Data Engineering Test

The purpose of this test is to assess problem-solving skills, creativity, experience with Python and familiarity of data engineering for those who want to join the Dott data team.

We don't expect you to demonstrate expertise only in frameworks (e.g. Flask), but feel free to impress your future teammates with your skills in:

- Python 3
- Data structure and algorithms
- A data store or database of your choice
- Git
- SQL
- GCP specific services if you feel comfortable, like Cloud Run, Cloud Functions, Composer, Dataproc

Problem

You will be creating an application to calculate performance metrics around our deployment cycles. We consider a deployment cycle as the period between a vehicle has been deployed on the street until the moment we pick it up from the street. Between deployment and pickup a vehicle makes rides. It is our ambition to generate as many rides from this vehicle in the shortest amount of time.

The app you will create should be able to surface key metrics about these deployments in terms of revenue, travel distance, and time to ride.

You are given sets of rides CSV files, as well as deployments(use time_resolved) and pickups(use time_created) files. They contain the following fields.

File name	File content
deployments.csv	[task_id, vehicle_id, time_task_created, time_task_resolved]
pickups.csv	[task_id, vehicle_id, qr_code, time_task_created, time_task_resolved]
rides.csv	[ride_id, vehicle_id, time_ride_start, time_ride_end, start_lat, start_lng, end_lat, end_lng, gross_amount]

Based on this data, you should be able to determine how well a vehicle performs for a deployment-ride-pickup cycle based on the **gross_amount** (which you find in the rides.csv)

The possible scenarios are:

Scenario (x = total of rides within a deploy-ride-pickup cycle)	Requirements to be implemented
X >= 5	Show the top 5 rides (revenue, ride_distance, start- and end-points) ordered by the time since the last deployment
X in range(0,4)	Your collection of rides should consist of 5 items, containing: - The top X rides(revenue, distance, start & end point) based on revenue within that cycle - Deployments with the most recent time_task_created to make up a collection of 5 unique items
X == 0	Show up to the most recent 5 deployments based on Time_task_created.

So when you request hits your endpoint, like
http://yourdomain.com/vehicles/{qr_code} or http://yourdomain.com/vehicles/{vehicle_id}

it should render or redirect to your vehicle performance metrics as described above for the last deployment of that vehicle.

It is completely up to you which tools you will use: such as framework, data stores or database, data processing framework, web server, background worker, scheduler etc. But please provide us an explanation why you have chosen a specific tool for your task.

Non functional requirements

1. Your application should serve at least 5000 requests per minute The results of the stress test should be provided.
2. Loading data from CSV files, make sure you have de-duplication logic in place. If I add a new csv file and its content is duplicate of previous files, then gracefully reject the duplicate but log it somewhere.
3. Code should be tested (be it in unittest or another library) both for green path scenario and alternative scenarios like duplicate csv, etc.

4. Deploy to GCP (deployment automation is not needed).
5. Code pushed to a public Github or CodeCommit repo (temporarily).
6. Documentation: Please tell us which approach you have taken, which tools you have selected for your project etc..... and most importantly the reasons behind your decisions. There is no right or wrong when it comes to tool selection in this test.