Step 3: Practicing queries for the property database

1. List the names of all suburbs that have an average rate < 2.9

```
SELECT suburbname

FROM SUBURB

WHERE avgrate < 2.9;
```

```
select ____
from ____
where rate<2.9;
```

2. List the unit-code and date_of_installation of meters that have a meter-type which has a description of conventional

```
SELECT P.unitcode, P.date_of_installation

FROM PROPERTYMETER P

WHERE P.metertype IN (

  SELECT M.metertype

  FROM METERTYPE M

  WHERE M.description = "conventional"

);
```

```
Select unit-code, data_of_installation
from ____ where meter-type in (
  select metertype from ____
  wher description Like "conventional" );
```

3. List the pid and complete address (stno, stname, suburbname, postcode) of all properties that have the letter B or C in the name of the customer and have number of occupants > 3.

```
SELECT P.pid, P.stno, P.stname, S.suburbname, S.postcode

FROM PROPERTY P, SUBURB S, CUSTOMER C

WHERE P.suburbcode = S.suburbcode

AND P.TFN = C.TFN

AND (C.name LIKE "%B%" OR C.name LIKE "%C%")

 AND P.number_of_occupants > 3;
```

4. List the pid of all properties that do not have the metertype advanced installed. Use a sub-query in your answer.

```
SELECT pid

FROM PROPERTY

WHERE pid NOT IN (

    SELECT P.pid

    FROM PROPERTY P, PROPERTYMETER PM

    WHERE P.pid = PM.pid

    AND PM.metertype IN

    (SELECT MT.metertype
                    FROM METERTYPE MT

                   WHERE MT.description = "advanced" )

        );
```

5. Find the employees who have never done a meter reading. Display empid and empname

```
SELECT empid, ename

FROM METERREADER

WHERE empid NOT IN (

    SELECT DISTINCT empid

     FROM METERREADING

 );
```

6. List the suburbs (suburbcode) that have less than 3 properties with a metertype Advanced installed.

```
    SELECT S.suburbcode

    FROM SUBURB S, PROPERTY P

    WHERE S.suburbcode = P.suburbcode

    AND P.pid IN (

            SELECT PM.pid

            FROM METERTYPE M, PROPERTYMETER PM

            WHERE M.metertype = PM.metertype

            AND M.description = "advanced"

    )

    GROUP BY P.pid

    HAVING COUNT(*) < 3;
```

7. Find all properties that have an equal or greater no_of_occupants as pid 2.

```
SELECT P.pid

FROM PROPERTY P

WHERE P.number_of_occupants >=

    (SELECT number_of_occupants
        FROM PROPERTY

        WHERE pid = 2);
```

8. Find the customers whose home address (stno, stname, suburbcode) is different from any property they own. Display all details of the customer.

```
SELECT C.*

FROM PROPERTY P, CUSTOMER C

WHERE P.TFN = C.TFN

AND ( P.stno <> C.stno OR P.stname <> C.stname OR P.suburbcode <> C.suburbcode);
```

9. Find the suburb that has the lowest average rate. Display suburbcode.

```
SELECT suburbcode, avgrate

FROM SUBURB

WHERE avgrate = (SELECT Min(avgrate) FROM SUBURB);
```

10. Find the number of properties in suburbs that have an average rate > 2.7.

Solution I:

```
SELECT suburbcode, COUNT(*)

FROM PROPERTY

WHERE suburbcode IN (

    SELECT suburbcode

    FROM SUBURB

    WHERE avgrate >2.7)

GROUP BY suburbcode;
```

Solution II:

```
SELECT suburbname, COUNT(*)

FROM PROPERTY P, SUBURB B

WHERE B.suburbcode = P.suburbcode

AND avgrate > 2.7

GROUP BY B.suburbcode;
```

11. Find the number of readings done by each meter reader (empid) in 2009.

```
SELECT empid, COUNT(*)
```

```
FROM METERREADING

WHERE timestamp BETWEEN "2009-01-01" AND "2009-12-31"

GROUP BY empid;
```

12. Find all employees who have done a meter reading for the same property more than once.

```
SELECT ename

FROM METERREADER

WHERE empid IN

        (SELECT empid

        FROM METERREADING

        GROUP BY empid, pid

        HAVING COUNT(pid)>1)
```

13. How many apartments are there in the suburb of Barton.

```
SELECT COUNT(*)

FROM PROPERTY

WHERE propertytype = "apartment" AND suburbcode =

    (SELECT suburbcode

  FROM SUBURB

  WHERE suburbname = "Barton");
```

14. Find the properties (pids) that have at least ALL the meter types as the property with id 7 does.

```
SELECT FT.pid, COUNT(FT.pid)

FROM

(SELECT DISTINCT metertype, pid

    FROM PROPERTYMETER

    WHERE metertype IN (

            SELECT metertype

            FROM `PROPERTYMETER`

             WHERE pid = 7)

) AS FT

GROUP BY FT.pid

HAVING COUNT(FT.pid) = 2;
```

15. Find customers who have a property in at least ALL the suburbs that the customer with TFN 32792526 does.

```
SELECT name

FROM `CUSTOMER` X

WHERE NOT EXISTS (

    SELECT *

    FROM `PROPERTY` Y

    WHERE TFN = "32792526" AND NOT EXISTS (

            SELECT *

            FROM `PROPERTY` Z

            WHERE Z.suburbcode = Y.suburbcode and Z.TFN = X.TFN));
```

Step 4: Practicing queries for the coralbleach database

1. List the names of all reefs that have the word Is in their name.

```
SELECT reefname

FROM `reef`

WHERE reefname LIKE "%Is%";
```

2. List the names of all corals that have a thermal threshold greater than 22 degrees.

```
SELECT coralname
FROM `coral`
WHERE `thermalthreshold` > 22;
```

3. Find the names of all reefs which do not have a sample of the coral called Sea Mat and Button Polyps (Palythoa). Use a sub-query in your answer.

Solution I:

```
SELECT reefname

FROM reef

WHERE reefname NOT IN

    (SELECT CS.reefname

    FROM coralsampling CS, coral C
    WHERE CS.coralcode = C.coralcode

  AND C.coralname = "Sea Mat and Button Polyps (Palythoa)");
```

Solution II:

```
SELECT reefname
    FROM `reef`
    WHERE reefname NOT IN (
            SELECT reefname
            FROM `coralsampling` S
            WHERE coralcode IN (
                    SELECT coralcode
                    FROM `coral`
                    WHERE `coralname` = "Sea Mat and Button Polyps (Palythoa)"

            ));
```

4. List the coral name and the number of samples for each coral which had more than 5 samples taken between 01/01/2007 and 01/01/2008.

```
SELECT C.coralname, COUNT(*)

FROM coral C, coralsampling CS

WHERE C.coralcode = CS.coralcode

AND CS.dateofsampling BETWEEN "2007-01-01" AND "2008-01-01"

GROUP BY C.coralcode

HAVING COUNT(*) > 5;
```

5. Find the coral which has the highest average bleach percentage across all its samples.

```
SELECT c.coralcode

FROM coralsampling c

GROUP BY coralcode

HAVING AVG(bleachpercent) >= ALL

    (SELECT AVG(bleachpercent)

    FROM coralsampling s

    GROUP BY s.coralcode);
```

6. List all dates where the temperature reading for the reef Herron Is. (south) was greater than 22 degrees.

```
SELECT dateofreading

FROM reeftemp
```

```
WHERE reefname = 'Herron Is. (south)'
AND temperaturereading > 22;
```

7. Find the names of reefs which have a sample of ALL corals that the reef Linnet has.

Solution I:

```
SELECT reefname
FROM reef X
WHERE NOT EXISTS
     (SELECT *
   FROM coralsampling Y
   WHERE Y.reefname = "Linnet"
   AND NOT EXISTS
               (SELECT *
               FROM coralsampling Z
               WHERE Y.coralcode = Z.coralcode
               AND Z.reefname = X.reefname)
   )
AND X.reefname <> "Linnet";
```

Solution II:

```
SELECT DISTINCT s1.reefname
FROM coralsampling s1
WHERE NOT EXISTS (
     SELECT coralname
     FROM coral
     WHERE coralname NOT IN (
               SELECT coralname
               FROM coralsampling s2
               WHERE s2.reefname = 'Linnet')
) AND s1.reefname <> 'Linnet';
```

8. How many reefs have had a temperature reading > 22 degrees after 01/06/2008

```
SELECT COUNT(DISTINCT reefname)
FROM reeftemp
WHERE dateofreading > "2008-06-01"
AND temperaturereading > 22;
```

9. Find all corals which have a thermal threshold of > 21 and have a bleach percentage of more than 10%.

```
SELECT DISTINCT C.coralname
FROM coral C, coralsampling CS
WHERE C.coralcode = CS.coralcode
AND C.thermalthreshold > 21
AND CS.bleachpercent > 10;
```

10. List details of all samples taken from the reef Keppel-Halfway Is..

```
SELECT s.coralcode, c.coralname, s.dateofsampling, s.bleachpercent, c.thermalthreshold
FROM coralsampling s, coral c
WHERE s.coralcode = c.coralcode
AND s.reefname = 'Keppel-Halfway Is.';
```