51,timestamp '2021-01-18 17:17:17' , ' Student  
Plaza',' Malpe Beach');
```

```
select * from RideGiver;
```

ID	DATE_TIME	STARTADDRESS	ENDADDRESS
51	18-JAN-21 05.17.17.000000 PM	Student Plaza	Malpe Beach
101	05-APR-21 10.25.30.000000 AM	Royal Embassy	Tiger Circle
80	21-FEB-21 11.30.40.000000 AM	Manipal Stores	AB5

[Download CSV](#)

3 rows selected.

NEXT →

### 3. RIDETAKER

```
create table RideTaker(  
ID number(10),  
date_time timestamp,  
pickup_address varchar(25),  
dropoff_address varchar(25),  
primary key(ID),  
Foreign key(ID) references Account  
);  
Insert into RideTaker values(002,timestamp '2021-01-22 16:15:21','Shack Point','End Point');  
Insert into RideTaker values(083,timestamp '2021-03-17 19:28:43','MIT Library','Mandavi Emerald');  
select * from RideTaker;
```

ID	DATE_TIME	PICKUP_ADDRESS	DROPOFF_ADDRESS
2	22-JAN-21 04.15.21.000000 PM	Shack Point	End Point
83	17-MAR-21 07.28.43.000000 PM	MIT Library	Mandavi Emerald

[Download CSV](#)

2 rows selected.

## 4. Profile

```
create table Profile(  
IdentityProof number(12),  
DriversLicense varchar(7),  
primary key(IdentityProof)  
);  
insert into Profile values(111111111111,'UP23456');  
insert into Profile values(333333333333,'TS98765');  
insert into Profile values(555555555555,'GA55555');  
insert into Profile values(444444444444, NULL);  
  
SELECT * FROM Profile
```

<b>IDENTITYPROOF</b>	<b>DRIVERSLICENSE</b>
444444444444	-
111111111111	UP23456
333333333333	TS98765
555555555555	GA55555

[Download CSV](#)

4 rows selected.

# 5. Vehicle

VEHICLENUMBER	VEHICLETYPE	VEHICLEMODEL	VEHICLECOLOUR
GA07RT5555	Sedan	Mini Cooper	Green
UP05EU1111	Sedan	BMW Series 5	Black
TS08BL7777	SUV	Lamborghini Urus	Yellow

[Download CSV](#)

3 rows selected.

```
create table Vehicle(  
    VehicleNumber varchar(10),  
    VehicleType varchar(10) check (VehicleType in ('Sedan','SUV','Mini')),  
    VehicleModel varchar(20),  
    VehicleColour varchar(10),  
    primary key (VehicleNumber)  
);  
insert into Vehicle values('UP05EU1111','Sedan','BMW Series 5','Black');  
insert into Vehicle values('TS08BL7777','SUV','Lamborghini Urus','Yellow');  
insert into Vehicle values('GA07RT5555','Sedan','Mini Cooper','Green');  
select * from Vehicle;
```

# 6. Booking

```
Create table Booking(  
BookingID number(10),  
BookingStatus varchar(10) check (BookingStatus in  
('Completed','Approved','Scheduled','Pending','Cancelled')),  
primary key(BookingID)  
);  
insert into Booking values(67845674,'Completed');  
insert into Booking values(4566789,'Scheduled');  
insert into Booking values(9872323,'Pending');  
insert into Booking values(13347892,'Cancelled');
```

```
Select * from Booking;
```

BOOKINGID	BOOKINGSTATUS
9872323	Pending
13347892	Cancelled
4566789	Scheduled
67845674	Completed

[Download CSV](#)

4 rows selected.

## 7. makeBooking

```
Create table makeBooking(  
BookingID number(10),  
ID number(10),  
Primary key(BookingID, ID)  
);  
insert into makeBooking values(67845674,080);  
insert into makeBooking values(4566789,101);  
insert into makeBooking values(9872323,002);  
insert into makeBooking values(13347892,083);  
select * from makeBooking;
```

BOOKINGID	ID
4566789	101
9872323	2
13347892	83
67845674	80

[Download CSV](#)  
4 rows selected.

## 8. Payment

```
reate table Payment(  
BookingID number(10),  
PaymentMode varchar(10) check (PaymentMode in('Cash','Card','UPI')),  
primary key(bookingID),  
Foreign key(BookingID) references Booking  
);  
insert into Payment values(67845674,'UPI');  
insert into Payment values(4566789,'Cash');  
insert into Payment values(9872323,'Card');  
insert into Payment values(13347892,'Card');  
select * from Payment;
```

BOOKINGID	PAYMENTMODE
67845674	UPI
4566789	Cash
9872323	Card
13347892	Card

[Download CSV](#)

4 rows selected.

NEXT →

## 9. RidelInfo

```
create table RidelInfo(  
    AvailableSeats varchar(10),  
    Location varchar(10),  
    VehicleNumber varchar(10),  
    ContactNo varchar(10),  
    date_time timestamp,  
    primary key(VehicleNumber)  
);  
insert into RidelInfo values('2','Bareilly','UP05EU1111','9992221111',timestamp  
'2021-02-21 11:30:40');  
insert into RidelInfo  
values('1','Hyderabad','TS08BL7777','9992223333',timestamp '2021-04-05  
10:25:30');  
insert into RidelInfo values('3','Goa','GA07RT5555','9993332222',timestamp  
'2021-01-18 17:17:17');  
select * from RidelInfo;
```

AVAILABLESEATS	LOCATION	VEHICLENUMBER	CONTACTNO	DATE_TIME
2	Bareilly	UP05EU1111	9992221111	21-FEB-21 11.30.40.000000 AM
1	Hyderabad	TS08BL7777	9992223333	05-APR-21 10.25.30.000000 AM
3	Goa	GA07RT5555	9993332222	18-JAN-21 05.17.17.000000 PM

[Download CSV](#)

3 rows selected.

## 10. has

```
Create table has(  
ID number(10),  
primary key (ID),  
Foreign key(ID) references Account  
);  
insert into has values(101);  
insert into has values(080);  
insert into has values(051);  
insert into has values(083);  
insert into has values(002);  
select * from has;
```

ID
2
51
80
83
101

[Download CSV](#)  
5 rows selected.

NEXT →

# 11. contains

```
create table contains(
    ID number(10),
    BookingID number(10),
    VehicleNumber varchar(10),
    IdentityProof number(12),
    primary key(ID,BookingID, VehicleNumber, IdentityProof),
    Foreign key(ID) references Account,
    Foreign key(BookingID) references Payment,
    Foreign key(VehicleNumber) references Vehicle,
    Foreign key(IdentityProof) references Profile
);
Insert into contains values (101,4566789,'UP05EU1111',111111111111);
Insert into contains values (080,67845674,'TS08BL7777',333333333333);
Insert into contains values (101,9872323,'GA07RT5555',555555555555);
select * from contains;
```

ID	BOOKINGID	VEHICLENUMBER	IDENTITYPROOF
80	67845674	TS08BL7777	333333333333
101	4566789	UP05EU1111	111111111111
101	9872323	GA07RT5555	555555555555

[Download CSV](#)

3 rows selected.

# QUERY 1

```
update RideGiver set StartAddress='Canara Mall', EndAddress='Marena' where  
ID=101;  
select * from RideGiver;
```

ID	DATE_TIME	STARTADDRESS	ENDADDRESS
101	05-APR-21 10.25.30.000000 AM	Canara Mall	Marena
80	21-FEB-21 11.30.40.000000 AM	Manipal Stores	AB5
51	18-JAN-21 05.17.17.000000 PM	Student Plaza	Malpe Beach

[Download CSV](#)

3 rows selected.

NEXT →

# QUERY 2

```
select * from Profile natural join contains, Account where  
contains.ID=Account.ID;
```

IDENTITYPROOF	DRIVERSLICENSE	ID	BOOKINGID	VEHICLENUMBER	NAME	ID	EMAILID	CONTACTNO
333333333333	TS98765	80	67845674	TS08BL7777	Deepti	80	deepti@rider.com	9992223333
111111111111	UP23456	101	4566789	UP05EU1111	Simran	101	simran@rider.com	9992221111
555555555555	GA55555	101	9872323	GA07RT5555	Simran	101	simran@rider.com	9992221111

[Download CSV](#)

3 rows selected.

## QUERY 3

```
select ID, BookingID,BookingStatus  
from Booking natural join makeBooking;
```

ID	BOOKINGID	BOOKINGSTATUS
101	4566789	Scheduled
2	9872323	Pending
83	13347892	Cancelled
80	67845674	Completed

[Download CSV](#)

4 rows selected.

NEXT →

# QUERY 4

```
select * from RideGiver union select * from RideTaker;
```

ID	DATE_TIME	STARTADDRESS	ENDADDRESS
2	22-JAN-21 04.15.21.000000 PM	Shack Point	End Point
51	18-JAN-21 05.17.17.000000 PM	Student Plaza	Malpe Beach
80	21-FEB-21 11.30.40.000000 AM	Manipal Stores	AB5
83	17-MAR-21 07.28.43.000000 PM	MIT Library	Mandavi Emerald
101	05-APR-21 10.25.30.000000 AM	Canara Mall	Marena

[Download CSV](#)

5 rows selected.

NEXT →

# QUERY 5

```
select * from RideInfo order by AvailableSeats;
```

AVAILABLESEATS	LOCATION	VEHICLENUMBER	CONTACTNO	DATE_TIME
1	Hyderabad	TS08BL7777	9992223333	05-APR-21 10.25.30.000000 AM
2	Bareilly	UP05EU1111	9992221111	21-FEB-21 11.30.40.000000 AM
3	Goa	GA07RT5555	9993332222	18-JAN-21 05.17.17.000000 PM

[Download CSV](#)

3 rows selected.

# QUERY 6

```
select * from Account  
where emailID like '%@rider.com'  
order by Name;
```

NAME	ID	EMAILID	CONTACTNO
Akarsh	2	ache@rider.com	9991112222
Deepti	80	deepti@rider.com	9992223333
Shreyan	83	reddy@rider.com	9993331111
Simran	101	simran@rider.com	9992221111

[Download CSV](#)

4 rows selected.

## QUERY 7

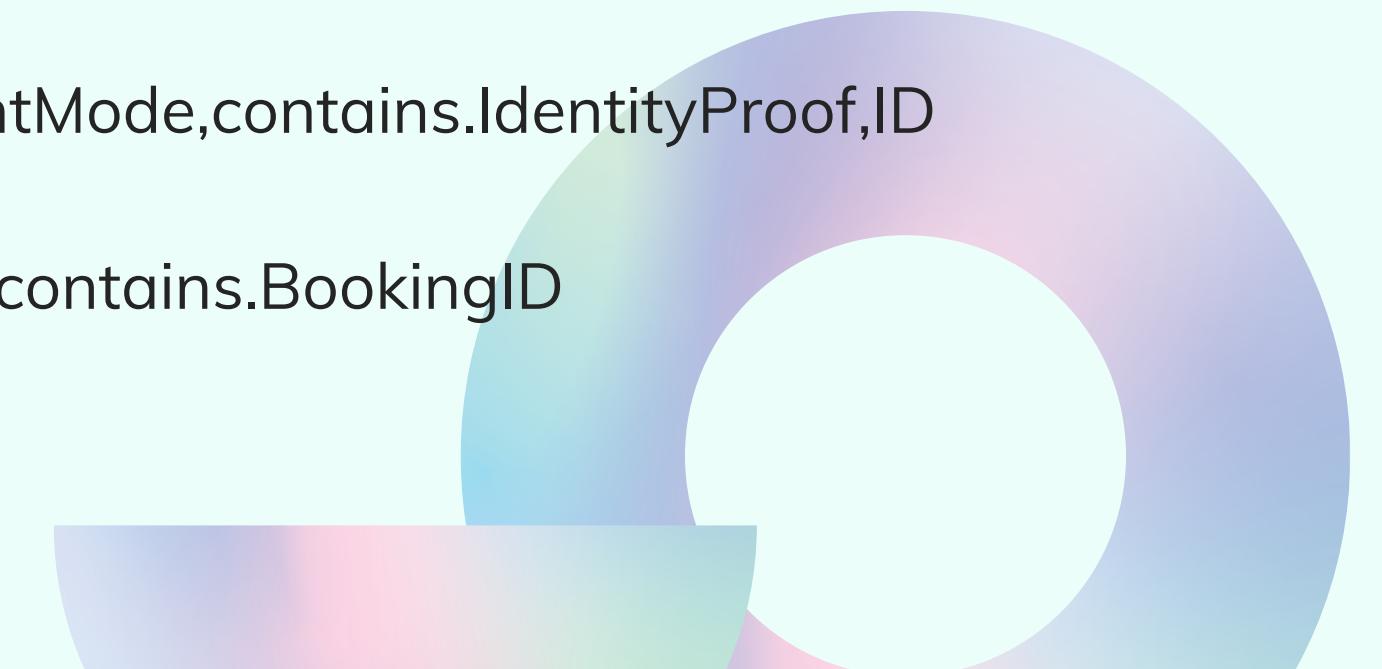
```
select count(Name) as T_Name  
from Account;
```

T_NAME
5

[Download CSV](#)

# QUERY 8

```
create view Pay as select  
Payment.BookingId,PaymentMode,contains.IdentityProof.ID  
from Payment,contains  
where Payment.BookingId=contains.BookingID  
and PaymentMode='Card';  
select * from Pay;
```



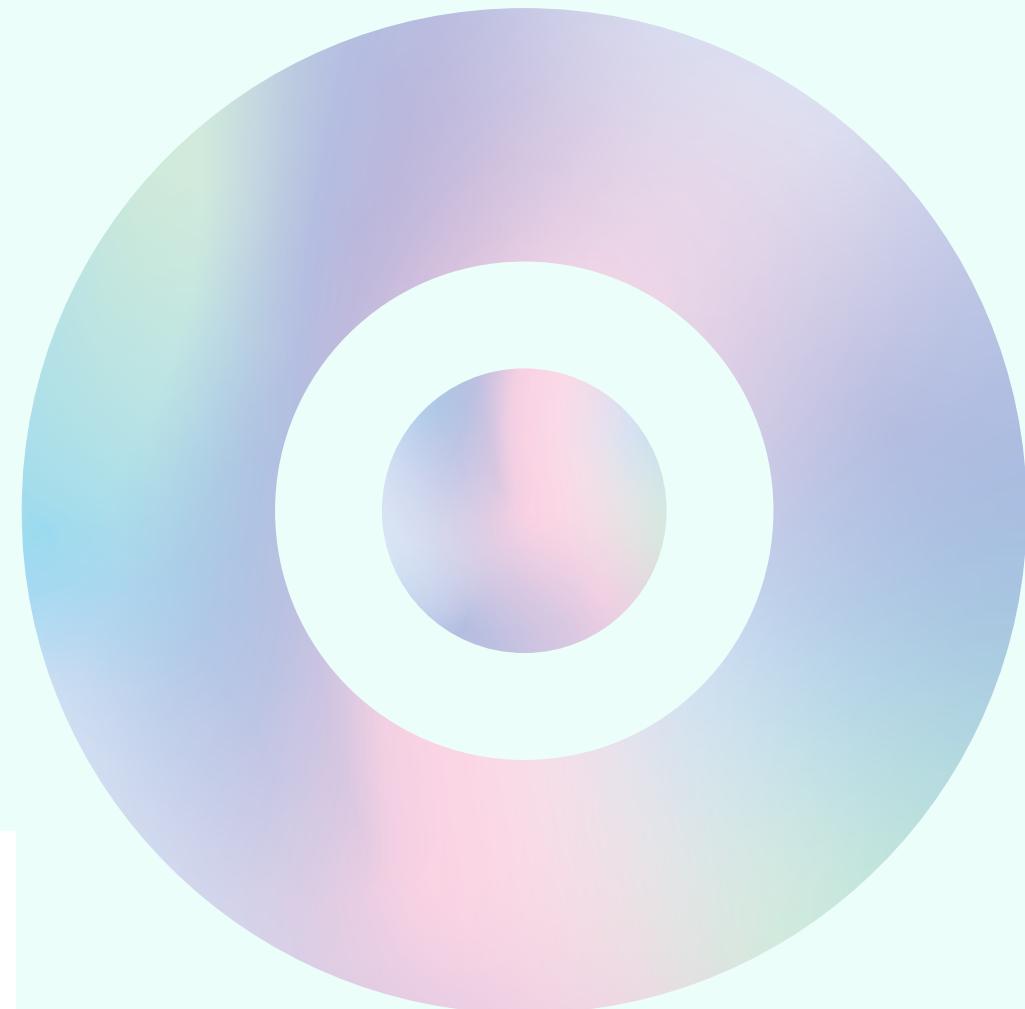
BOOKINGID	PAYMENTMODE	IDENTITYPROOF	ID
9872323	Card	555555555555	101

[Download CSV](#)

NEXT →

# QUERY 9

```
select BookingID,VehicleNumber,IdentityProof,DriversLicense  
from contains natural left outer join Profile
```



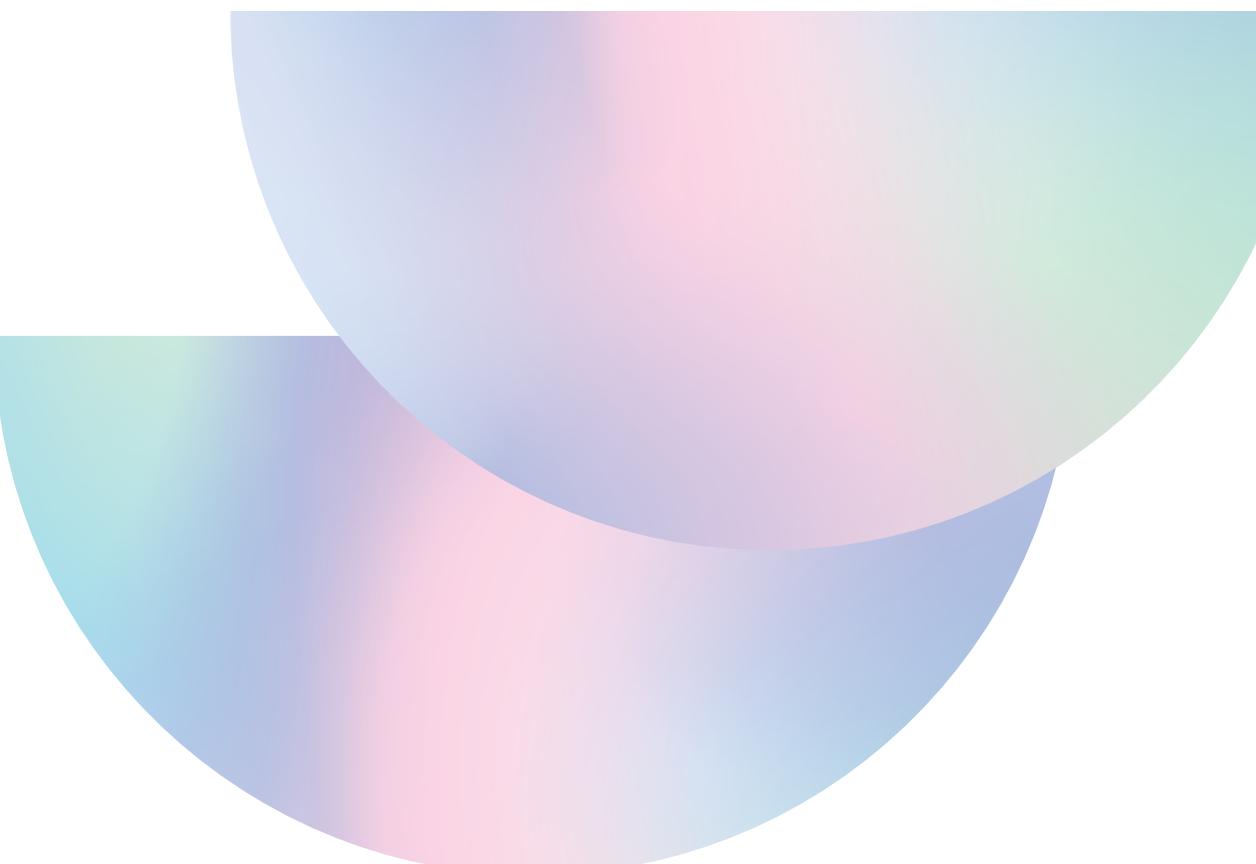
BOOKINGID	VEHICLENUMBER	IDENTITYPROOF	DRIVERSLICENSE
67845674	TS08BL7777	333333333333	TS98765
4566789	UP05EU1111	111111111111	UP23456
9872323	GA07RT5555	555555555555	GA55555

[Download CSV](#)

3 rows selected.

# QUERY 10

```
select BookingID,VehicleNumber,IdentityProof,DriversLicense  
from contains natural right outer join Profile
```



BOOKINGID	VEHICLENUMBER	IDENTITYPROOF	DRIVERSLICENSE
67845674	TS08BL7777	333333333333	TS98765
4566789	UP05EU1111	111111111111	UP23456
9872323	GA07RT5555	555555555555	GA55555

[Download CSV](#)

3 rows selected.

# Conclusion

This project is a report of the process of developing an efficient and easy to use database system for a Ride Sharing Application. It shows the project flow for the database system, initial tables as well a relational schema and subsequent database implementation. We have also provided a few sample codes for querying the database and additional screenshots showcasing the output of the queries.

## **Future Scope**

In spite of big players like Uber, Ola and Lyft capturing a significant portion of the cab and ride-sharing market, there is still scope left for new businesses wanting to venture into this field. For their business's success, they need to have a unique business model. Also, they need to get built a user friendly ride sharing app from an established mobile application development company in Chandigarh like Matrid Technologies.



