To send to candidate

Data Engineering Exercise: Patient Data Pipeline

Objective

Set up a local database, ingest a provided patient CSV file into the database and transform the data into a standardized FHIR Patient table.

We suggest using Python as the base language for ingestion and SQL for transformation but feel free to use any tool or package you prefer.

We provide the code to create the source table in the database, the data to load the table (check .csv in attachment) and the code to create the destination table.

Any questions you have feel free to contact the team.

Below you can find the instructions and template for the exercise solution

Prerequisites

- Write here the prerequisites to run the code

Part 1: Database Setup

1. **Create Database**
2. **Create Source Table (`raw_patient`):**

```
CREATE TABLE raw_patient (
    id SERIAL PRIMARY KEY,
    first_name VARCHAR(100),
    last_name VARCHAR(100),
    birth_date DATE,
    gender VARCHAR(20),
    address VARCHAR(255),
    city VARCHAR(100),
    state VARCHAR(2),
    zip_code VARCHAR(10),
    phone_number VARCHAR(20),
    email VARCHAR(100),
    emergency_contact_name VARCHAR(200),
    emergency_contact_phone VARCHAR(20),
    blood_type VARCHAR(5),
    insurance_provider VARCHAR(100),
    insurance_number VARCHAR(50),
    marital_status VARCHAR(20),
    preferred_language VARCHAR(50),
    nationality VARCHAR(100),
    allergies TEXT,
    last_visit_date DATE,
    created_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TI
MESTAMP,
    updated_at TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TI
MESTAMP
 )
```

Part 2: Data Ingestion with Python

1. **Sample CSV (`patient_data.csv`): sent in annex**

2. **Python Script to Load Data:**

Part 3: Data Transformation to FHIR Patient Table

1. **Create FHIR Patient Table:**

```
CREATE TABLE fhir_patient (
    id VARCHAR(255) PRIMARY KEY, -- Unique ID generated fr
om patient attributes
    full_name VARCHAR(200),
    birth_date DATE,
    gender VARCHAR(20),
    address VARCHAR(255),
    telecom JSONB, -- JSON object with two fields, phone a
nd email
    marital_status VARCHAR(20),
    insurance_number VARCHAR(255),
    nationality VARCHAR(20)
);
```

2. **Transform and Insert Data:**

Deliverables

- GitHub repository containing script(s) that run the following parts
  - Data ingestion: gets the data from .csv and inserts in the database
  - Data transformation: transforms the loaded data into the final FHIR format defined above.
- Small report or README explaining how to rerun the process
- Code should be easily reproducible

Bonus Tasks

- Handle missing `birth_date` or `address`.

- Validate email format before inserting into the database.
- Create a yaml documentation of the two data assets with description and tests