Technical challenge - data/analytics engineer

The goal of this project is to showcase an end to end ELT pipeline from a data source to any data warehouse/data source using Python, SQL and Git to answer the following question:

- **Find the patient(s) with the most generated minutes**

In order to answer this question, you've been provided with the following three datasets:

- steps.csv
  - ID: table ID
  - External_id: patient_id
  - Steps: number of generated steps per submission time
  - Submission_time: the time when the steps are submitted
  - Updated_at: last time when the row was updated
- exercises.csv
  - ID: table ID
  - External_id: patient_id
  - minutes: total duration of an exercise in minutes
  - completed_at: the time when the exercise was completed
  - Updated_at: last time when the row was updated
- patients.csv
  - Patient_id
  - First_name
  - Last_name
  - Country

To solve the problem above, these are the business rules, assumptions and some handy tips:

- As you have probably already noticed, in the steps.csv dataset, there is no column with the completed minutes, but rather a column with the submitted steps. That's fine, as there is a business requirement saying how to convert steps into minutes using the following formula ->

  *minutes = steps * 0.002*

- A single patient can submit steps multiple times, and can complete multiple exercises.
- Please use Python only for data extraction and injection, and use SQL for data manipulation.
- Have in mind that multiple patients can generate the same amount of minutes, meaning that the output table can in theory have more than 1 row
- The output should have the following columns

patient_id:int	first_name:str	last_name:str	country:str	total_minutes:int

Feel free to be creative and if you have knowledge of any of the following technologies (Docker, cloud services, AirFlow, DBT), feel free to use them as well in your solution. If not, no worries, you will learn them at Caspar :)

Delivery of the project: Please share Python code and SQL queries via public git repository including a README file explaining your assumptions and solution implementation details.

Good luck! :)