DATABASE SCHEMAS

City:

- Id
- Name

Business_id:

- Id
- Name
- City_id

Service_id:

- Id
- Service_name
- business_id

Zone:

- Id
- Name
- business_id

Cluster:

- Id
- Name
- Zone_id

Locality:

- Id
- Name
- Cluster_id

User:

- Id
- First name
- Last name
- Mobile number
- Role: [manager, supervisor, cleaner, registrar]
- Status [active, inactive]
- Create_date
- Created_by
- Updated_on
- Updated_by

Roles:

- Id
- Role_name
- Status [active, inactive]

Toilet:

- Id
- Toilet_name
- Toilet_type []
- Frequency
- Locality_id
- business_id
- Created_on
- Created_by
- Updated_by
- Updated_on
- Image1
- Image2
- Latitude
- Longitude
- Status [active, inactive]

Work_order_history:

- Id
- toilet_id
- action_code
- Created_on
- created_by
- Latitude
- Longitude
- Image1
- Image2

Action_code_master:

- Id
- Action_code [added, updated, clean]
- Action_code_description

- service_id

Servey_question_master:

- Id
- Id_type [supervisor, cleaner, customer]
- Question
- Option - single / multiple / free
- Answers - yes,no
- Status [active / inactive] - (once inactive cannot be activated)

Query to find number of times toilet cleaned in a day by toilet serial_number:

```
SELECT
   t.Id,
   t.Ref_name,
   t.toilet_serial_number,
   t.Frequency,
   COALESCE(w.cleaning_count, 0) AS cleaned_today,
   CASE
      WHEN COALESCE(w.cleaning_count, 0) >= t.Frequency THEN 'Cleaned'
      ELSE 'Not Cleaned'
   END AS cleaning_status
FROM
   Toilet t
LEFT JOIN (
   SELECT
      toilet_serial_number,
      COUNT(*) AS cleaning_count
   FROM
      work_order_history
   WHERE
      cleaning_date = CURDATE() -- Replace with the desired date if needed
   GROUP BY
      toilet_serial_number
) w ON t.toilet_serial_number = w.toilet_serial_number;
```