DBMS - Mini Project

Apartment Management System

Submitted By:

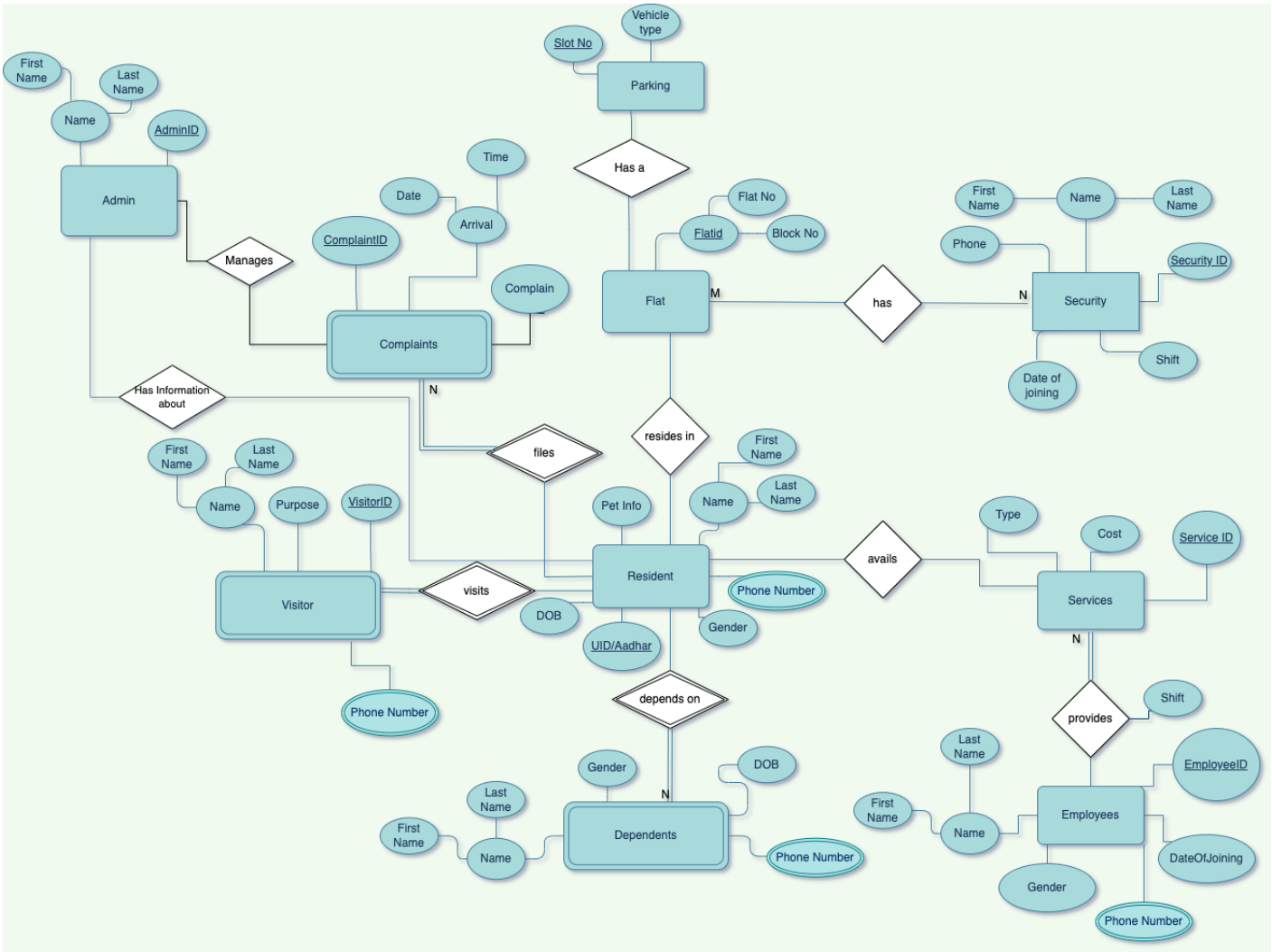
Thejas N U
PES1UG20CS606
V Semester J section

Description and Scope of the project:

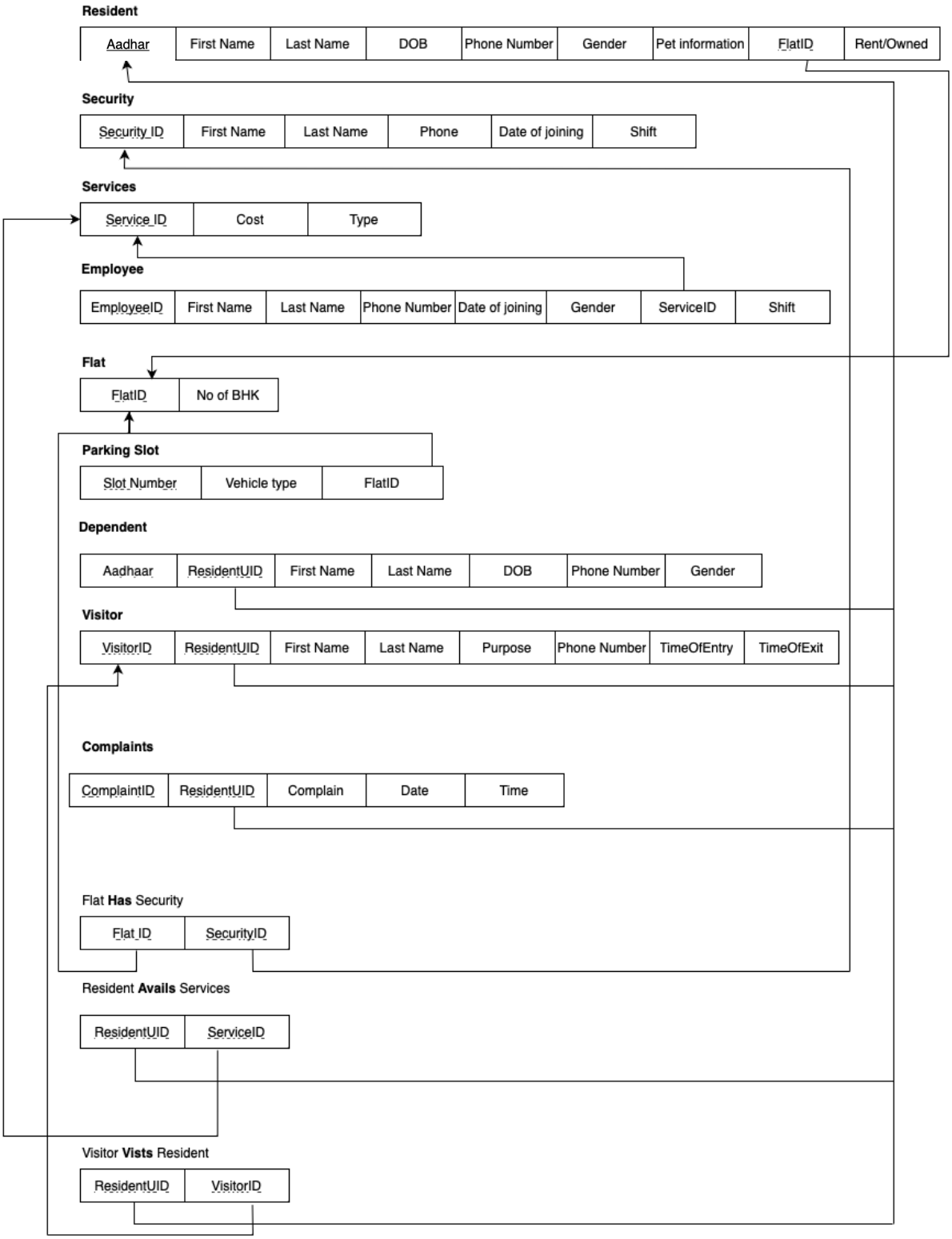
This project aims at automating the apartment management . It can keep track of residents, employees, visitors and all flats. Admin can login and update all details, whereas residents can login to avail any services or to give complaints and security guards will have the power to add and update the visitor details.

This project can be implemented in any apartment and all the data can be managed easily without any physical copies of records.

ER diagram:



Relational schema diagram:



DDL statements-Building the database:

```
-- Creating a new database
create database apna_ghar;

-- Creating resident table
create table resident
(
   Aadhar varchar(12) not null,
   Fname varchar(15) not null,
   Lname varchar(15),
   DOB date not null,
   phone varchar(10) not null,
   gender varchar(2) not null,
   pet_info varchar(20) default 'no pet',
   FlatID varchar(10),
   rent_owned varchar(10),
   primary key (Aadhar),
   foreign key(FlatID) references flat(FlatID)
);

-- Creating security table
create table security
(
   securityID varchar(10) not null,
   Fname varchar(15) not null,
   Lname varchar(15),
   phone varchar(10) not null,
   doj date not null,
   shift varchar(5) default 'day',
   primary key (securityID)
);

-- Creating services table
create table services
(
   serviceID varchar(10) not null,
   cost real,
   type varchar(20),
   primary key (serviceID)
);

-- Creating employee table
create table employee
(
   employeeID varchar(10) not null,
   Fname varchar(15) not null,
   Lname varchar(15),
   phone varchar(10) not null,
   doj date not null,
   gender varchar(2) not null,
   shift varchar(5) default 'day',
   serviceID varchar(10) not null,
   salary real not null check(salary>15000),
   primary key (employeeID),
   foreign key (serviceID) references services(serviceID)
);

-- Creating flat table
```

```sql
create table flat
(
   FlatID varchar(10) primary key not null,
   no_bhk int not null
);

-- Creating parking_slot table
create table parking_slot
(
   slotNo integer not null,
   vehicle_type varchar(10),
   flatID varchar(10) not null,
   primary key (slotNo),
   foreign key (flatID) references flat(flatID)
);

-- Creating dependent table
create table dependent
(
   Aadhar varchar(12) not null,
   residentUID varchar(12) not null,
   Fname varchar(15) not null,
   Lname varchar(15),
   DOB date not null,
   phone varchar(10) not null,
   gender varchar(2) not null,
   primary key (Aadhar),
   foreign key(residentUID) references resident(Aadhar)
);

-- Creating visitor table
create table visitor
(
   visitorID varchar(10) not null,
   flatID varchar(10) not null,
   Fname char(15) not null,
   Lname char(15),
   purpose varchar(100),
   phonenumber varchar(10),
   time_of_entry timestamp,
   time_of_exit timestamp,
   primary key(visitorID,flatID),
   foreign key(flatID) references flat(FlatID)
);

-- Creating complaints table
create table complaints
(
   ComplaintID varchar(10) primary key,
   ResidentUID varchar(14),
   Complain char(100),
   date date,time time,
   foreign key(ResidentUID) references resident(Aadhar)
);


-- Creating flat_has_security table
create table flat_has_security
(
   FlatID varchar(10),
```

```
   securityID varchar(10),
   primary key(FlatID,securityID),
   foreign key(FlatID) references flat(FlatID),
   foreign key(securityID) references security(securityID)
   );

-- Creating resident_avails_services table
create table resident_avails_services
(
   ResidentUID varchar(14),
   serviceID varchar(10),
   serviceTime timestamp,
   primary key(ResidentUID,serviceID,serviceTime),
   foreign key(ResidentUID) references resident(Aadhar)
   );

create table auth
(
   username varchar(12) not null,
   password varchar(20) not null,
   type varchar(10) not null,
   Primary key (username)
);
```

Populating the database:

```
insert into resident
values('763676231429','Sahana','Ramesh','2001-07-10',9837837628,'F','Dog','A1001','Own');
insert into resident (Aadhar,Fname,Lname,DOB,phone,gender,FlatID,rent_owned)
values('542155761651','Ritesh','Kumar','2001-04-03',9987452411,'M','C202','Own');
insert into resident
values('109382652765','Sachin','Shenoy','2001-06-16',9848367537,'M','Cat','A103','Rent');
insert into resident (Aadhar,Fname,Lname,DOB,phone,gender,FlatID,rent_owned)
values('265342837265','Vaibhav','Suresh','2001-07-16',8723786821,'M','A208','Rent');
insert into resident (Aadhar,Fname,Lname,DOB,phone,gender,FlatID,rent_owned)
values('782628762672','Shreyas','Devraj','2001-01-05',8362627761,'M','B304','Own');

insert into flat values('A1001',2);
insert into flat values('C202',3);
insert into flat values('A103',2);
insert into flat values('A208',2);
insert into flat values('B304',3);

insert into parking_slot values(100,'E-Scooter','A1001');
insert into parking_slot values(322,'SUV','C202');
insert into parking_slot values(103,'Cycle','A103');
insert into parking_slot values(128,'Sedan','A208');
insert into parking_slot values(234,'SUV','B304');


insert into security values('5626451561','Bahadur','J','6366536763','2018-10-01','Day');
insert into security values('4352761521','Praveen','Singh','3675372582','2015-01-04','Day');
insert into security values('9827632415','Ravi','Kumar','9847638375','2016-11-01','Night');
insert into security values('2435142351','Sharma','Ji','7375167251','2014-06-01','Night');
insert into security values('5526517571','Nayak','A','7636453771','2017-03-05','Day');

insert into dependent
values('864235907950','763676231429','Ikya','Beria','2001-08-10','8736278563','F');
```

```
insert into dependent
values('168221720046','763676231429','Sahan','Shetty','2010-04-01','7465736245','M');
insert into dependent
values('168221720036','763676231429','Shreya','Shetty','1950-04-10','7465736245','F');
insert into dependent
values('223641701075','763676231429','Raksha','Hemmige','2001-03-11','6237161571','F');
insert into dependent
values('699137916788','542155761651','Ramesh','G','2001-02-08','8762542161','M');
insert into dependent
values('653227375868','109382652765','Suhas','K','2001-11-24','9863534643','M');
insert into dependent
values('784134707754','109382652765','Mehul','Mehta','2004-05-03','9123534643','M');
insert into dependent
values('591343435901','265342837265','Raghu','NV','2001-10-01','2753625711','M');
insert into dependent
values('862278327098','782628762672','Sanjana','Atrey','2001-11-01','8726353171','M');
insert into dependent
values('862278327099','782628762672','Sanjay','Atrey','1940-11-01','8726353171','M');

insert into visitor (visitorID,flatID,Fname,Lname,purpose,phonenumber,time_of_entry)
values('4146987178','A103','Raghu','NV','Personal','8926781618','2020-10-19 12:30:01');
insert into visitor (visitorID,flatID,Fname,Lname,purpose,phonenumber,time_of_entry)
values('4146987120','A1001','Nihal','Shetty','Visit','8926781623','2021-11-19 11:36:01');
insert into visitor (visitorID,flatID,Fname,Lname,purpose,phonenumber,time_of_entry)
values('4146987121','A1001','Neha','Shetty','Visit','8926121623','2021-11-19 12:36:01');
insert into visitor (visitorID,flatID,Fname,Lname,purpose,phonenumber,time_of_entry)
values('4142887122','A1001','Roopak','M','Inspection','8926671623','2021-11-23 09:18:19');
insert into visitor (visitorID,flatID,Fname,Lname,purpose,phonenumber,time_of_entry)
values('4146981226','B304','Nihal','Shetty','Visit','8926781623','2021-11-21 10:32:01');
insert into visitor (visitorID,flatID,Fname,Lname,purpose,phonenumber,time_of_entry)
values('4145687127','B304','Dhruv','Shetty','Professional','823726531','2021-11-18 09:01:12');
insert into visitor (visitorID,flatID,Fname,Lname,purpose,phonenumber,time_of_entry)
values('4146127128','C202','Dhanush','Kumar','Party','8237231531','2021-11-18 21:11:12');
insert into visitor (visitorID,flatID,Fname,Lname,purpose,phonenumber,time_of_entry)
values('4146677129','C202','Iqrar','Ahmed','Sports','8224231531','2021-11-18 16:16:12');

insert into complaints values('2785367152','763676231429','Water leakage from
faucet','2021-10-01','18:01:10');
insert into complaints values('2785367151','542155761651','Plug points not
working','2021-11-01','15:01:10');
insert into complaints values('2785367131','782628762672','Internet
down','2021-11-01','06:11:21');
insert into complaints values('2785227131','782628762672','Street
Light','2021-11-03','22:19:21');

insert into services values('1290892081',200.00,'Plumbing');
insert into services values('1290892082',500.00,'Electrician');
insert into services values('1290892083',600.00,'Internet');
insert into services values('1290892084',200.00,'Car Wash');

insert into employee
values('7283681618','Ramesh','Appa','8767381681','2017-01-04','M','Day','1290892084',16000);
insert into employee
values('7283681611','Saral','V','8767381688','2010-01-04','M','Night','1290892084',18000);
insert into employee
values('7283681619','Suresh','Appa','8767381611','2015-11-14','M','Night','1290892083',25000);
insert into employee
values('7283681629','Gangu','Thai','8767381621','2015-07-14','F','Day','1290892083',15700);
insert into employee
values('7283681281','Kanta','Bai','8767387652','2015-08-18','F','Day','1290892081',19000);
insert into employee
values('7283681612','Ganesha','K','8767381612','2016-11-14','M','Day','1290892082',19500);

insert into flat_has_security values('A1001','4352761521');
insert into flat_has_security values('A103','5526517571');
```

```
insert into flat_has_security values('C202','9827632415');
insert into flat_has_security values('A208','2435142351');
insert into flat_has_security values('B304','5526517571');

insert into resident_avails_services (ResidentUID,serviceID) values('763676231429','1290892081');
insert into resident_avails_services (ResidentUID,serviceID) values('542155761651','1290892082');
insert into resident_avails_services (ResidentUID,serviceID) values('782628762672','1290892083');
insert into resident_avails_services (ResidentUID,serviceID) values('265342837265','1290892083');
```

Join Queries:

Display firstname,lastname,shift,gender,type of service and cost of the service given by an employee

select Fname,Lname,shift,gender,type,cost from employee as e INNER JOIN services as s ON e.serviceID = s.serviceID;

Fname	Lname	shift	gender	type	cost
Kanta	Bai	Day	F	Plumbing	200
Saral	V	Night	M	Car Wash	200
Ganesha	K	Day	M	Electrician	500
Ramesh	Appa	Day	M	Car Wash	200
Suresh	Appa	Night	M	Internet	600
Gangu	Thai	Day	F	Internet	600

Display all resident details and their visitor details(NULL if no visitor)

select r.Aadhar,r.Fname,r.Lname,r.FlatID,v.visitorID,v.Fname,v.Lname,v.purpose from resident as r LEFT OUTER JOIN visitor as v on r.FlatID=v.flatID;

Aadhar	Fname	Lname	FlatID	visitorID	Fname	Lname	purpose
109382652765	Sachin	Shenoy	A103	4146987178	Raghu	NV	Personal
265342837265	Vaibhav	Suresh	A208	*NULL*	*NULL*	*NULL*	*NULL*
542155761651	Ritesh	Kumar	C202	4146127128	Dhanush	Kumar	Party
542155761651	Ritesh	Kumar	C202	4146677129	Iqrar	Ahmed	Sports
763676231429	Sahana	Ramesh	A1001	4142887122	Roopak	M	Inspection
763676231429	Sahana	Ramesh	A1001	4146987120	Nihal	Shetty	Visit
763676231429	Sahana	Ramesh	A1001	4146987121	Neha	Shetty	Visit
782628762672	Shreyas	Devraj	B304	4145687127	Dhruv	Shetty	Professional
782628762672	Shreyas	Devraj	B304	4146981226	Nihal	Shetty	Visit

Display all security guard and their details along with the flat which they are assigned with

select s.securityID,Fname,Lname,phone,doj,shift,FlatID from security as s join flat_has_security as fhs on s.securityID=fhs.securityID ;

securityID	Fname	Lname	phone	doj	shift	FlatID
4352761521	Praveen	Singh	3675372582	2015-01-04	Day	A1001
5526517571	Nayak	A	7636453771	2017-03-05	Day	A103
2435142351	Sharma	Ji	7375167251	2014-06-01	Night	A208
5526517571	Nayak	A	7636453771	2017-03-05	Day	B304
9827632415	Ravi	Kumar	9847638375	2016-11-01	Night	C202

Display all resident details and their dependents details(NULL if no dependent)

select r.Aadhar,r.Fname,r.Lname,r.FlatID,d.Aadhar,d.Fname,d.Lname,d.gender from resident as r LEFT OUTER JOIN dependent as d on r.Aadhar=d.residentUID;

Aadhar	Fname	Lname	FlatID	Aadhar	Fname	Lname	gender
109382652765	Sachin	Shenoy	A103	653227375868	Suhas	K	M
109382652765	Sachin	Shenoy	A103	784134707754	Mehul	Mehta	M
265342837265	Vaibhav	Suresh	A208	591343435901	Raghu	NV	M
542155761651	Ritesh	Kumar	C202	699137916788	Ramesh	G	M
763676231429	Sahana	Ramesh	A1001	168221720036	Shreya	Shetty	F
763676231429	Sahana	Ramesh	A1001	168221720046	Sahan	Shetty	M
763676231429	Sahana	Ramesh	A1001	223641701075	Raksha	Hemmige	F
763676231429	Sahana	Ramesh	A1001	864235907950	Ikya	Beria	F
782628762672	Shreyas	Devraj	B304	862278327098	Sanjana	Atrey	M
782628762672	Shreyas	Devraj	B304	862278327099	Sanjay	Atrey	M

Aggregate Functions:

Display resident's first name,last name and number of complaints made so far:

select r.Fname,r.Lname,(select count(*) from complaints where ResidentUID=r.Aadhar) as no_of_complaints from resident as r;

Fname	Lname	no_of_complaints
Sachin	Shenoy	0
Vaibhav	Suresh	0
Ritesh	Kumar	1
Sahana	Ramesh	1
Shreyas	Devraj	2

Select the max salary of an employee for every service :

select type as service_name,max(salary) as max_salary from services as s, employee as e WHERE s.serviceID=e.serviceID GROUP BY s.serviceID;

service_name	max_salary
Plumbing	19000
Electrician	19500
Internet	25000
Car Wash	18000

Show average salary of all employees:

SELECT AVG(salary) as average_salary FROM employee;

average_salary
18866.666666666668

Count number of dependents for each resident:

select r.Aadhar,r.Fname,r.Lname, count(*) as dependents_count from resident as r LEFT OUTER JOIN dependent as d on r.Aadhar=d.residentUID group by r.Aadhar;

Aadhar	Fname	Lname	dependents_count
109382652765	Sachin	Shenoy	2
265342837265	Vaibhav	Suresh	1
542155761651	Ritesh	Kumar	1
763676231429	Sahana	Ramesh	4
782628762672	Shreyas	Devraj	2

Set Operations:

Select services which cost less than 500 or are provided in the night:

select serviceID from services where cost<500 UNION select serviceID from employee where shift='night';

serviceID
1290892081
1290892084
1290892083

Select securityID and their name who guard Flat's in A block and are on night shift:

select securityID,Fname,Lname from security where shift='Night' INTERSECT (select s.securityID,Fname,Lname from security as s, flat_has_security as fas WHERE fas.securityID = s.securityID and fas.FlatID LIKE 'A%');

securityID	Fname	Lname
2435142351	Sharma	Ji

Select flat id with no parking slot:

select FlatID from flat
EXCEPT
(select flatID from parking_slot);

FlatID
D101

Select resident ids who have taken services and given complaints:

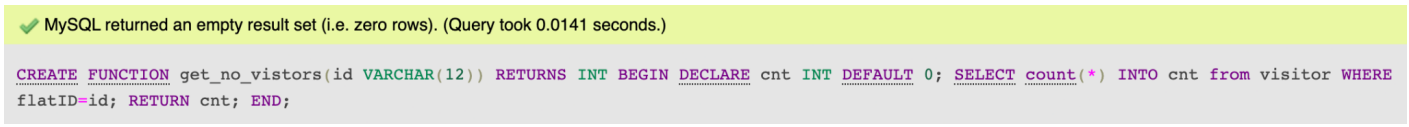
```
select ResidentUID from complaints
INTERSECT
select ResidentUID from resident_avails_services;
```

ResidentUID
542155761651
763676231429
782628762672

Functions and Procedures:

Function to get total number of visitors for a given flat id

```
DELIMITER $$
CREATE FUNCTION get_no_vistors(id VARCHAR(12)) RETURNS INT
BEGIN
DECLARE cnt INT DEFAULT 0;
SELECT count(*) INTO cnt from visitor WHERE flatID=id;
RETURN cnt;
END $$
```

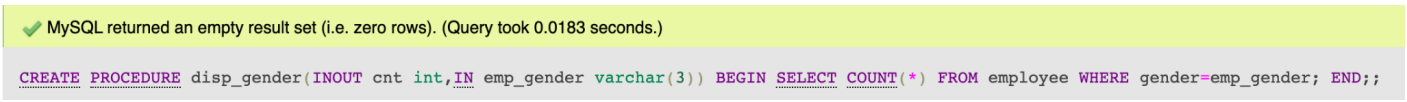


SELECT FlatID,get_no_vistors(FlatID) as num_visitors FROM flat;

FlatID	num_visitors
A1001	3
A103	1
A208	0
B304	2
C202	2
D101	0

Procedure to return the count of the employees entered gender

```
DELIMITER $$
CREATE PROCEDURE disp_gender(INOUT cnt int,IN emp_gender varchar(3))
BEGIN
SELECT COUNT(*) FROM employee WHERE gender=emp_gender;
END; $$
```




```
call disp_gender(@M,"M");
SELECT @M;
```

COUNT(*)
4

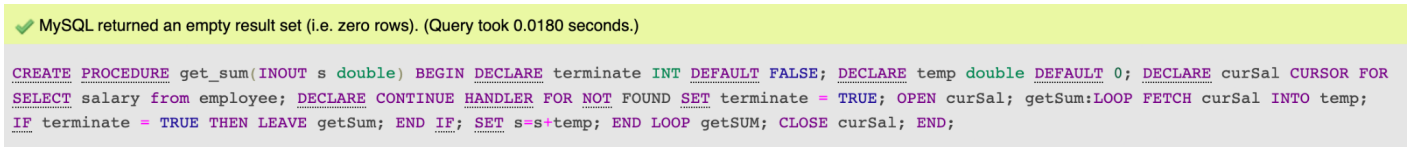
```
call disp_gender(@F,"F");
SELECT @F;
```

COUNT(*)
2

Triggers and Cursors:

Cursor based procedure to get sum of salaries of all employees

```
DELIMITER $$
CREATE PROCEDURE get_sum(INOUT s double)
BEGIN
DECLARE terminate INT DEFAULT FALSE;
DECLARE temp double DEFAULT 0;
DECLARE curSal CURSOR FOR SELECT salary from employee;
DECLARE CONTINUE HANDLER FOR NOT FOUND SET terminate = TRUE;
OPEN curSal;
getSum:LOOP
      FETCH curSal INTO temp;
      IF terminate = TRUE THEN
         LEAVE getSum;
      END IF;
      SET s=s+temp;
END LOOP getSUM;
CLOSE curSal;
END $$
```




```
SET @s=0;
CALL get_sum(@s);
SELECT @s;
```

@s
113200

Trigger to check if date of joining of employee being added is proper or not(date of joining cannot be any future date)

```
DELIMITER $$
CREATE TRIGGER doj_check BEFORE INSERT ON employee
FOR EACH ROW
BEGIN
IF DATEDIFF(curdate(),NEW.doj)<0 THEN SIGNAL SQLSTATE '50001' SET
MESSAGE_TEXT="Date of joining cannot be in future";
END IF;
END; $$
```

MySQL returned an empty result set (i.e. zero rows). (Query took 0.0125 seconds.)

```
CREATE TRIGGER doj_check BEFORE INSERT ON employee FOR EACH ROW BEGIN IF DATEDIFF(curdate(),NEW.doj)<0 THEN SIGNAL SQLSTATE '50001' SET
MESSAGE_TEXT="Date of joining cannot be in future"; END IF; END;;
```

```
INSERT INTO employee
values("7457674555","Raju","Mittal","9786775757","2022-12-01","M","Day","1290892081
",18000);
```

Error

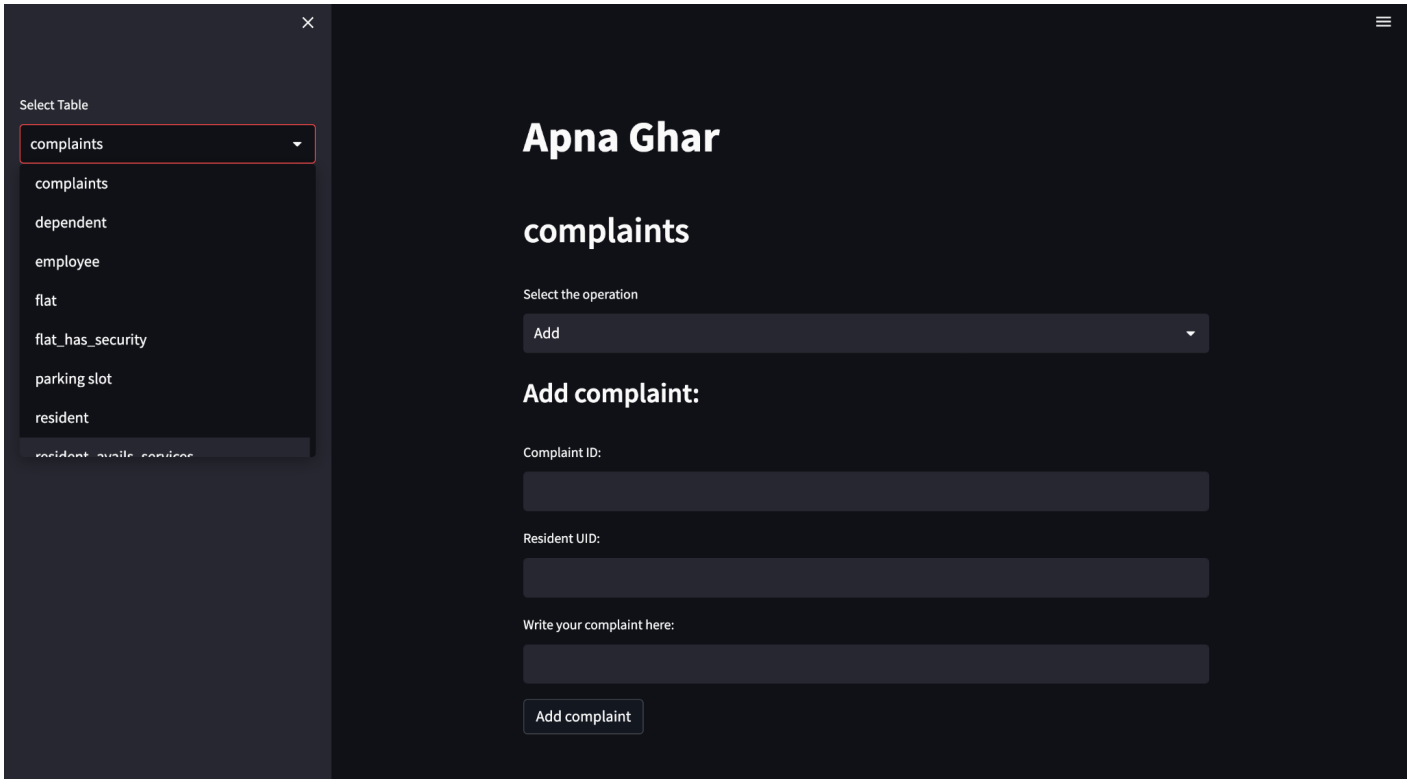
SQL query: Copy

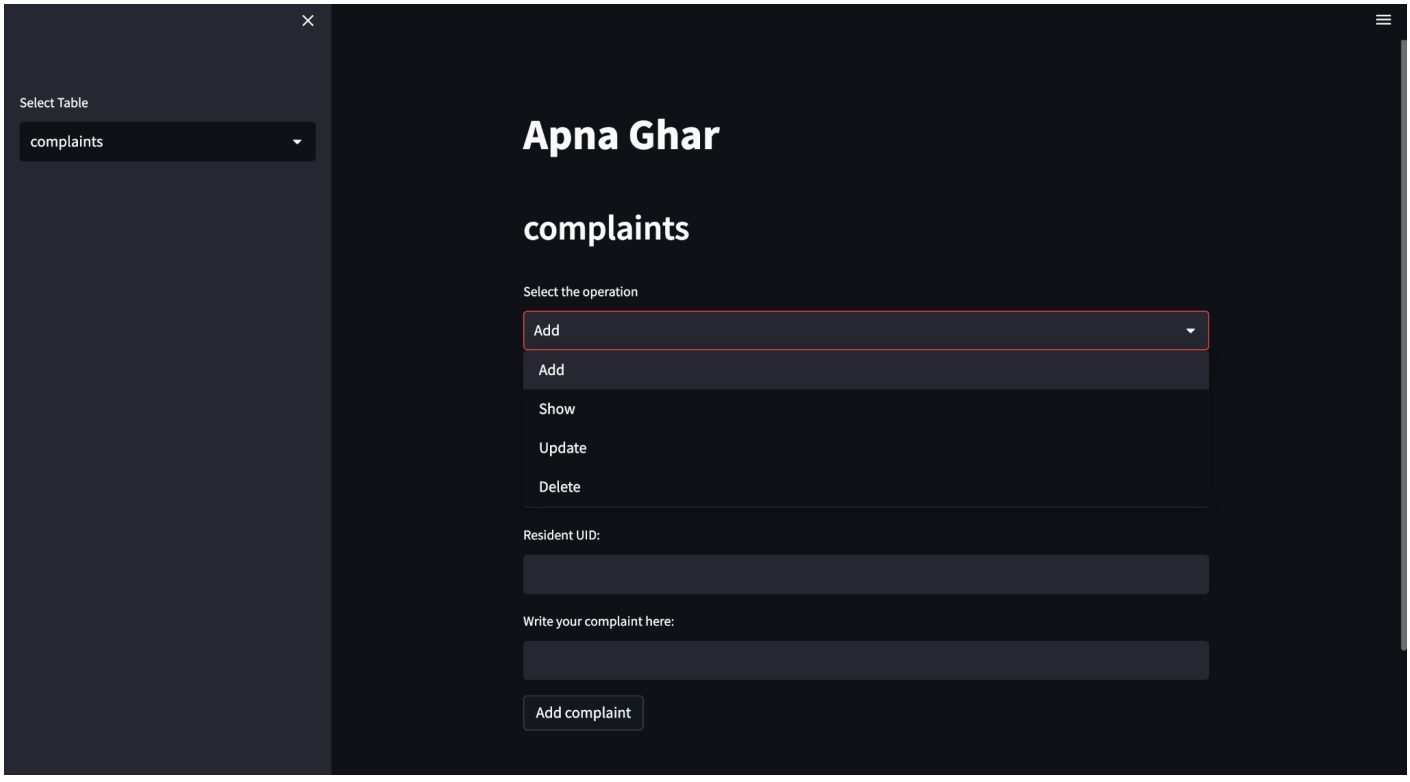
```
INSERT INTO employee values("7457674555","Raju","Mittal","9786775757","2022-12-01","M","Day","1290892081",18000);
```

MySQL said:

```
#1644 - Date of joining cannot be in future
```

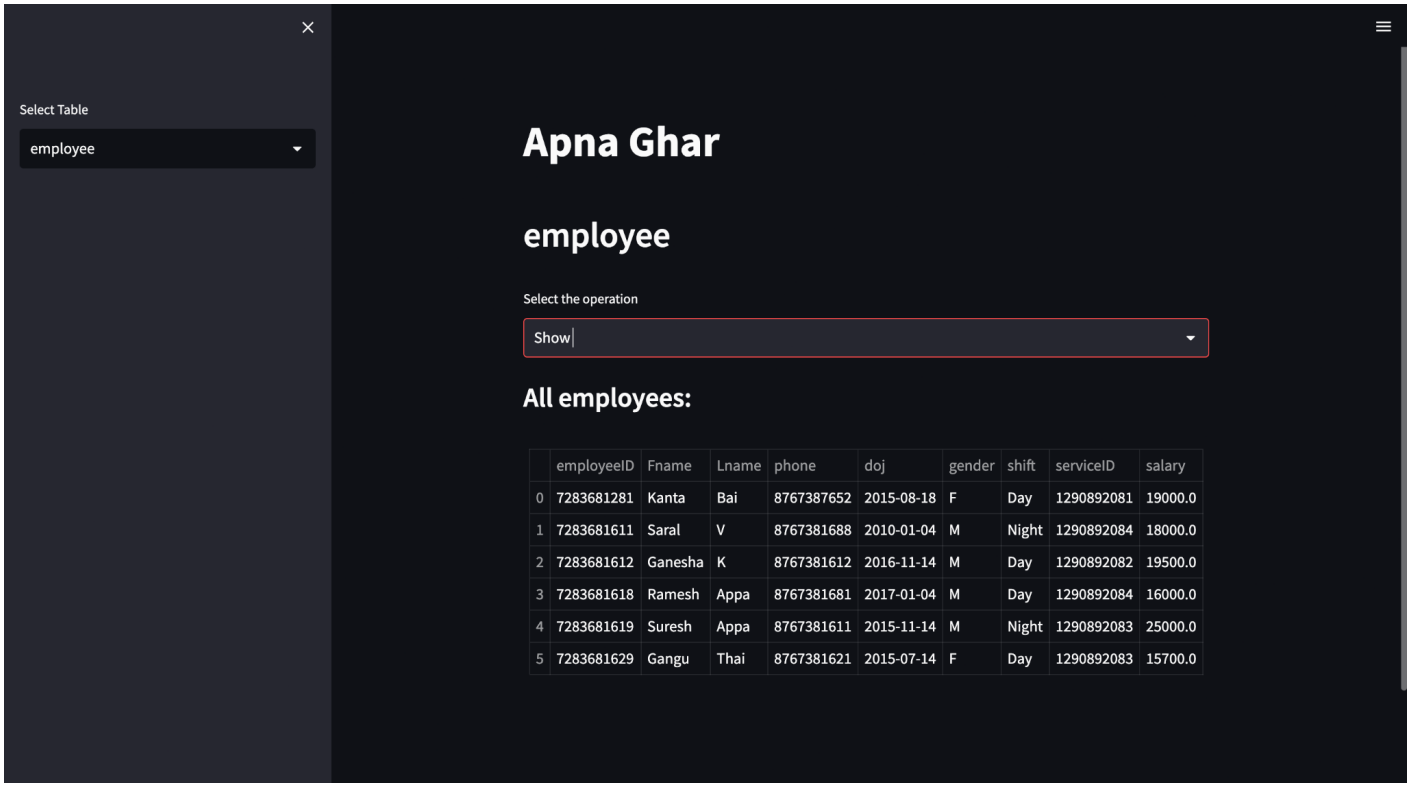
Frontend:


Select Table
complaints
complaints
dependent
employee
flat
flat_has_security
parking slot
resident
Apna Ghar
complaints
Select the operation
Add
Add complaint:
Complaint ID:
Resident UID:
Write your complaint here:
Add complaint


Select Table
complaints
Apna Ghar
complaints
Select the operation
Add
Add
Show
Update
Delete
Resident UID:
Write your complaint here:
Add complaint

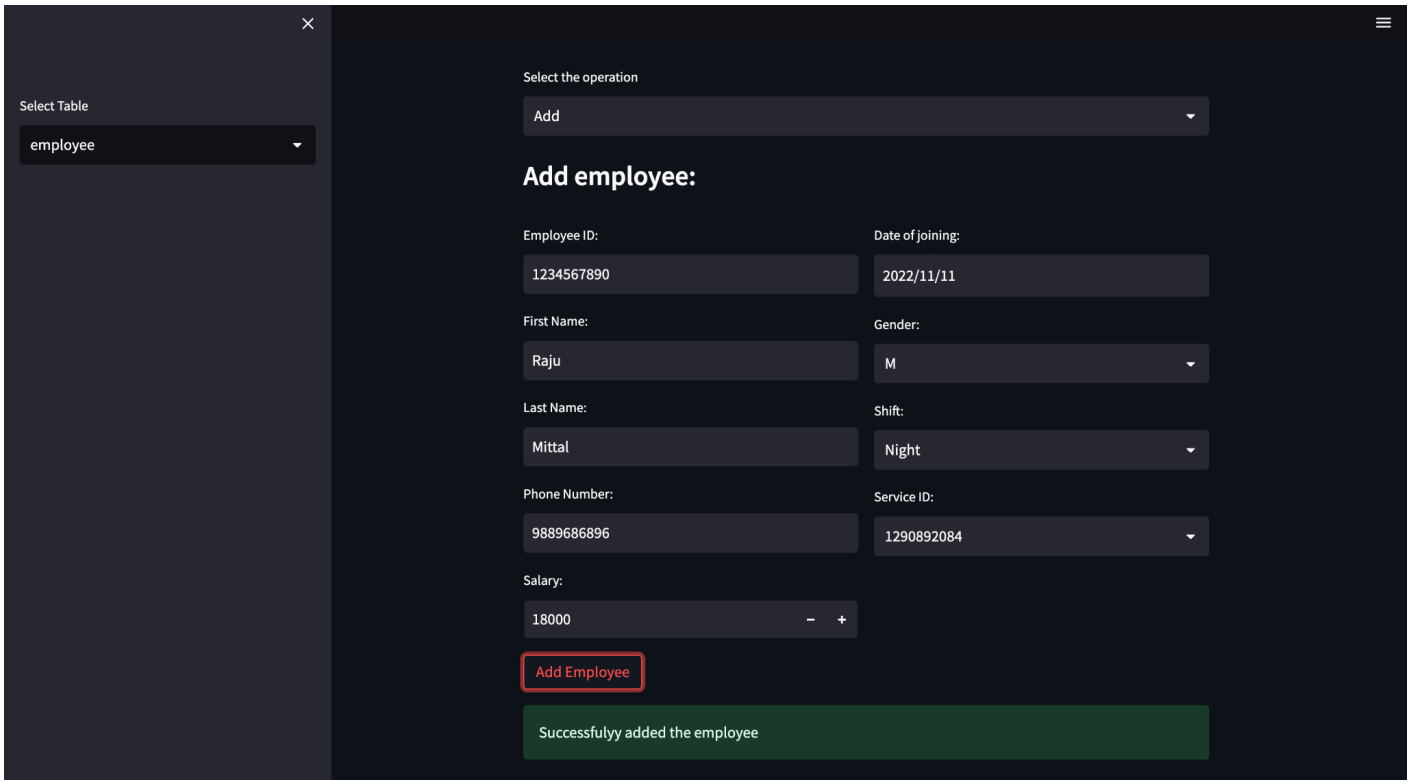
CRUD operation demo on one table from frontend

Initial values:

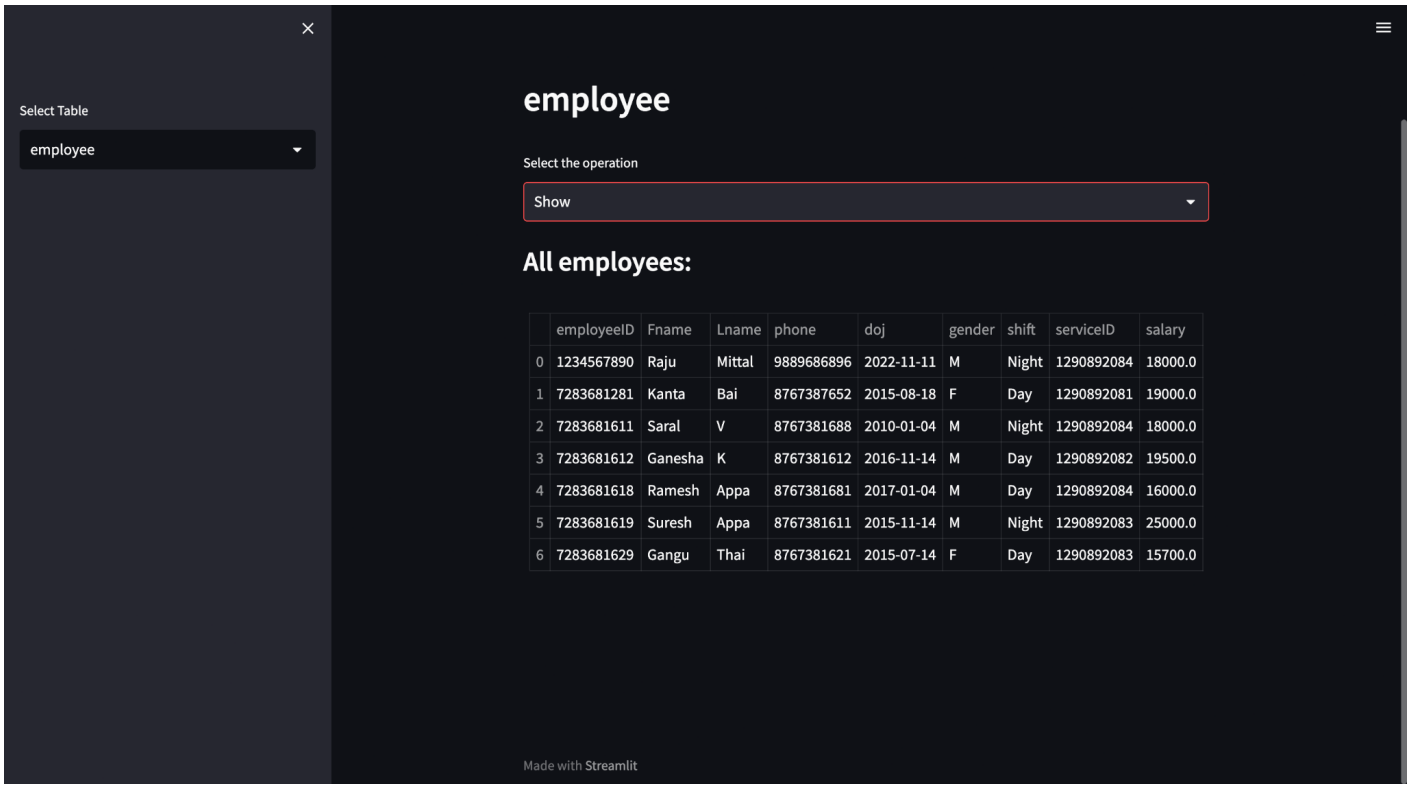


	employeeID	Fname	Lname	phone	doj	gender	shift	serviceID	salary
0	7283681281	Kanta	Bai	8767387652	2015-08-18	F	Day	1290892081	19000.0
1	7283681611	Saral	V	8767381688	2010-01-04	M	Night	1290892084	18000.0
2	7283681612	Ganesha	K	8767381612	2016-11-14	M	Day	1290892082	19500.0
3	7283681618	Ramesh	Appa	8767381681	2017-01-04	M	Day	1290892084	16000.0
4	7283681619	Suresh	Appa	8767381611	2015-11-14	M	Night	1290892083	25000.0
5	7283681629	Gangu	Thai	8767381621	2015-07-14	F	Day	1290892083	15700.0

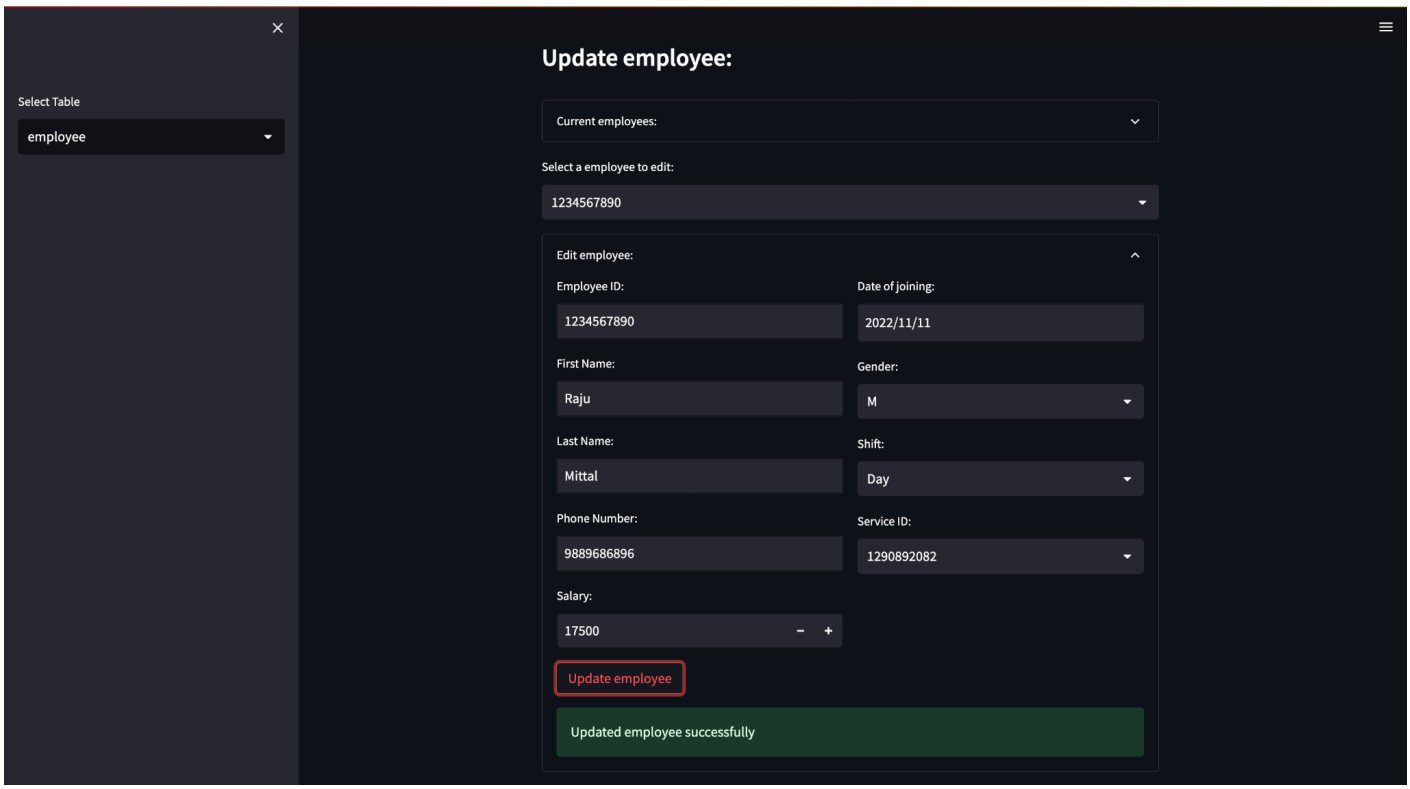
Adding a tuple:



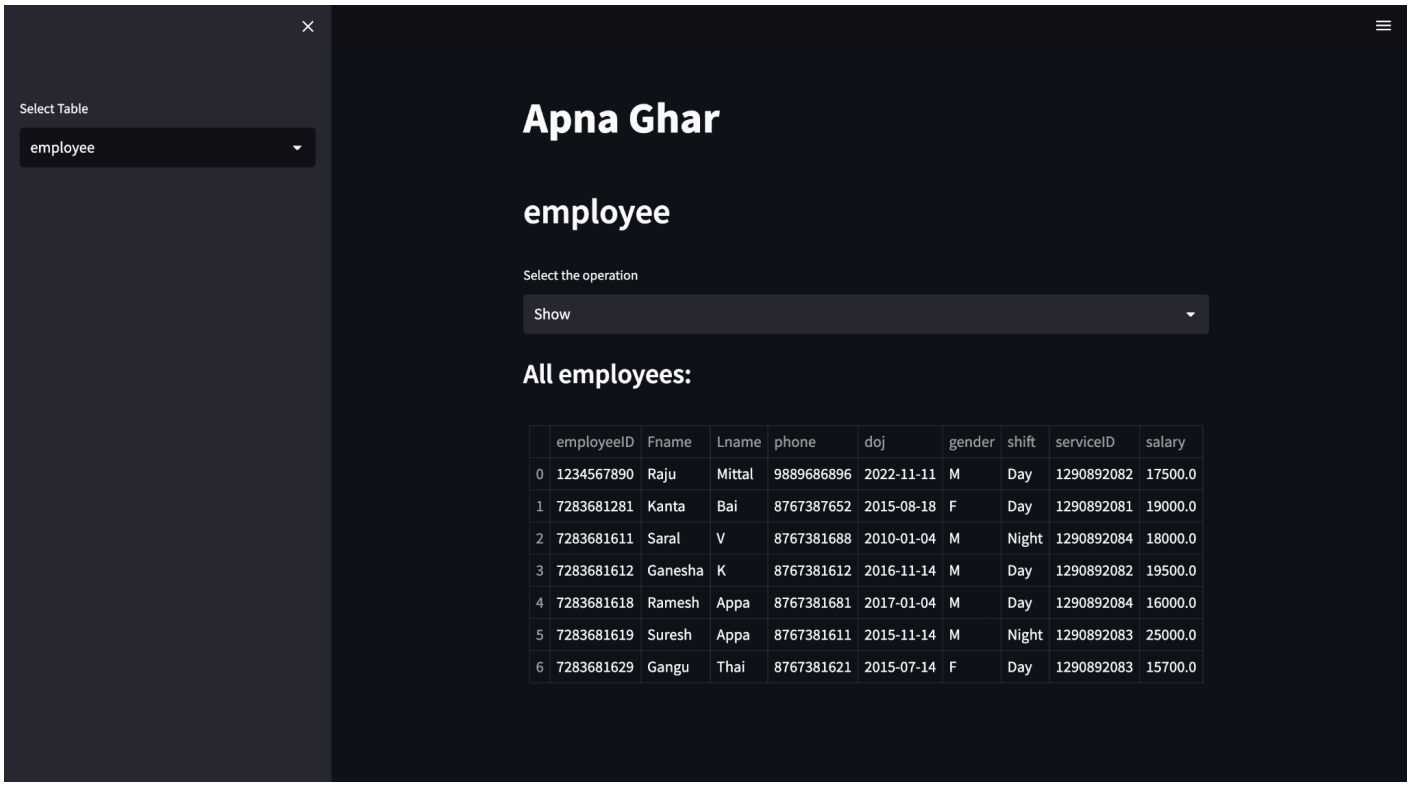
Values after addition:



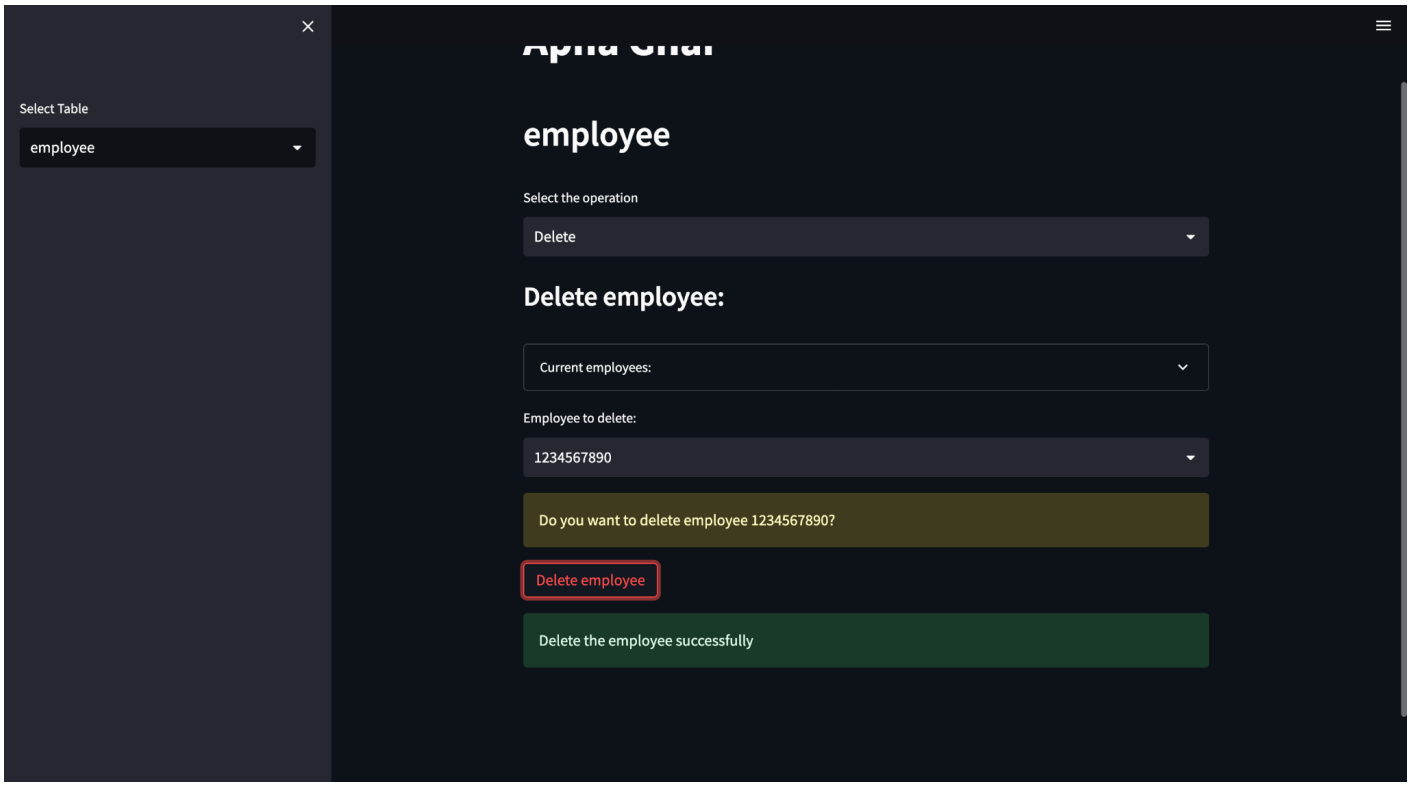
Updating the details of employee:



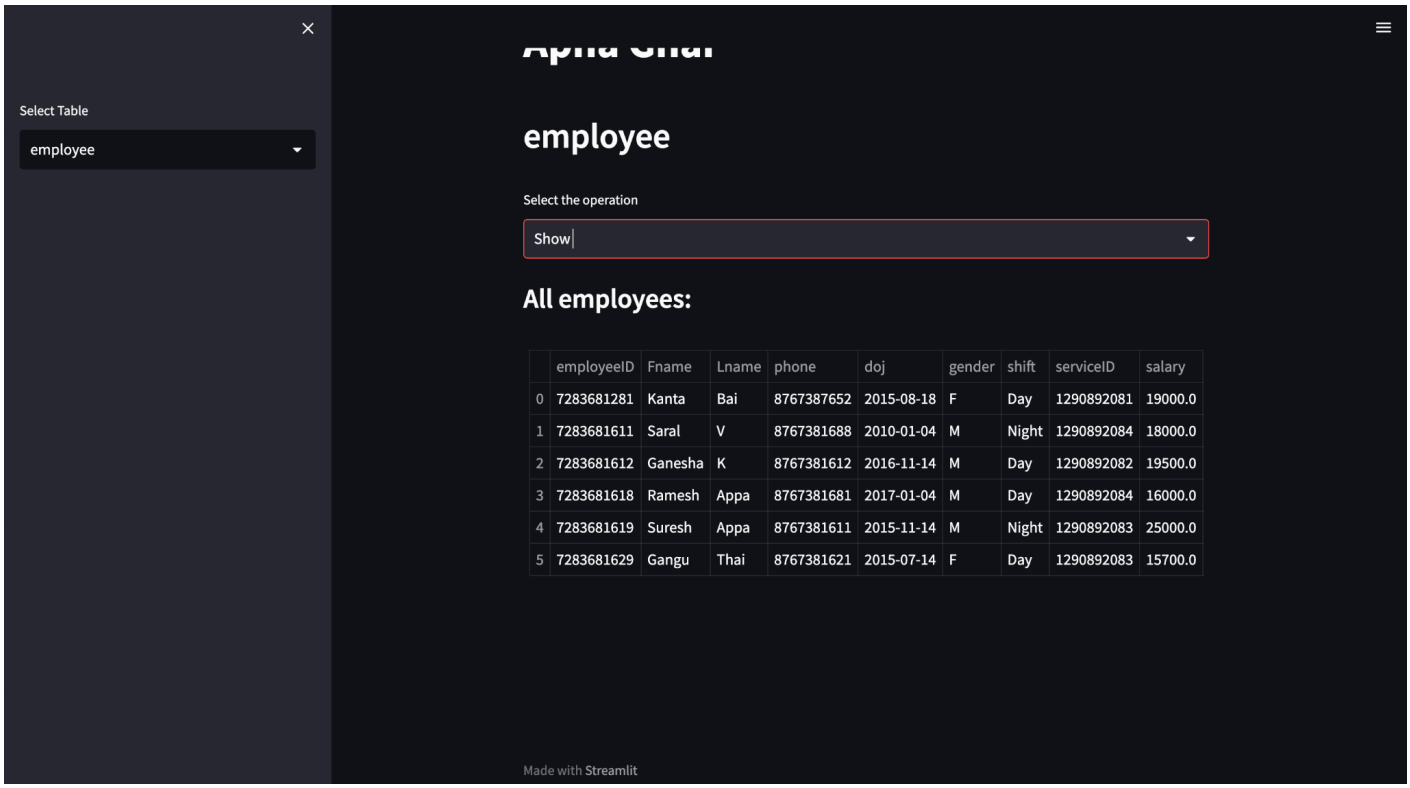
Values after updating:



Deleting a employee:



Values after deleting:



	employeeID	Fname	Lname	phone	doj	gender	shift	serviceID	salary
0	7283681281	Kanta	Bai	8767387652	2015-08-18	F	Day	1290892081	19000.0
1	7283681611	Saral	V	8767381688	2010-01-04	M	Night	1290892084	18000.0
2	7283681612	Ganesha	K	8767381612	2016-11-14	M	Day	1290892082	19500.0
3	7283681618	Ramesh	Appa	8767381681	2017-01-04	M	Day	1290892084	16000.0
4	7283681619	Suresh	Appa	8767381611	2015-11-14	M	Night	1290892083	25000.0
5	7283681629	Gangu	Thai	8767381621	2015-07-14	F	Day	1290892083	15700.0