**HOUSE BOAT MANAGEMENT SYSTEM**

**Prepared For**

Dr. Kiran Sharma

Ass. Professor

BML Munjal University

**By**

D Veera Harsha Vardhan Reddy

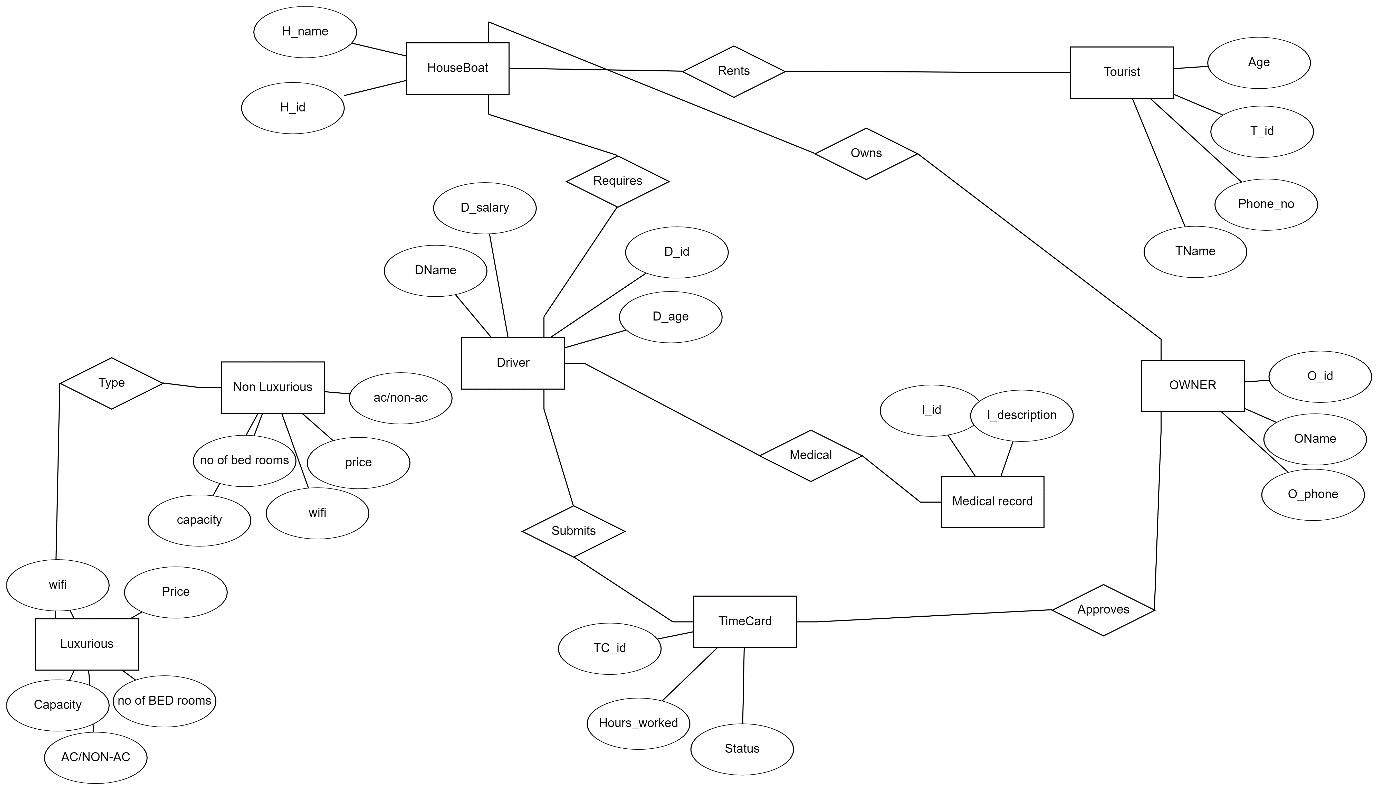
210C2030061

CSE-2



**Problem Statement:**

An agency keeps track of house boats, its owners and the customers who rented it.

**ER DIAGRAM**

**Relational Schema:** Primary Keys: \_\_\_\_\_\_\_

Foreign Keys: grey

Tourist(T\_id, TFirst\_name, TLast\_name, Age)

Rent(H\_id, T\_id, D\_id)

Driver(D\_id, DFirst\_name, DLast\_name, D\_salary, D\_age)

Medical\_Record(D\_id, I\_id, I\_description)

Time Card(TC\_id, Hours\_worked, Status, D\_id)

Owner(O\_id, OFirst\_name, OLast\_name, O\_phone, TC\_id)

Luxurious(H\_id, H\_name, O\_id, Capacity, AC/non AC, Noofbedrooms,Wifi, Price)

Non-Luxurious(H\_id,H\_name,O\_id,Capacity, AC/non AC, Noofbedrooms,Wifi, Price)

The relation between entities:

One to one relationship: Time card and Driver, Driver and Medical records.

One to many relationship: owner and luxurious, Owner and Non-Luxurious

Many to many relationship: Tourist and Luxurious, Tourist and Non-Luxurious.

**SQL:**

create table tourist(t\_id varchar(10) primary key, tfirst\_name varchar(20), tlast\_name varchar(20), age int, phone varchar(13), country varchar(20));

create table houseboat(h\_id varchar(10) primary key, h\_name varchar(20), capacity int, acfacility varchar(6), noofbedrooms int, wifi varchar(6), price int);

create table medicalrecords(i\_id varchar(10) primary key, idescription varchar(80));

create table driver(d\_id varchar(10) primary key, dfirst\_name varchar(20), dlast\_name varchar(20), dsalary int, dage int, i\_id varchar(10)) ;

create table boatowner(o\_id varchar(10) primary key, ofirst\_name varchar(20),olast\_name varchar(20), ophone varchar(13));

create table management(rent\_date date, t\_id varchar(10), h\_id varchar(10), d\_id varchar(10), o\_id varchar(10), foreign key(t\_id) references tourist(t\_id), foreign key(d\_id) references driver(d\_id), foreign key(h\_id) references houseboat(h\_id) , foreign key(o\_id) references boatowner(o\_id) , primary key(rent\_date, t\_id));

**Tourist Entity:**

insert into tourist values('T1', 'Sowmya', 'Khanna', 29, '+919446252773', 'India');

insert into tourist values('T2', 'Rajesh', 'Khanna', 34, '+919497822113', 'India');

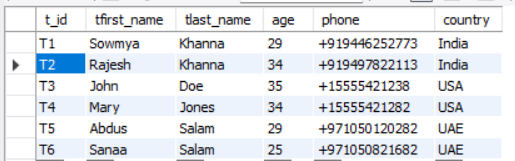
insert into tourist values('T3', 'John', 'Doe', 35, '+15555421238', 'USA');

insert into tourist values('T4', 'Mary', 'Jones', 34, '+15555421282', 'USA');

insert into tourist values('T5', 'Abdus', 'Salam', 29, '+971050120282', 'UAE');

insert into tourist values('T6', 'Sanaa', 'Salam', 25, '+971050821682', 'UAE');

select \* from tourist;



**HouseBoat Entity:**

insert into houseboat values('H123', 'Aqua Castle', 6, 'AC', 2, 'Y', 33810);

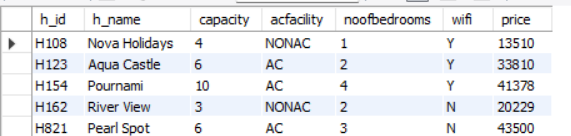
insert into houseboat values('H108', 'Nova Holidays', 4, 'NONAC', 1, 'Y', 13510);

insert into houseboat values('H154', 'Pournami', 10, 'AC', 4, 'Y', 41378);

insert into houseboat values('H162', 'River View', 3, 'NONAC', 2, 'N', 20229);

insert into houseboat values('H821', 'Pearl Spot', 6, 'AC', 3, 'N', 43500);

select \* from houseboat;



**Medical Records Entity:**

insert into medicalrecords values('I454','Diabeties');

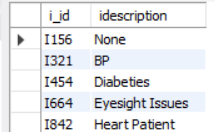
insert into medicalrecords values('I156','None');

insert into medicalrecords values('I664','Eyesight Issues');

insert into medicalrecords values('I321','BP');

insert into medicalrecords values('I842','Heart Patient');

select \* from medicalrecords;



**Driver Entity:**

insert into driver values('D454', 'Suresh', 'M. N', 12000, 40, 'I454');

insert into driver values('D156', 'Venu', 'S. K', 8000, 35, 'I156');

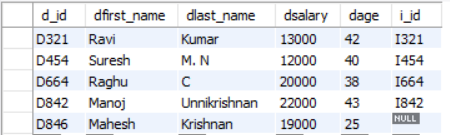
insert into driver values('D664', 'Raghu', 'C', 20000, 38, 'I664');

insert into driver values('D321', 'Ravi', 'Kumar', 13000, 42, 'I321');

insert into driver values('D842', 'Manoj', 'Unnikrishnan', 22000, 43, 'I842');

insert into driver values('D846', 'Mahesh', 'Krishnan', 19000, 25, NULL);

select \* from driver;



**BoatOwner Entity:**

insert into boatowner values('O123','Aravindan', 'Menon', '+919746576958');

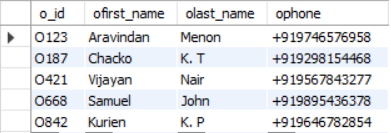
insert into boatowner values('O421','Vijayan', 'Nair', '+919567843277');

insert into boatowner values('O668','Samuel', 'John', '+919895436378');

insert into boatowner values('O187','Chacko', 'K. T', '+919298154468');

insert into boatowner values('O842','Kurien', 'K. P', '+919646782854');

select \* from boatowner;



**Management Values Entity:**

insert into management values('06-11-18', 'T1', 'H108', 'D156', 'O187');

insert into management values('06-11-18', 'T2', 'H108', 'D156', 'O187');

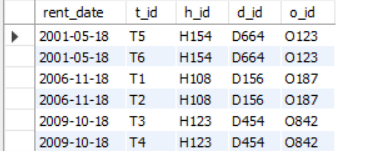
insert into management values('09-10-18', 'T3', 'H123', 'D454', 'O842');

insert into management values('09-10-18', 'T4', 'H123', 'D454', 'O842');

insert into management values('01-05-18', 'T5', 'H154', 'D664', 'O123');

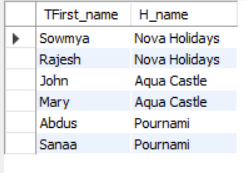
insert into management values('01-05-18', 'T6', 'H154', 'D664', 'O123');

select \* from management;

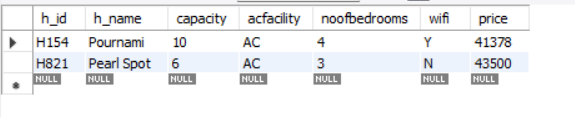


**QUERIES:**

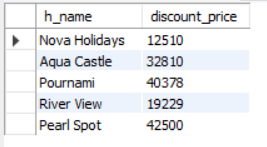
1. **Display first name of each tourist and the name of the houseboat that they have rented.**
2. select Tourist.TFirst\_name, Houseboat.H\_name from ((Management INNER JOIN Tourist ON Management.T\_id = Tourist.T\_id) INNER JOIN Houseboat ON Management.H\_id = Houseboat.H\_id);



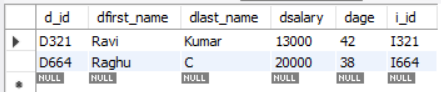
1. **Display the details of each housboat whose price is greater than 40000.**
2. select \* from houseboat where price>40000;



1. **Display the name and the discount price of each houseboat, where the discount price = price – 1000.**
2. select h\_name, price-1000 as discount\_price from houseboat;

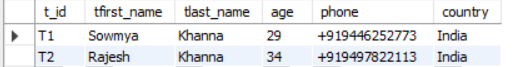


1. **Select details of drivers whose first name starts with R.**
2. select \* from driver where dfirst\_name like 'R%';

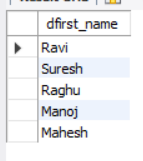


**Select details of tourist whose last name has a h in it.**

1. select \* from tourist where tlast\_name like '%h%';



1. **Display first name from driver whose salary is greater than 9000**
2. select dfirst\_name from driver where dsalary in (select dsalary from driver where dsalary>9000);

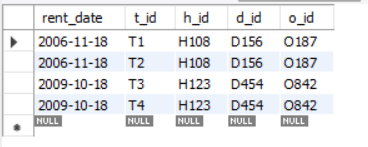


**Display first name of driver whose salary is not greater than 9000,**

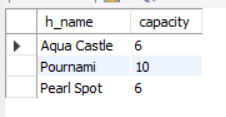
1. select dfirst\_name from driver where dsalary not in (select dsalary from driver where dsalary>9000);



1. **Display details of records where rent dates is between '06-10-18' and '09-11-18'**
2. select \* from Management where Rent\_date BETWEEN '06-10-18' AND '09-11-18';

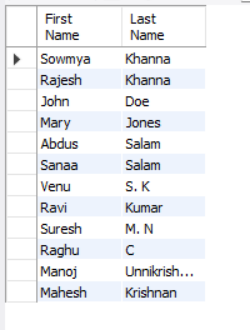


1. **Display houseboat name and capacity of houseboats that have ac facility**
2. select h\_name, capacity from houseboat where h\_id in(select h\_id from houseboat where acfacility like 'AC');



**8)Display first name and last name of all tourists and drivers,**

A) select tfirst\_name "First Name", tlast\_name "Last Name" from tourist union select dfirst\_name, dlast\_name from driver;



**Normalization:**

The Normalization process can be shown as follows:

A = Tid

B = TFirst\_name

C = TLast\_name

D = Age

E = Phone

F = Country

G = H\_id

H = H\_name

I = Capacity

J = AC/NON-AC

K = Bedroom

L = Wifi

M = Price

N = Rent\_date

O = D\_id

P = DFirst\_name

Q = DLast\_name

R = D\_salary

S = D\_age

T = I\_id

U = I\_description

V = O\_id

W = OFirst\_name,

X = OLast\_name,

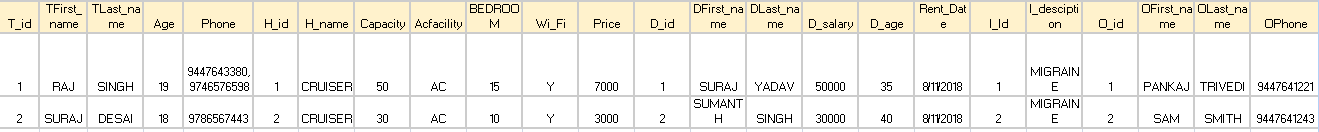
Y = OPhone.

0NF Form:

In the 0NF Form, we list all the attributes as follows:

T\_id, TFirst\_name, TLast\_name, Age, Phone, Country, H\_id, H\_name, Capacity, Acfacility, Bedroom, Wifi, Price, Rent\_date, D\_id, DFirst\_name, DLast\_name, D\_salary, D\_age, I\_id, I\_description, O\_id, OFirst\_name, OLast\_name, OPhone.

We combined all of the attributes into a single table. The table is shown here as follows, with a few values added in the table:

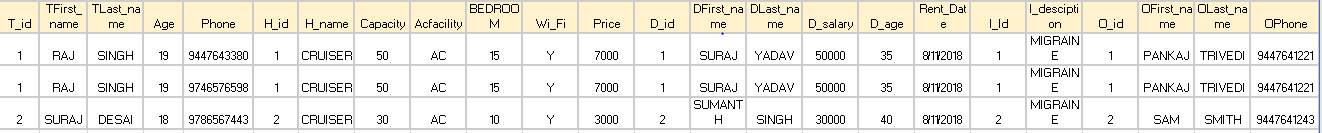


Functional dependencies are:

AGNOV -> ABCDEFGHIJKLMNOPQRSTUVWXY

1NF Form:

In the 1NF Form, we split the rows that had the multivalued attributes, so that each column can have unique values.



Functional Dependencies:

AGNOV -> ABCDEFGHIJKLMNOPQRSTUVWXY

2NF Form:

In the 2NF Form, the partial dependencies are removed, and the tables can be split as follows:

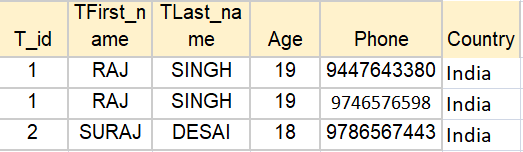
Tourist( T\_id, TFirst\_name, TLast\_name, Age, Phone, Country)

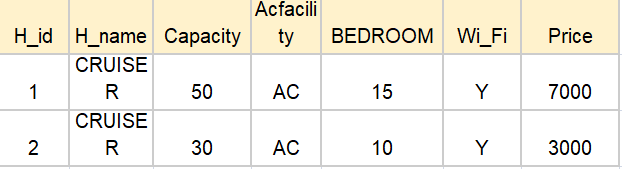
Houseboat(H\_id, H\_name, Capacity, Acfacility, Bedroom, Wifi, Price)

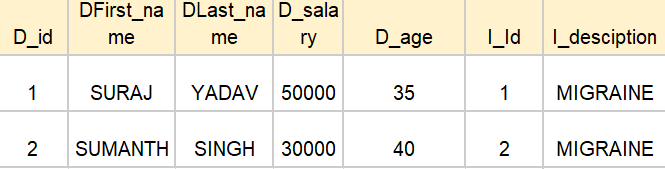
Driver(D\_id, DFirst\_name, DLast\_name, D\_salary, D\_age, I\_id, I\_description)

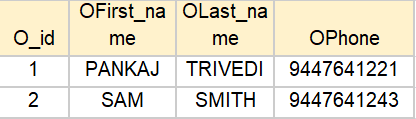
Owner(O\_id, OFirst\_name, OLast\_name, OPhone)

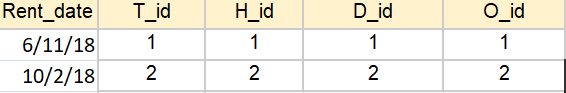
Management(Rent\_date, T\_id, H\_id, D\_id, O\_id)











The Tourist attributes have partial dependencies only on T\_id.

The Houseboat attributes have partial dependencies only on H\_id.

The Driver attributes have partial dependencies only on D\_id.

The Medical\_Record attributes have partial dependencies only on I\_id.

The Owner attributes have partial dependencies only on O\_id.

The Management entitiy contains the primary keys of all the other entities.

Functional Dependencies are:

A -> BCDEF

G -> HIJKLM

O -> PQRSTU

T -> U

V ->WXY

3NF Form:

In the 3NF Form, the tables can be further divided as follows:

Tourist( T\_id, TFirst\_name, TLast\_name, Age, Phone, Country)

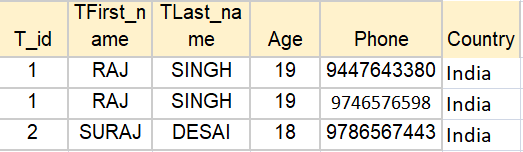
Houseboat(H\_id, H\_name, Capacity, AC/NON-AC, Bedroom, Wifi, Price)

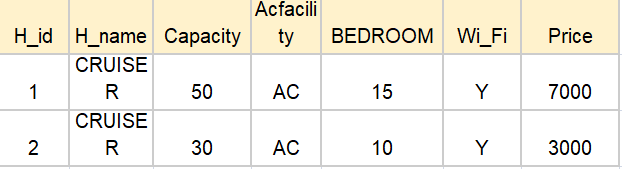
Driver(D\_id, DFirst\_name, DLast\_name, D\_salary, D\_age, I\_id)

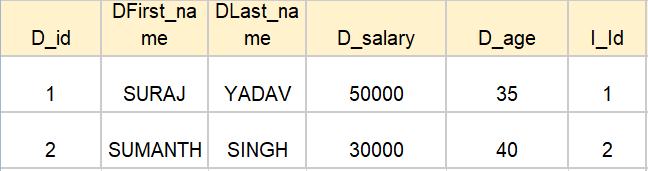
Medical\_Record(I\_id, I\_description)

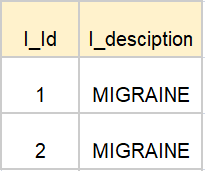
Owner(O\_id, OFirst\_name, OLast\_name, OPhone)

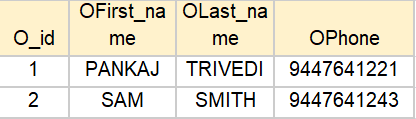
Management(Rent\_date, T\_id, H\_id, D\_id, O\_id)

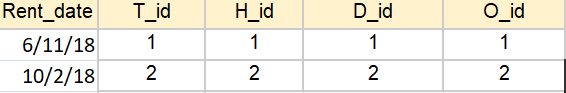












In the 3NF form, we remove the transitive dependencies. The transitive dependency here is found in the Driver entity, where the I\_description depends only on I\_id.

Functional Dependencies are:

A -> BCDEF

G -> HIJKLM

O -> PQRST

T -> U

V ->WXY