CWT Data Engineer Case Study

Context

You are tasked with creating a small application that will allow a business travel company, Jameson Business Travel (JBT), to manage records on behalf of its customers. JBT’s customers are other companies for whom JBT must maintain records regarding the travel activities of their staff.

You are provided in the attached file with such travel related records. Each record represents a booked journey, with the following attributes:

• booking ID  
• traveller ID  
• company ID  
• booking date (e.g. 2018-04-28)  
• departure date (e.g. 2020-11-02)  
• origin  
• destination  
• price in USD (integer, for simplicity)

• status

The initial status of a booking is BOOKED.

A booking may be altered, but such alterations are considered valid only if they are performed before the travel date. Booking modifications can be done in one of the 2 ways described below.

A booking may be cancelled by creating a new record with the CANCELLED status. This record copies the original booking record, except for the status, booking date and price. The booking date for the new record is set to the date when the cancellation is performed, and the price is set to the amount JBT will refund the customer. The refunded amount may be less than or equal to the original amount.

A booking may also be exchanged, once or several times. In practice, this is usually done in order to change the date of departure or destination, but in our example this detail will be ignored. Hence, the departure date and destination WILL NOT CHANGE.

For each exchange, a new record is created, with a status of EXCHANGED. This new record copies the original booking, except for the status and the following 2 parameters:

• the booking date is set to the date when the exchange is made

• the price is set to the cost of the exchange, which the customer will have to pay in addition to the original price



© 2022 CWT

We define a TRIP as the grouping of all records that share the same booking ID. A trip record will have a similar structure to a booking record, with the following differences:

* the booking ID renamed to trip ID
* the status field is removed
* a trip date (instead of booking date) corresponding to the latest date of any of the contributing records
* the total sum of the original booking and any alterations

For simplicity, consider a trip being either an original booking, a booking that is cancelled or a booking that was exchanged once or several times. In this exercise, a cancelled booking can’t be exchanged, and an exchanged booking can’t be cancelled.

When answering the next questions, ensure the date invariants described above are applied; if not, exclude the faulty records.

Questions

1. Transform the input records to remove invalid booking alterations (not compliant with the temporal restriction defined above). Use of a UDF for the filtering function (although simpler alternatives exist) is a bonus. How many records are left?
2. Transform the records generated at 1) by combining them into trips (as defined above). How many trips are there? Display the 10 most expensive trips.
3. For each company, find the most expensive route (origin-destination pair). Display the top 10 results in descending order. The records should contain the company ID, route and total spending on that route.
4. For each traveller, calculate the percentage that the cost of exchanges represents out of the total cost of travel. For each company, display the top 2 travellers with the highest percentage, but only if it’s at least 8%. How many records did you find? NB: travellers with the same ID but from different companies are considered different travellers.
5. Open ended, bonus question: identify travel patterns that would help JBT steer its development strategy. What those patterns are and the means you use to identify and analyse them are left to your discretion.

Constraints

* Use of Spark is preferred
* Use of Scala over Python is preferred
* Your solution should be provided as a zipped email attachment
* Your solution should contain instructions on how to run it and should run with minimal setup
* Your solutions to each question can be delivered as files or as output in the console, as long as the context is clear
* Testing and validation of your answers are highly encouraged

Please return your solution within 1 week.