Task

Given the current normalised database structure:

1. Suggest secondary indices based on expected query patterns. Consider factors such as read performance, write amplification and storage overhead. How would you measure the tradeoffs of adding these indices? Provide specific examples of queries that benefit from indexing.
2. Evaluate the tradeoffs of moving the Agent Turns, Human Turns and Steps table to ScyllaDB. Propose a denormalised schema, specifying partition and clustering keys. Discuss how data consistency, query performance and scalability would be affected. How would you measure the tradeoffs?
3. Define key queries that should be supported for analytics and evaluation? For the latter, refer to the offerings of vendors like Langfuse and Arize. Provide query examples and discuss how they would be optimised in both relational and NoSQL environments.
4. Describe how you would implement CQRS for handling queries efficiently. Would you separate the read and write models at ingestion, or use Change Data Capture to maintain a read optimised store? Justify your approach based on factors such as latency consistency, scalability and fault tolerance
5. Write a helm chart to deploy the required database components for a NoSQL setup along with necessary configurations to support steps 2-4. The deployment should run on a single node Kubernetes setup. At a minimum, it should:
    a. Deploy the database along with chosen persistence settings
    b. Configure CDC mechanisms if applicable
    c. Be parameterised so that the chart can be customised and deployed as a part of another installation
    d. Include deployment instructions and validation steps

Database Schema

A session is a directed acyclic graph of agent or human turns. The parent of a human turn is either null (if it is the first turn), or an agent turn. The parent of an agent turn is a human turn. Each agent has one or more steps, differentiated by an integer index. Each step can be of a predefined integer enum type, and the content of the step is a json serialized object

Sessions Table

```
column_name      | data_type
-----------------+--------------------------
uid              | character varying
account_uid      | character varying
title            | character varying
created_at       | timestamp with time zone
deleted_at       | timestamp with time zone
```

Agent Turns

```
column_name  | data_type
-------------+--------------------------
uid          | character varying
session_uid  | character varying
parent_uid   | character varying
active_child | character varying
liked        | boolean
created_at   | timestamp with time zone
```

Human Turns

```
column_name  |        data_type
-------------+--------------------------
uid          | character varying
session_uid  | character varying
parent_uid   | character varying
active_child | character varying
prompt       | character varying
created_at   | timestamp with time zone
```

Steps

```
column_name    |         data_type
---------------+---------------------------
agent_turn_uid | character varying
idx            | integer
tpe            | integer
content        | character varying
created_at     | timestamp with time zone
```

Step Type

```python
class StepType(IntEnum):
    Think = 0
    Text = 1
    Retrieval = 2
    Code = 3
    WebSearch = 4
    Card = 5
    Audio = 6
    Plot = 7
    Table = 8
    Document = 9
    InputRequest = 10
    TurnClassifier = 11
    LangDetection = 12
```