I have a table containing patient data with the following columns:

- patient_id: Unique identifier for each patient.
- admission_date (timestamp): Date of the the patient entered the ER.
- Adminssion_time (timestamp) : Time of the the patient entered the ER.
- discharge_date (timestamp): Date of the patient left the ER.
- discharge_time (timestamp): time of the patient left the ER.
- Patient_waittime – time frame a patient wait in ER.

Considering this is an ER setting, wait times are typically measured in minutes.

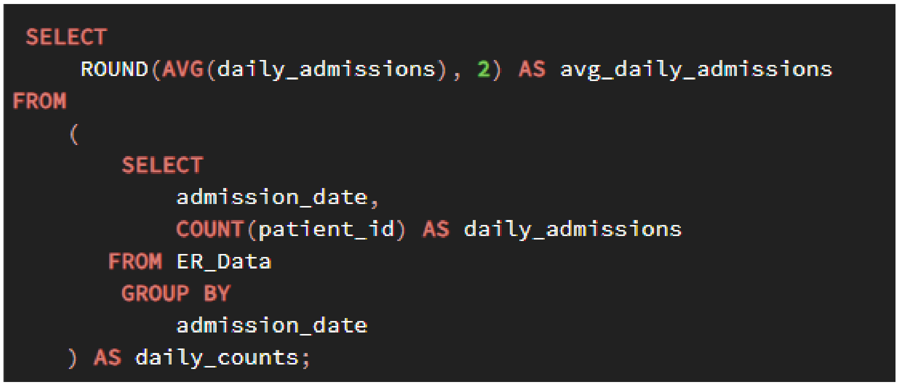
While both admission and discharge timestamps are stored in a 24-hour format (e.g., 2020-03-20 08:47:00), we're interested in analyzing patient volume within the ER.

Specifically, we want to determine the number of patients present in the ER at any given 30-minute interval throughout a 24-hour period. This could be for a specific day or a broader timeframe.

Here are some queries and outcomes :

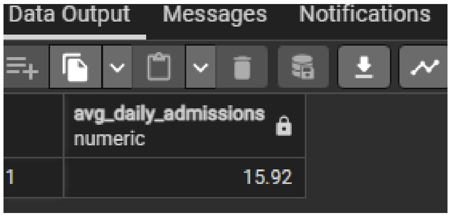
Q. What is the average number of patients admitted to the ER each day?

Query –



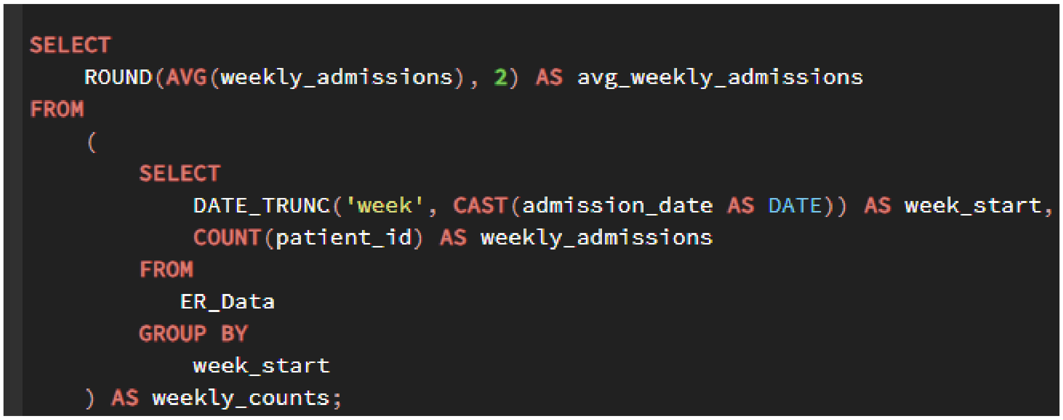
```
 SELECT
     ROUND(AVG(daily_admissions), 2) AS avg_daily_admissions
FROM
    (
        SELECT
            admission_date,
            COUNT(patient_id) AS daily_admissions
       FROM ER_Data
        GROUP BY
            admission_date
    ) AS daily_counts;
```

Outcome



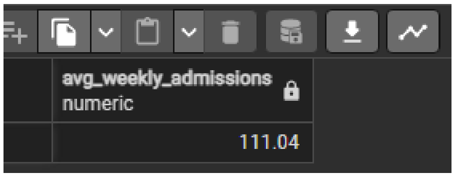
|   | avg_daily_admissions<br>numeric |
|---|---|
| 1 | 15.92 |

Q. How many average admissions do we see on weekly basis.



```
SELECT
    ROUND(AVG(weekly_admissions), 2) AS avg_weekly_admissions
FROM
    (
        SELECT
            DATE_TRUNC('week', CAST(admission_date AS DATE)) AS week_start,
            COUNT(patient_id) AS weekly_admissions
        FROM
           ER_Data
        GROUP BY
            week_start
    ) AS weekly_counts;
```

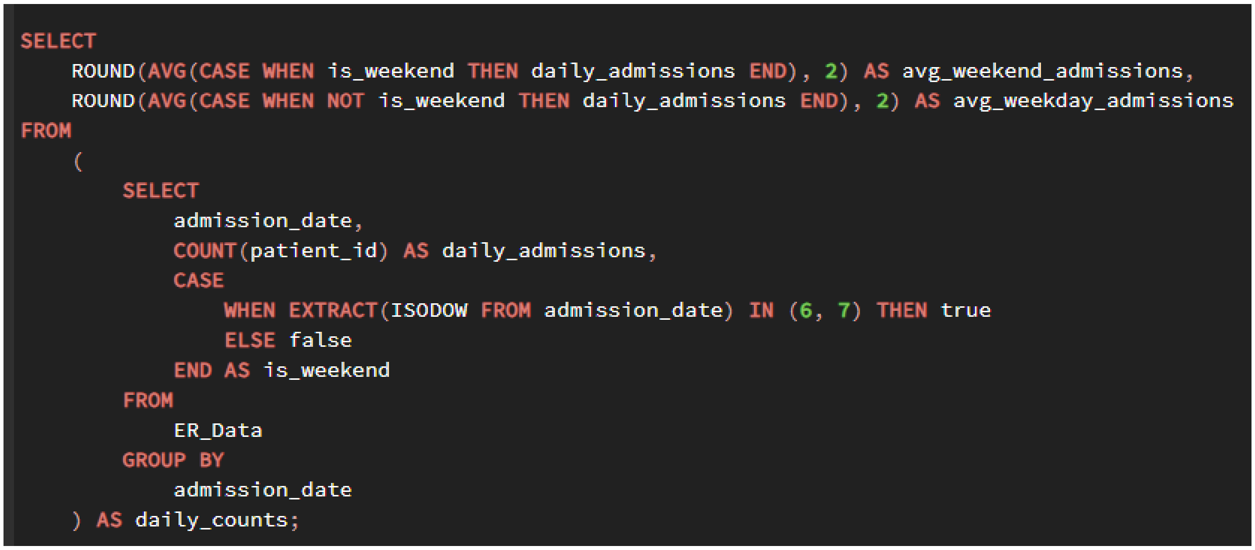
Outcome –



| avg_weekly_admissions<br>numeric |
|---|
| 111.04 |

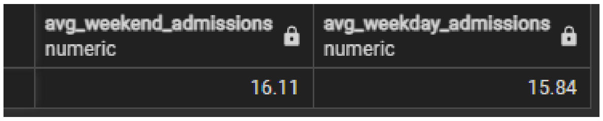
We see about 111-112 footfall on weekly basis.

Q. what's the average weekday vs weekend break up



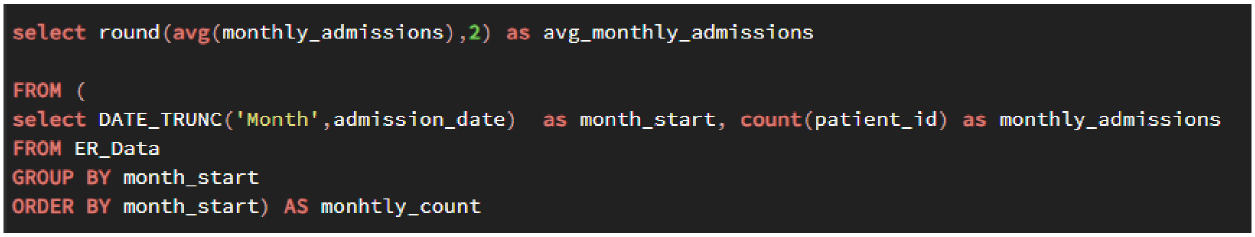
```
SELECT
    ROUND(AVG(CASE WHEN is_weekend THEN daily_admissions END), 2) AS avg_weekend_admissions,
    ROUND(AVG(CASE WHEN NOT is_weekend THEN daily_admissions END), 2) AS avg_weekday_admissions
FROM
    (
        SELECT
            admission_date,
            COUNT(patient_id) AS daily_admissions,
            CASE
                WHEN EXTRACT(ISODOW FROM admission_date) IN (6, 7) THEN true
                ELSE false
            END AS is_weekend
        FROM
            ER_Data
        GROUP BY
            admission_date
    ) AS daily_counts;
```

Outcome –



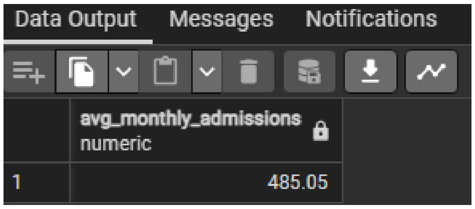
| avg_weekend_admissions<br>numeric | avg_weekday_admissions<br>numeric |
|---|---|
| 16.11 | 15.84 |

Q. What are our monthly admissions?



```
select round(avg(monthly_admissions),2) as avg_monthly_admissions

FROM (
select DATE_TRUNC('Month',admission_date)  as month_start, count(patient_id) as monthly_admissions
FROM ER_Data
GROUP BY month_start
ORDER BY month_start) AS monhtly_count
```

Outcome –



| | avg_monthly_admissions<br>numeric |
|---|---|
| 1 | 485.05 |

Q. can we group average patient wait time?- some conditions are as below

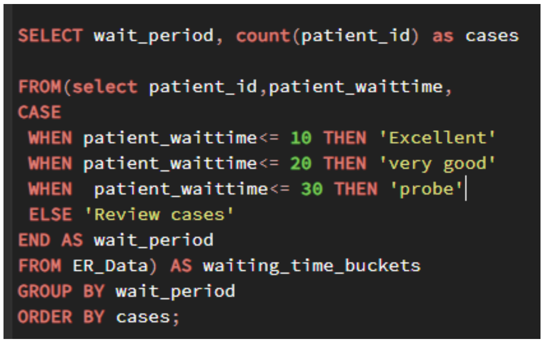
--we need to find cases where average wait time is

--less than 10 mins- excellent

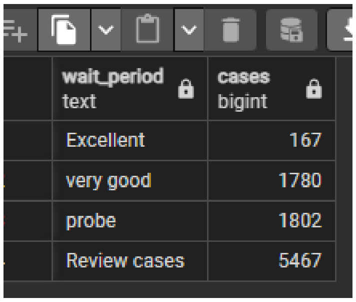
--10-20 mins - very good

-- 20-30 mins- good

- more than 30 – Review cases

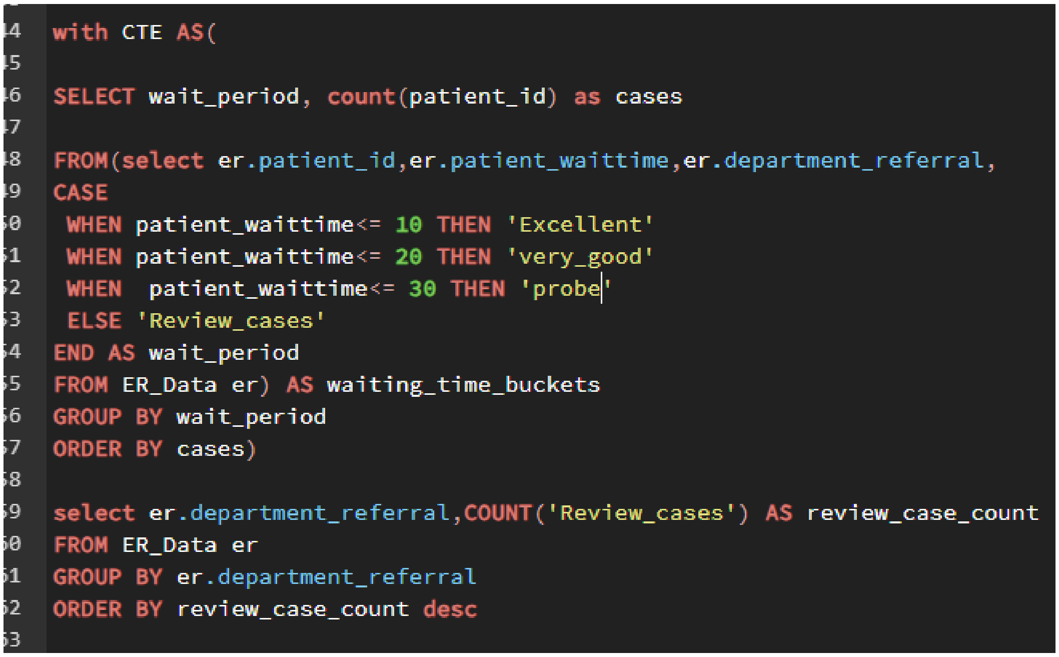


```
SELECT wait_period, count(patient_id) as cases

FROM(select patient_id,patient_waittime,
CASE
 WHEN patient_waittime<= 10 THEN 'Excellent'
 WHEN patient_waittime<= 20 THEN 'very good'
 WHEN  patient_waittime<= 30 THEN 'probe'
 ELSE 'Review cases'
END AS wait_period
FROM ER_Data) AS waiting_time_buckets
GROUP BY wait_period
ORDER BY cases;
```

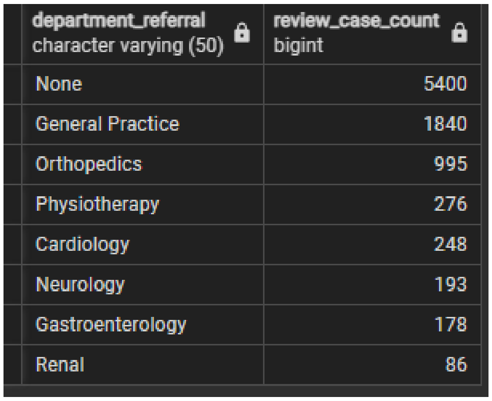


| wait_period<br>text | cases<br>bigint |
|---|---|
| Excellent | 167 |
| very good | 1780 |
| probe | 1802 |
| Review cases | 5467 |

Probe- --now we wish to see review_cases are from which department referral?

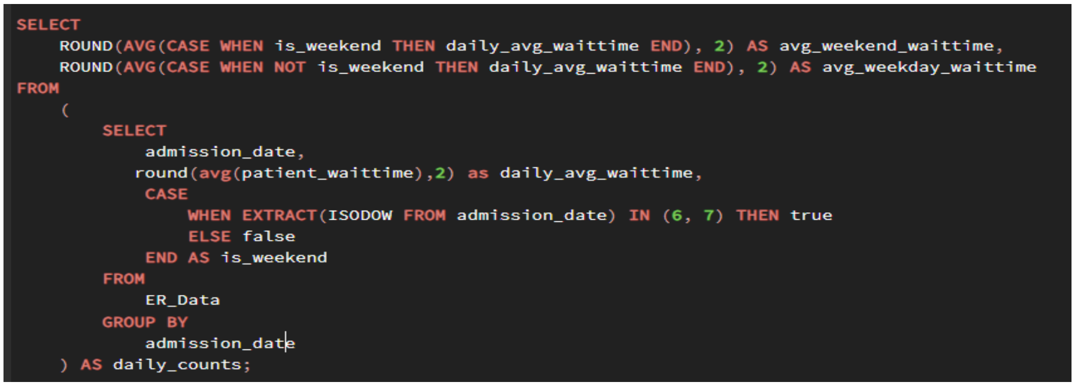


```
with CTE AS(

SELECT wait_period, count(patient_id) as cases

FROM(select er.patient_id,er.patient_waittime,er.department_referral,
CASE
 WHEN patient_waittime<= 10 THEN 'Excellent'
 WHEN patient_waittime<= 20 THEN 'very_good'
 WHEN  patient_waittime<= 30 THEN 'probe'
 ELSE 'Review_cases'
END AS wait_period
FROM ER_Data er) AS waiting_time_buckets
GROUP BY wait_period
ORDER BY cases)

select er.department_referral,COUNT('Review_cases') AS review_case_count
FROM ER_Data er
GROUP BY er.department_referral
ORDER BY review_case_count desc
```



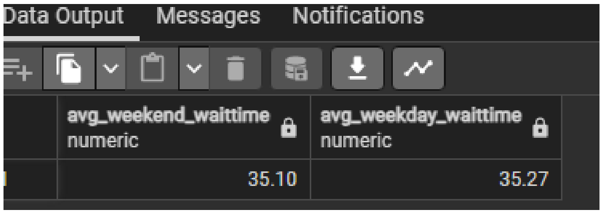
| department_referral<br>character varying (50) | review_case_count<br>bigint |
|---|---|
| None | 5400 |
| General Practice | 1840 |
| Orthopedics | 995 |
| Physiotherapy | 276 |
| Cardiology | 248 |
| Neurology | 193 |
| Gastroenterology | 178 |
| Renal | 86 |

Q. Lets see how wait time affects weekday vs weekend- wait time is the time taken to admit in ER.



```
SELECT
    ROUND(AVG(CASE WHEN is_weekend THEN daily_avg_waittime END), 2) AS avg_weekend_waittime,
    ROUND(AVG(CASE WHEN NOT is_weekend THEN daily_avg_waittime END), 2) AS avg_weekday_waittime
FROM
    (
        SELECT
            admission_date,
           round(avg(patient_waittime),2) as daily_avg_waittime,
            CASE
                WHEN EXTRACT(ISODOW FROM admission_date) IN (6, 7) THEN true
                ELSE false
            END AS is_weekend
        FROM
            ER_Data
        GROUP BY
            admission_date
    ) AS daily_counts;
```

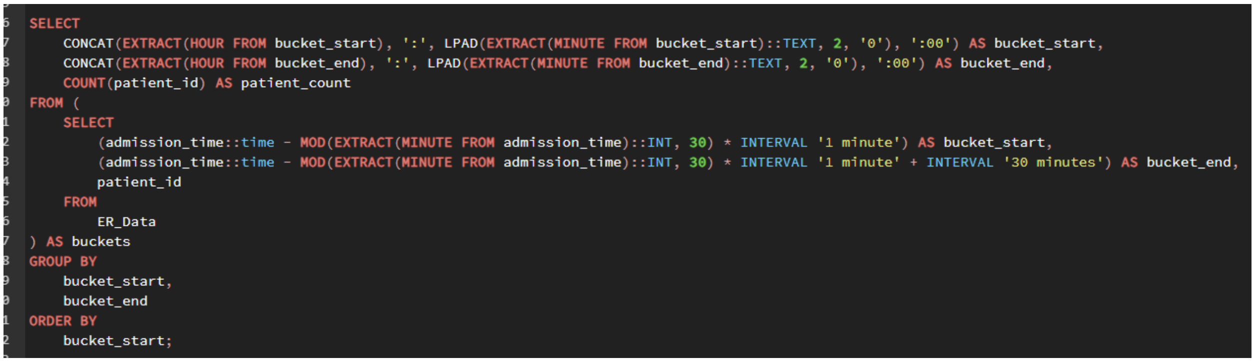
Outcome-



| | avg_weekend_waittime<br>numeric | avg_weekday_waittime<br>numeric |
|---|---|---|
| | 35.10 | 35.27 |

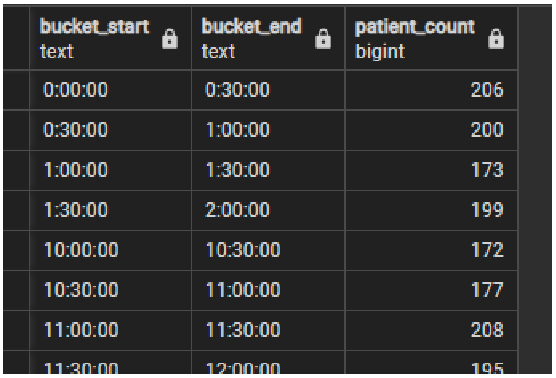
Questions- whats the most busiest time of the day?

Solution – here we group all patient admission data in a half an hour buckets. This way we calculate admissions for the given time period as per the time of admissions. This to find out when we expect admissions.



```sql
SELECT
    CONCAT(EXTRACT(HOUR FROM bucket_start), ':', LPAD(EXTRACT(MINUTE FROM bucket_start)::TEXT, 2, '0'), ':00') AS bucket_start,
    CONCAT(EXTRACT(HOUR FROM bucket_end), ':', LPAD(EXTRACT(MINUTE FROM bucket_end)::TEXT, 2, '0'), ':00') AS bucket_end,
    COUNT(patient_id) AS patient_count
FROM (
    SELECT
        (admission_time::time - MOD(EXTRACT(MINUTE FROM admission_time)::INT, 30) * INTERVAL '1 minute') AS bucket_start,
        (admission_time::time - MOD(EXTRACT(MINUTE FROM admission_time)::INT, 30) * INTERVAL '1 minute' + INTERVAL '30 minutes') AS bucket_end,
        patient_id
    FROM
        ER_Data
) AS buckets
GROUP BY
    bucket_start,
    bucket_end
ORDER BY
    bucket_start;
```

Outcome –



| bucket_start<br>text | bucket_end<br>text | patient_count<br>bigint |
|---|---|---|
| 0:00:00 | 0:30:00 | 206 |
| 0:30:00 | 1:00:00 | 200 |
| 1:00:00 | 1:30:00 | 173 |
| 1:30:00 | 2:00:00 | 199 |
| 10:00:00 | 10:30:00 | 172 |
| 10:30:00 | 11:00:00 | 177 |
| 11:00:00 | 11:30:00 | 208 |
| 11:30:00 | 12:00:00 | 195 |

Note – you can refer to PowerBI visualization to see detailed view.

THE-END.