Internship Assignment - Slack AI Data Bot MVP

Goal

Build a minimal Slack app that turns a natural language question into SQL using LangChain, runs it on a Postgres table, and replies in Slack with the results. Keep it simple - no SQL validation, no special guardrails.

What you will build

- A Slack slash command `/ask-data "<question>"`.
- A LangChain-powered NL→SQL step that returns a single SQL statement.
- Direct SQL execution on Postgres and a compact reply in Slack with a small preview.

Tech stack

- Langchain based solution
- N8N workflow solution (Optional)

Database schema and seed

You can use this table otherwise feel free to create your own table

```
CREATE DATABASE analytics;

\c analytics

CREATE TABLE IF NOT EXISTS public.sales_daily (
  date        date           NOT NULL,
  region      text           NOT NULL,
  category    text           NOT NULL,
  revenue     numeric(12,2)  NOT NULL,
  orders      integer        NOT NULL,
  created_at  timestamptz    NOT NULL DEFAULT now(),
  PRIMARY KEY (date, region, category)
);
```

```
INSERT INTO public.sales_daily (date, region, category, revenue, orders) VALUES
('2025-09-01','North','Electronics',125000.50,310),
('2025-09-01','South','Grocery',54000.00,820),
('2025-09-01','West','Fashion',40500.00,190),
('2025-09-02','North','Electronics',132500.00,332),
('2025-09-02','West','Fashion',45500.00,210),
('2025-09-02','East','Grocery',62000.00,870);
```

Minimal behavior requirements

1. Slash command input
   - User posts: `/ask-data show revenue by region for 2025-09-01`.
   - Your app receives the raw text after the command.
2. NL→SQL with LangChain
   - Provide a short prompt that describes the single table and columns.
   - Ask the model to output SQL only - a single SELECT statement.
3. Execute and respond
   - Run the SQL against Postgres
   - Reply in Slack with a well formatted reply :
4. Error handling
   - If execution fails, reply with the error message in a code block.

Stretch goals - optional

- Add a button to export CSV for the last query.
- Add a small chart image for a date range query.
- Add caching to speed up repeated questions.

- In a follow-up phase, add safeguards and validation