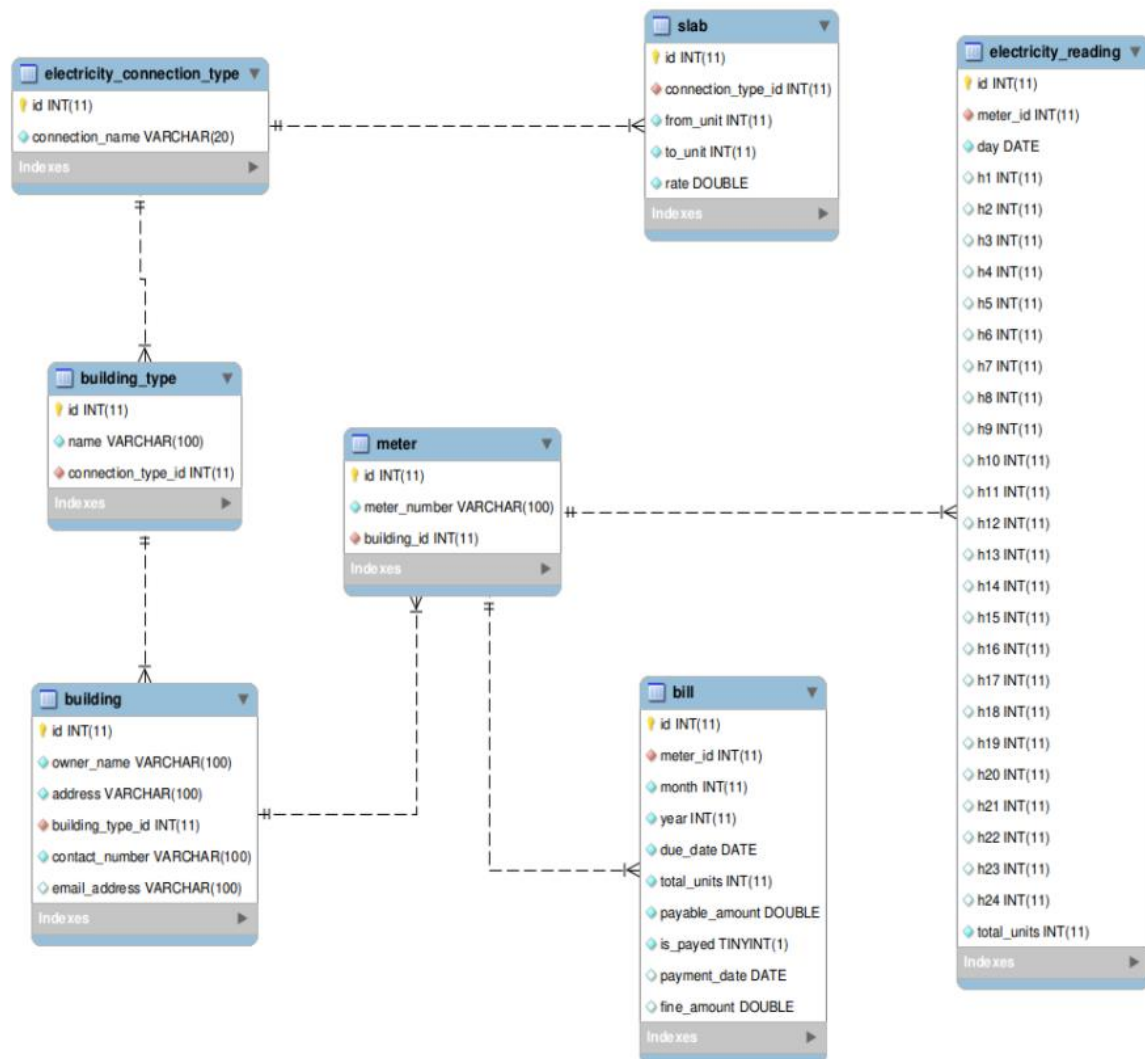


- ! is a Red colored Key so it's a Primary Key which is also a Foreign Key
- ! is a Yellow (non Red) Key so it's only a Primary Key
- ◆ is a blue lined filled diamond so it's a NOT NULL simple attribute
- ◆ is a red colored filled diamond so it's a NOT NULL Foreign Key
- ◇ is a blue lined not filled diamond so it's a simple attribute which can be NULL
- ◇ is a red colored not filled diamond so it's a Foreign Key which can be NULL



Write a query to display all the **bills** with **meter** details. Display the records in ascending order based on **payable amount**.

```

SELECT b.*, m.* FROM bill b
LEFT OUTER JOIN meter m
ON b.meter_id = m.id
ORDER BY b.payable_amount ASC
  
```

Write a query to display the **meter number,owner name,address** and **contact number** of all the buildings. Display the records in ascending order based on **owner name** then by **meter_number**.

```
SELECT m.meter_number, b.owner_name, b.address, b.contact_number FROM building b
LEFT OUTER JOIN meter m
ON b.id = m.building_id
ORDER BY b.owner_name, m.meter_number
```

Write a query to display **owner name,address ,meter number** and **total units** consumed by all the customers in the month **december 2017**.Display the records in descending order by **total_units**.

```
SELECT b.owner_name, b.address, m.meter_number, d.total_units
FROM building b
JOIN meter m
on b.id = m.building_id
JOIN bill d ON m.id = d.meter_id
WHERE month = 12 AND year = 2017
ORDER BY d.total_units DESC
```

Write a query to display the **connection type name** and **number of connections** under each connection name and give alias name as **connection_count**.Display the records in descending order based on **connection_count**.

```
SELECT ect.connection_name, COUNT(ect.id) AS
connection_count FROM electricity_connection_type ect, building_type bt,
building b, meter m
WHERE ect.id = bt.connection_type_id
AND b.building_type_id = bt.id
AND m.building_id = b.id
GROUP BY ect.connection_name
ORDER BY connection_count DESC
```

Write a query to display the **meter number,owner name,address,contact number,building type name** and **connection name** of all buildings.Display the record in ascending order based on **owner name** then by **meter_number**..

```
SELECT m.meter_number, b.owner_name,b.address, b.contact_number,
bt.name, ect.connection_name
FROM meter m, building b, building_type bt,
```

```

electricity_connection_type ect
WHERE m.building_id = b.id
AND b.building_type_id = bt.id AND
bt.connection_type_id = ect.id
ORDER BY owner_name, meter_number

```

Write a query to display the building **owner name** ,**address** and **number of connections** present in each building. Display the records in ascending order based on **owner name**. Give an alias name as **connection_count**.

```

SELECT b.owner_name, b.address, COUNT(bt.connection_type_id)
AS connection_count
FROM building b, building_type bt, electricity_connection_type ect
WHERE b.building_type_id = bt.id AND
bt.connection_type_id = ect.id
GROUP BY b.address, b.owner_name
ORDER BY connection_count

```

Write a query to display **owner name, address ,meter number** and **payable amount** of all the bill which are all not having fine amount that are generated for **2017 December**. Display the records in ascending order based on **owner name**.

```

SELECT b.owner_name, b.address, m.meter_number, b2.payable_amount
FROM building b, meter m, bill b2
WHERE b2.meter_id = m.id AND m.building_id = b.id
AND b2.month = 12 AND b2.year = 2017 AND b2.fine_amount IS NULL
ORDER BY b.owner_name

```

Write a query to display all the building details in which building type is '**Library**'. Display the records in ascending order based on their **owner_name**.

```

SELECT * FROM building
WHERE building_type_id = (SELECT id
FROM building_type
WHERE name = 'Library')
ORDER BY owner_name

```

Write a query to display all the contact number of the building type '**Police Station**'. Display the records in ascending order based on the contact number.

```

SELECT contact_number FROM building

```

```
WHERE building_type_id = (SELECT id
                           FROM building_type
                           WHERE name = 'Police Station')
```

```
ORDER BY contact_number;
```

Write a query to display the **meter number** of the building who have given the **maximum fine amount** of a month. Display the records sorted in ascending order based on **meter_number**.

```
SELECT meter_number FROM meter
WHERE meter_id = (SELECT id
                  FROM bill
                  WHERE fine_amount = 'maximum fine amount')
```

```
ORDER BY meter_number ASC
```

Write a query to display the **owner name, contact number** of the building whose are all billed on '**October**' Month. Display the records in ascending order based on their **owner name**.

```
SELECT owner_name, contact_number FROM building
WHERE id IN (select building_id FROM meter
             WHERE id IN (select meter_id from bill
                           where month = 10))
```

```
ORDER BY owner_name
```

Write a query to display the **meter number** of the building who consumed minimum amount of units in a month. Display the records in ascending order based on **meter number**.

```
SELECT meter_number FROM meter
WHERE id IN (SELECT meter_id FROM bill
             WHERE total_units = (SELECT MIN(total_units)
                                   FROM bill))
```

```
ORDER BY meter_number;
```

Write a query to display the **meter number** of the building who consumed minimum amount of units during 8PM for a day. Display the records sorted in ascending order based on **meter number**.

```
SELECT meter_number
FROM meter
WHERE id IN (SELECT meter_id
```

```

FROM electricity_reading
WHERE h8 IN (SELECT MIN(h8)FROM electricity_reading))
ORDER BY meter_number

```

Write a query to display the **average payable amount** of the buildings whose slab rate is less than **24**. Give an alias name for payable amount as '**payable_amount**'.

```

SELECT avg(payable_amount) AS 'payable_amount'
FROM bill WHERE meter_id IN(
SELECT id FROM meter WHERE building_id IN(
SELECT id FROM building
WHERE building_type_id IN
(SELECT id FROM building_type
WHERE connection_type_id = (
SELECT id FROM electricity_connection_type
WHERE id IN(SELECT connection_type_id FROM slab WHERE rate < 24)))));

```

Write a query to display the number of the '**Commercial**' connection type buildings who have payed the bill. Give an alias name as '**payment_count**'.

```

SELECT COUNT(id) AS payment_count FROM bill
WHERE id IN
(SELECT id FROM bill WHERE meter_id IN
(SELECT id FROM meter WHERE building_id IN
(SELECT id FROM building WHERE building_type_id IN
(SELECT id FROM building_type WHERE connection_type_id = 2))))
AND is_payed = 1

```