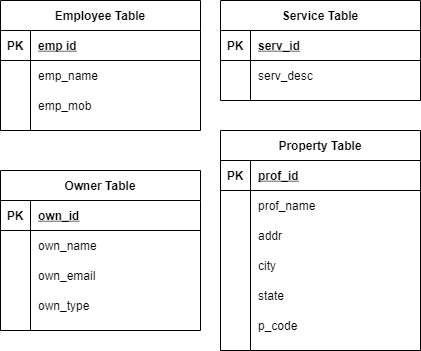
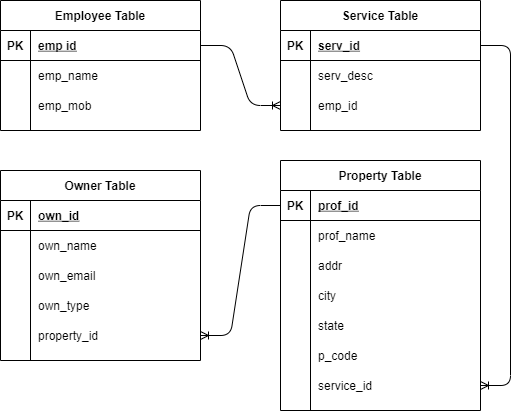
**a. Using the table and data in Figure 1, state assumptions about dependencies among the columns of the table. Justify your assumptions on the basis of the sample data and also on the basis of what you know about service business.**

in the figure 1 Property Column is totally depend upon owners’ column, service of each property is depend upon the type of service column and whom is providing by the employee column. So will separate and create Property, Owner, Service and Employees Tables. Below is my example



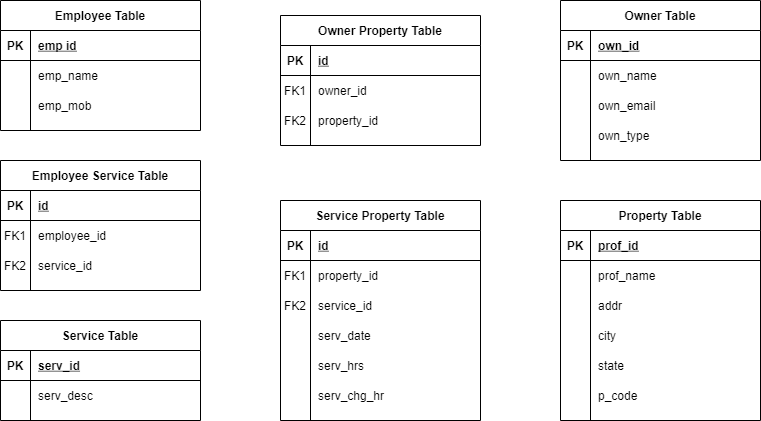
**b. Employing the dependencies stated in in (a), write a relational schema and draw a dependency diagram for the table in 1NF. The dependency diagram must have proper labels for all functional, partial and/or transitive dependencies, if there are any.**

Figure 1 is in already in 1NF because there are no multi-valued attributes. Following is the relational schema for the table:

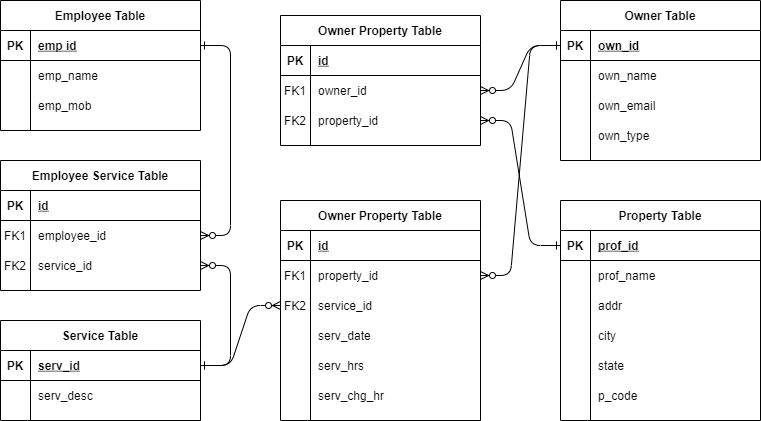


**c. Break up the dependency diagram you drew in (b) to produce dependency diagrams that are in 3 NF and also write the relational schemas for the table in 3NF. Make sure the new dependency diagrams contain attributes that meet proper design criteria; that is, make sure that there are no multivalued attributes, that the naming conventions are met, and so on.)**

From above diagram, I have created three more tables: Employee Service Table, Owner Property Table, Service Property Table. In above table, there were chances of redundancy to occur. By separating dependency tables furthermore, schema is in 3NF now, and chances of redundancy are quite less.



**d. Develop an E-R diagram based on the task done in (c). Use crow-foot style and specify entities, attributes, relationships, and multiplicity. Justify the decisions you make regarding minimum and maximum cardinality. Describe how you would go about validating this model.**



**e. Using the E-R diagram you developed in(d), convert it to a relational design. Document your design as follows:**

* **Specify tables, primary keys, and foreign keys.**

|  |  |  |  |
| --- | --- | --- | --- |
| Sro No | Table Name | Primary Key | Foreign Key |
| 1 | employee\_table | emp\_id | - |
| 2 | service\_table | serv\_id | - |
| 3 | employee\_service\_table | id | employee\_id, service\_id |
| 4 | owner\_table | own\_id | - |
| 5 | propety\_table | prof\_id | - |
| 6 | owner\_property\_table | Id | owner\_id, property\_id |
| 7 | service\_propert\_table | Id | property\_id, service\_id |

* **Describe how you have represented weak entities, if there are any.**
  1. There is no week entities in all the tables. A weak entity is a type of entity which doesn’t have its key attribute. It can be identified uniquely by considering the primary key of another entity. For that, weak entity sets need to have participation.
* Document relationship enforcement.

**f. Create a database using MySQL, with primary keys, foreign keys, and other attributes mentioned for each table developed in (e) using proper constraints.**

CREATE Database database\_project;

USE database\_project;

CREATE TABLE `employee\_table` (

`emp\_id` int(11) NOT NULL PRIMARY KEY,

`emp\_name` varchar(150) DEFAULT NULL,

`emp\_mob` varchar(15) DEFAULT NULL

)ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `service\_table` (

`serv\_id` int(11) NOT NULL PRIMARY KEY,

`serv\_desc` varchar(255) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `employee\_service\_table` (

`id` int(11) NOT NULL PRIMARY KEY,

`employee\_id` int(11) DEFAULT NULL,

`service\_id` int(11) DEFAULT NULL,

CONSTRAINT `fk\_employee\_id` FOREIGN KEY (`employee\_id`) REFERENCES `employee\_table` (`emp\_id`),

CONSTRAINT `fk\_service\_id\_` FOREIGN KEY (`service\_id`) REFERENCES `service\_table` (`serv\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `owner\_table` (

`own\_id` int(11) NOT NULL PRIMARY KEY,

`own\_name` varchar(150) DEFAULT NULL,

`own\_email` varchar(150) DEFAULT NULL,

`own\_type` varchar(150) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `propety\_table` (

`prof\_id` int(11) NOT NULL PRIMARY KEY,

`prof\_name` varchar(150) DEFAULT NULL,

`addr` varchar(150) DEFAULT NULL,

`city` varchar(150) DEFAULT NULL,

`state` varchar(150) DEFAULT NULL,

`p\_code` varchar(150) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `owner\_property\_table` (

`id` int(11) NOT NULL PRIMARY KEY,

`owner\_id` int(11) DEFAULT NULL,

`property\_id` int(11) DEFAULT NULL,

CONSTRAINT `fk\_owner\_id` FOREIGN KEY (`owner\_id`) REFERENCES `owner\_table` (`own\_id`),

CONSTRAINT `fk\_property\_id` FOREIGN KEY (`property\_id`) REFERENCES `propety\_table` (`prof\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

CREATE TABLE `service\_propert\_table` (

`id` int(11) NOT NULL PRIMARY KEY,

`property\_id` int(11) NOT NULL,

`service\_id` int(11) NOT NULL,

`serv\_date` date DEFAULT NULL,

`serv\_hrs` float DEFAULT NULL,

`serv\_chg\_hr` float DEFAULT NULL,

CONSTRAINT `fk\_property\_id\_` FOREIGN KEY (`property\_id`) REFERENCES `propety\_table` (`prof\_id`),

CONSTRAINT `fk\_service\_id` FOREIGN KEY (`service\_id`) REFERENCES `service\_table` (`serv\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

**g. Create SQL statements to satisfy the following:**

**1. Write SQL statements to insert at least 7 rows of data into each of the table created in task (f). You may use the sample data provided in Figure 1 for this task**

INSERT INTO `employee\_table` (`emp\_id`, `emp\_name`, `emp\_mob`) VALUES

(101, 'Robert McGraw', '401234567'),

(102, 'Mike Burrows', '414563453'),

(103, 'Ben Keen', '414563453');

INSERT INTO `owner\_table` (`own\_id`, `own\_name`, `own\_email`, `own\_type`) VALUES

(1, 'Helmet Jones', 'helmet.jones@gmail.com', 'Individual'),

(2, 'Kenny Blackmore', 'Kenny.blackmore@gmail.com', 'Coroporation'),

(3, 'Barry Wilson', 'barry.wilson@gmail.com', 'Individual'),

(4, 'Craig Noon', 'carig.noon@gmail.com', 'Individual'),

(5, 'Wendy Sullivan', 'wendy.sullivan@gmail.com', 'Coroporation'),

(6, 'Jim Bruno', 'jim.gruno@brunoandosone.com', 'Individual'),

(7, 'Peter Emerson', 'peter.emerson@emersonflyingjet.com', 'Coroporation');

NSERT INTO `propety\_table` (`prof\_id`, `prof\_name`, `addr`, `city`, `state`, `p\_code`) VALUES

(10, 'Eastlake Building', '123 Eastlake', 'Maroona', 'VIC', '3210'),

(11, 'Earls Court', '235 East West', 'Portland', 'VIC', '3330'),

(12, 'Barry Wilson', '75 West Bound', 'Dundee', 'VIC', '3500'),

(13, 'Jack and Jill', '105 Young', 'Freshy', 'VIC', '3350'),

(14, 'Cosey Here', '144 Sensible', 'Sunshine', 'VIC', '3456'),

(15, 'Bruno & Son', '66/30 Palm Beach', 'Newland', 'VIC', '3333'),

(16, 'Emerson Flying Jet', '707 Ardunino', 'Mega', 'VIC', '3256');

INSERT INTO `service\_table` (`serv\_id`, `serv\_desc`) VALUES

(1, 'Garden Service'),

(2, 'Lawn Mow');

INSERT INTO `employee\_service\_table` (`id`, `employee\_id`, `service\_id`) VALUES

(1, 101, 1),

(2, 102, 2),

(3, 101, 1),

(4, 102, 2),

(5, 103, 1),

(6, 102, 2),

(7, 103, 2),

(8, 101, 2);

INSERT INTO `owner\_property\_table` (`id`, `owner\_id`, `property\_id`) VALUES

(1, 1, 10),

(2, 2, 11),

(3, 3, 12),

(4, 4, 13),

(5, 5, 14),

(6, 6, 15),

(7, 7, 16);

INSERT INTO `service\_propert\_table` (`id`, `property\_id`, `service\_id`, `serv\_date`, `serv\_hrs`, `serv\_chg\_hr`) VALUES

(1, 10, 1, '2020-05-05', 2.5, 75),

(2, 11, 2, '2020-05-07', 2, 55),

(3, 12, 1, '2020-05-07', 2.5, 85),

(4, 11, 2, '2020-05-12', 1.5, 50),

(5, 13, 1, '2020-05-19', 3, 85),

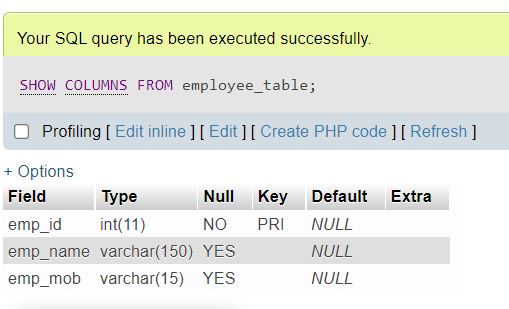
(6, 14, 2, '2020-05-17', 2, 90),

(7, 15, 2, '2020-05-14', 2, 55),

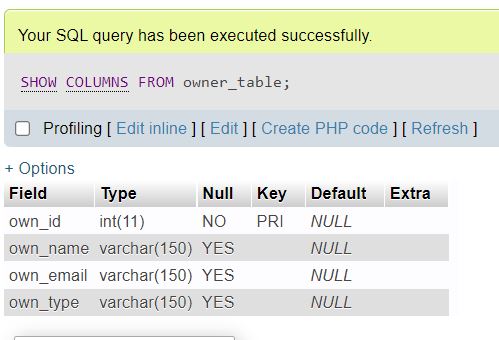
(8, 16, 2, '2020-05-10', 1, 50);

**2. Write SQL statements to list all columns of all tables**

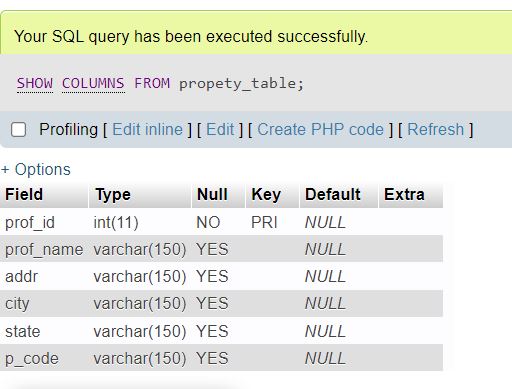
SHOW COLUMNS FROM employee\_table;



SHOW COLUMNS FROM owner\_table;



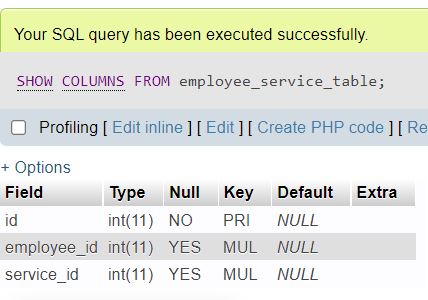
SHOW COLUMNS FROM propety\_table;



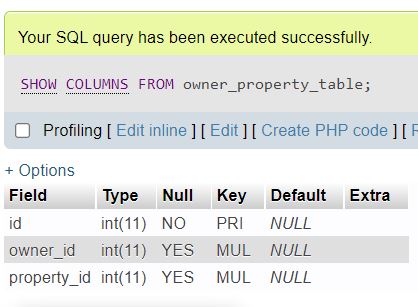
SHOW COLUMNS FROM service\_table;



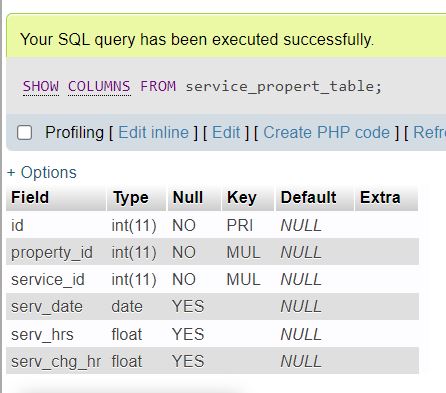
SHOW COLUMNS FROM employee\_service\_table;



SHOW COLUMNS FROM owner\_property\_table;



SHOW COLUMNS FROM service\_propert\_table;



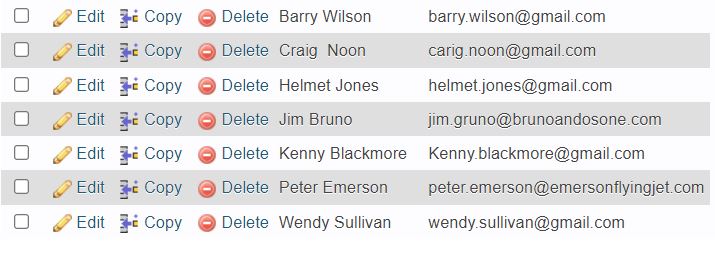
**3. Write SQL statements to list the name and mobile phone for all employees**

SELECT emp\_name,emp\_mob FROM employee\_table ORDER BY emp\_name



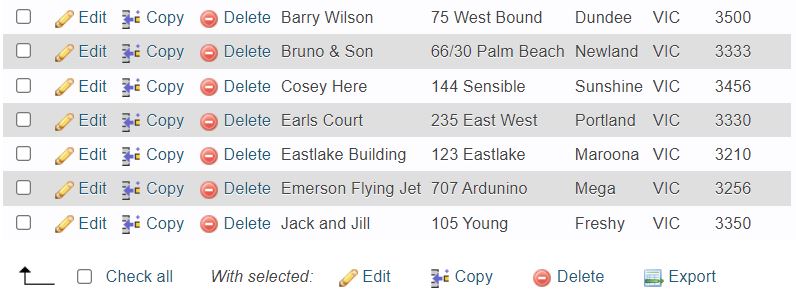
**4. Write SQL statements to list the name and email address for all owners.**

SELECT own\_name,own\_email FROM owner\_table ORDER BY own\_name ASC



**5. Write SQL statements to list the property name, address, state, and post code for all properties.**

SELECT prof\_name,addr,city,state,p\_code FROM propety\_table ORDER BY prof\_name ASC



**6. Write SQL statements to list all owner names and their property owned.**

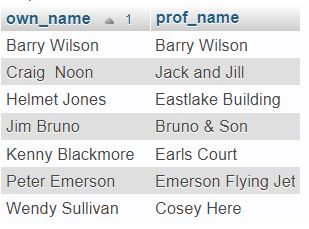
SELECT ot.own\_name,pt.prof\_name

FROM owner\_property\_table as opt

INNER JOIN owner\_table as ot ON opt.owner\_id=ot.own\_id

INNER JOIN propety\_table as pt ON opt.property\_id=pt.prof\_id

ORDER BY ot.own\_name ASC



**7. Write SQL statements to determine how many times of ‘Lawn Mow’ have been done at ‘Earls Courts’?**

SELECT

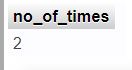
COUNT(\*) no\_of\_times

FROM service\_propert\_table as spt

INNER JOIN service\_table as st ON st.serv\_id=spt.service\_id

INNER JOIN propety\_table as pt ON pt.prof\_id=spt.property\_id

WHERE st.serv\_desc='Lawn Mow' and pt.prof\_name='Earls Court'



**8. Write SQL statements to list name of employees who have provided ‘Garden Service to a property owned by ‘Individual’.**

SELECT

e.emp\_name

FROM service\_propert\_table as spt

INNER JOIN service\_table as s ON s.serv\_id=spt.service\_id

INNER JOIN propety\_table as p ON p.prof\_id=spt.property\_id

INNER JOIN owner\_property\_table as opt ON opt.property\_id=spt.property\_id

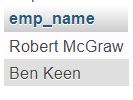
INNER JOIN owner\_table as o ON o.own\_id=opt.owner\_id

INNER JOIN employee\_service\_table as est ON est.service\_id=s.serv\_id

INNER JOIN employee\_table as e ON e.emp\_id=est.employee\_id

WHERE s.serv\_desc='Garden Service' and o.own\_type='Individual'

GROUP BY e.emp\_id



**9. Write SQL statements to list total service charge amounts for each property. Note that a service charge is calculated by multiplying the service hour by the service charged per hour.**

SELECT

p.prof\_name,(spt.serv\_hrs\*spt.serv\_chg\_hr) as service\_charges

FROM service\_propert\_table as spt

INNER JOIN propety\_table as p ON p.prof\_id=spt.property\_id

