1. **Create a summary of type of drugs and their total amount used by ethnicity. Report the top usage in each ethnicity group. You may have to make certain assumptions in calculating their total amount.**

Assumptions: I assumed the type of drugs referred to the name of the drug, not the generic "drug_type" column, and I assumed that total amount used referred to the total number of applications, due to variability in dosage units. However, I also think the number of distinct users is an interesting analysis.

Query 1:

```
SELECT
    p.drug,
    COUNT(DISTINCT p.subject_id) AS total_users
FROM
    PRESCRIPTIONS p
GROUP BY
    p.drug
ORDER BY
    total_users DESC
LIMIT 10
```

This query groups the prescriptions table by the unique drugs and counts the total number of distinct patients (by subject_id). It returns the 10 highest by total users descending.

Table 1. Top 10 most used drugs by total distinct users

drug	total_users
Sodium Chloride 0.9% Flush	72
Acetaminophen	69
Heparin	69
D5W	68
Insulin	67
Potassium Chloride	66
Iso-Osmotic Dextrose	60
Magnesium Sulfate	59
Senna	56
NS	56

The most commonly used drugs included Sodium Chloride 0.9% Flush, Acetaminophen, Heparin, and D5W, each prescribed to over 65 distinct patients. Notably, Sodium Chloride 0.9% Flush was the most widely used drug, reaching 72 unique patients. These drugs represent both supportive care items (e.g., IV fluids, electrolytes) and commonly used medications for symptom control or routine treatment.

Query 2:

```
WITH ranked_drug_types AS (
   SELECT
       a.ethnicity,
       p.drug,
       COUNT(*) AS total_used,
       ROW_NUMBER() OVER (
           PARTITION BY a.ethnicity
           ORDER BY COUNT(*) DESC
       ) AS rn
   FROM
       ADMISSIONS a
   JOIN
       PRESCRIPTIONS p ON a.subject_id = p.subject_id
   GROUP BY
       a.ethnicity,
       p.drug
)
SELECT
   ethnicity,
   drug,
   total_used
FROM
   ranked_drug_types
WHERE
   rn <= 1
ORDER BY
   ethnicity, total_used DESC
```

This query first creates the ranked_drug_types table by joining the admissions and prescriptions table on subject_ids, grouping by each drug and ethnicity, and counting the total number of prescriptions in each group. It then selects the most prescribed drug by ethnicity and returns that data.

Table 2. Most prescribed drug by ethnicity and total number of prescriptions

ethnicity	drug	total_used
AMERICAN INDIAN/ALASKA NATIVE FEDERALLY RECOGN...	5% Dextrose	54
ASIAN	D5W	27
BLACK/AFRICAN AMERICAN	Insulin	60
HISPANIC OR LATINO	5% Dextrose	28
HISPANIC/LATINO - PUERTO RICAN	0.9% Sodium Chloride	1290
OTHER	NS	11
UNABLE TO OBTAIN	0.9% Sodium Chloride	28
UNKNOWN/NOT SPECIFIED	D5W	41
WHITE	Potassium Chloride	508

For white patients, Potassium Chloride was the most prescribed drug with a total of 508 prescriptions, suggesting a high frequency of electrolyte supplementation. Puerto Rican Hispanic/Latino patients had 0.9% Sodium Chloride as the top drug, with a super high 1,290 prescriptions, likely from lots of IV usage for a single or few patients. For Black/African American patients, Insulin was most commonly prescribed (60 prescriptions), indicating potential focus on diabetes management. 5% Dextrose and D5W (both IV fluids) appeared as top drugs for several

groups, including American Indian/Alaska Native, Hispanic or Latino, and Asian patients, pointing to widespread supportive care.

2. **Create a summary of procedures performed on patients by age groups (<=19, 20-49, 50-79, >80). Report the top three procedures, along with the name of the procedures, performed in each age group.**

Query (this one's a little long):

```
WITH PATIENT_AGE AS (
   SELECT
       ADMISSIONS.hadm_id,
       DATE_DIFF('day', CAST(dob AS DATE), CAST(admittime AS DATE)) / 365.25 AS
age_at_admission
   FROM
       PATIENTS
   JOIN
       ADMISSIONS ON PATIENTS.subject_id = ADMISSIONS.subject_id
),
PROCS AS (
   SELECT
       PROCEDURES_ICD.hadm_id,
       D_ICD_PROCEDURES.short_title
   FROM
       PROCEDURES_ICD
   JOIN
       D_ICD_PROCEDURES ON D_ICD_PROCEDURES.icd9_code = PROCEDURES_ICD.icd9_code
),
COMBINED AS (
   SELECT
       PATIENT_AGE.age_at_admission,
       PROCS.short_title,
       CASE
           WHEN age_at_admission <= 19 THEN '0-19'
           WHEN age_at_admission BETWEEN 20 AND 49 THEN '20-49'
           WHEN age_at_admission BETWEEN 50 AND 79 THEN '50-79'
           ELSE '80+'
       END AS age_group
   FROM
       PATIENT_AGE
   JOIN
       PROCS ON PATIENT_AGE.hadm_id = PROCS.hadm_id
),
RANKED AS (
   SELECT
       age_group,
       short_title,
       COUNT(*) AS procedure_count,
       ROW_NUMBER() OVER (
           PARTITION BY age_group
           ORDER BY COUNT(*) DESC
       ) AS rn
   FROM
       COMBINED
   GROUP BY
       age_group, short_title
)
SELECT
   age_group,
   short_title AS procedure,
   procedure_count
```

```
FROM
   RANKED
WHERE
   rn <= 3
ORDER BY
   age_group, procedure_count DESC
```

This query does a few things in this order:

1. Create patient_age by calculating the difference between admission time and date of birth
2. Create procedure name by joining the procedure description into procedure_icd table
3. Combine the two tables and create an age_group column by grouping patient_age into the discrete categories specified
4. Create “ranked” column, which is the rank of the procedure type by number of procedures within each age group.
5. Get age group, procedure title, and procedure count for that age group (limiting to the top three by procedure count for each age group)

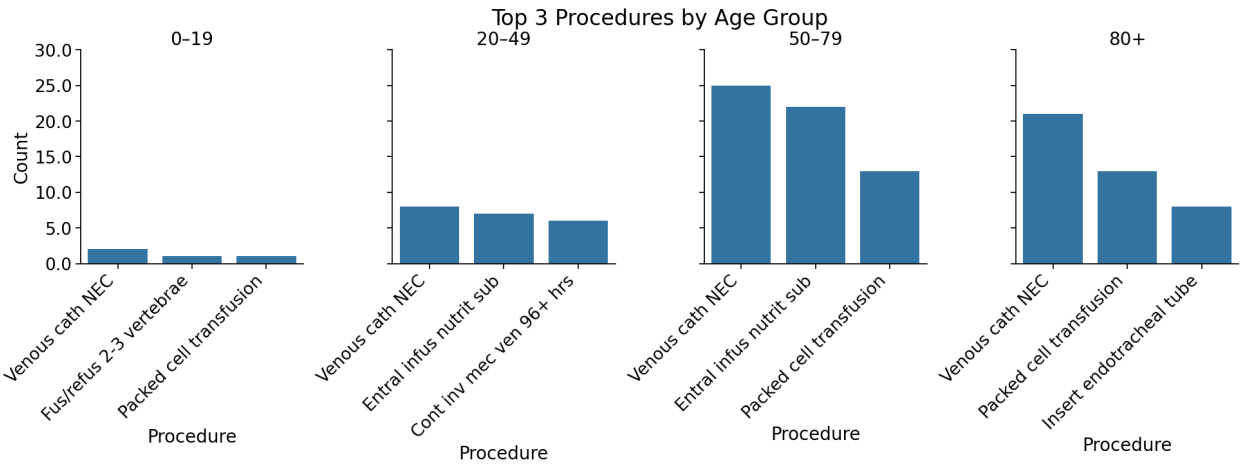

Figure 1. Top three most common procedures by age group

Venous cath NEC" is the most common procedure across all age groups. The number of procedures per age range increased with age. The 50-79 age group had the most procedures, followed closely by the 80+ age group, but the 80+ age group likely contains much less people. Blood transfusions were in the top three procedures for all but the 20-49 group, and the enteral infusion of nutritional substances, i.e. feeding tube, was the second most common procedure in the 20-79 age ranges.

3. How long do patients stay in the ICU? Is there a difference in the ICU length of stay among gender or ethnicity?

Query 1:

```
con.sql("""
```

```
SELECT AVG(LOS)
FROM ICUSTAYS
""")
```

This query simply returns the average length of stay in the ICU, 4.452 days.

Query 2:

```
    SELECT
        a.ETHNICITY,
        AVG(i.LOS) AS avg_los
    FROM ICUSTAYS i
    JOIN ADMISSIONS a ON i.subject_id = a.subject_id AND i.hadm_id = a.hadm_id
    JOIN PATIENTS p ON i.subject_id = p.subject_id
    GROUP BY a.ETHNICITY
    ORDER BY avg_los ASC
```

This query joins the admissions and patients and icu stays tables and returns the average length of stay grouped by ethnicity.

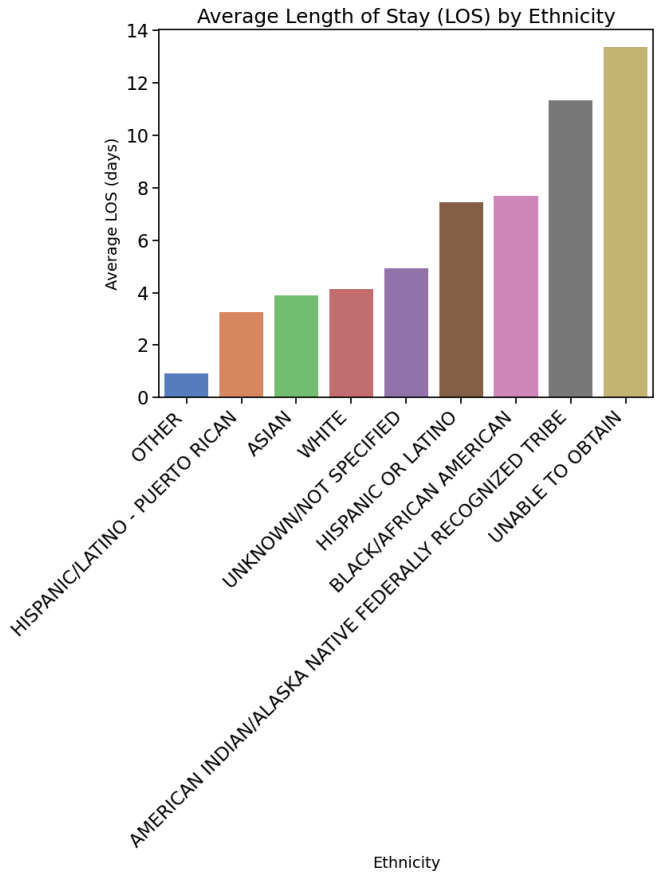

Figure 2. Average length of stay by ethnicity

The average length of stay (avg_los) in this dataset varies significantly across different ethnic groups. Individuals categorized as "OTHER" have the shortest average stay at 0.93 days, followed by Puerto Rican Hispanic/Latino individuals at 3.24 days, and Asian individuals at 3.89 days. White individuals have an average stay of 4.13 days, while those listed as "UNKNOWN/NOT SPECIFIED" stay slightly longer at 4.93 days. The average stay increases notably among broader Hispanic or Latino individuals (7.46 days) and Black/African American individuals (7.68 days). The longest average stays are observed among American Indian/Alaska Native individuals (11.34 days) and those for whom information was "UNABLE TO OBTAIN," who average 13.36 days. All of this information comes with a heavy asterisk because the dataset is tiny.

Query 3:

```
SELECT
    p.GENDER,
    AVG(i.LOS) AS avg_los
FROM ICUSTAYS i
JOIN ADMISSIONS a ON i.subject_id = a.subject_id AND i.hadm_id = a.hadm_id
JOIN PATIENTS p ON i.subject_id = p.subject_id
GROUP BY p.GENDER
ORDER BY avg_los DESC
```

This query joins the admissions and patients and icu stays tables and returns the average length of stay grouped by gender.

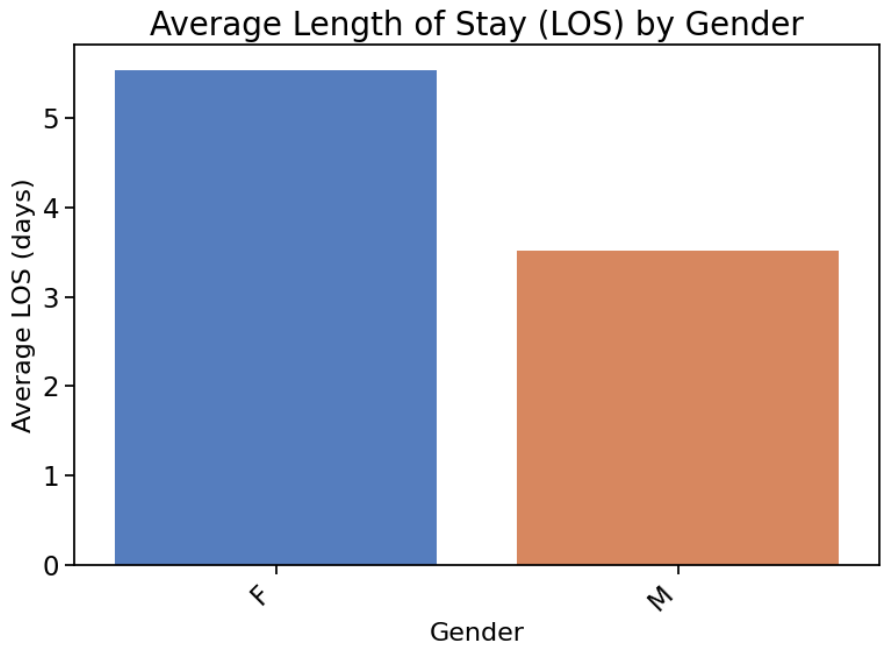

Figure 3. Average length of stay grouped by gender

The average length of stay was noticeably higher for females than for males, potentially reflecting biases, socioeconomic status gaps, or different procedures (men obviously can't get pregnant or give birth [yet?]).

CASSANDRA USAGE

I performed most aggregation and analysis after querying from cassandra because these types of queries are not compatible with the cassandra framework and are trivial to compute in pandas afterwards.

Each question was answered using a single cassandra table, designed for efficiently using a primary key and clustering that used all relevant columns for the question.

Data was uploaded to cassandra using batching (maximum had been set to batch size of 30) to increase the efficiency of the process.

Each cassandra query returned identical results to the above SQL queries.

GEN AI DISCLOSURE STATEMENT

Purpose of Use: I used a limited amount of GAI to assist with writing SQL queries, as I am quite rusty in SQL syntax, so I used ChatGPT to aid in correcting syntax errors. Additionally, though I try to avoid this on real projects, I was swamped this week and in some cells used ChatGPT to provide more descriptive comments on elementary python code.

Tool Used: ChatGPT by OpenAI

Prompts Used: "What is wrong with this SQL query", "Add comments to this python script"