

Database Systems, CSCI 4380-01

Homework # 1 Practice Problem for Question 1 Version 1.03

Question 1 [48 points]. Write the following queries using relational algebra. You may use any valid relational algebra expression, break into multiple steps as needed. However, please make sure that your answers are well-formatted and are easily readable. Also, pay attention to the attributes required in the output!

- (a) Find all passengers who traveled from one timezone to another in January 2026 (the flight must have departed and arrived in January 2026). Output the first and last name of each passenger, their actual flight departure and arrival, their departure and arrival airport code, name, city, and the timezone.

Solution using the standard notation.

- (a) – Rename attributes to enable Cartesian product/theta join

$\rho_{Airports1}(Code1, Name1, City1, Latitude1, Longitude1, Timezone1)(Airports)$

- Rename attributes to enable Cartesian product/theta join

$\rho_{Airports2}(Code2, Name2, City2, Latitude2, Longitude2, Timezone2)(Airports)$

- Pairs of airports in different timezones. Note that the condition is $Code1 > Code2$ (i.e., using $>$, not using $<>$) in order to avoid having pairs where the difference is only the order of the airports in the pair. The decision to use $>$ instead of $<$ is arbitrary. $<$ would have worked equally well.

$R1 = Airports1 \bowtie_{Code1 > Code2 \text{ and } Timezone1 <> Timezone2} Airports2$

- All flights that departed and arrived in January 2026

$C1 = ActualDeparture :: date \geq date'01/01/2026' \text{ and}$

$ActualDeparture :: date \leq date'01/31/2026' \text{ and}$

$ActualArrival :: date \geq date'01/01/2026' \text{ and}$

$ActualArrival :: date \leq date'01/31/2026'$

$R2 = \sigma_{C1}(Flights)$

- Passengers of the flights that departed and arrived in January 2026 and where the departure and arrival airport codes correspond to airports from different timezones. Note that we need to consider both flights from **Code1** to **Code2** and from **Code2** to **Code1**.

$C2 = DepartureAirportCode = Code1 \text{ and } ArrivalAirportCode = Code2$

$R3 = Passengers \bowtie Tickets \bowtie TicketFlights \bowtie R2 \bowtie_{C2} R1$

$C3 = DepartureAirportCode = Code2 \text{ and } ArrivalAirportCode = Code1$

$R4 = Passengers \bowtie Tickets \bowtie TicketFlights \bowtie R2 \bowtie_{C3} R1$

– While attributes are sets, so the order doesn't matter, here we want to ensure that departure airport info (code, name, city, timezone) is listed before the arrival airport info to avoid any mixing between the departure and the arrival airports.

$A1 = FirstName, LastName, ActualDeparture, ActualArrival, DepartureAirportCode, Name1, City1, Timezone1, ArrivalAirportCode, Name2, City2, Timezone2$

$R5 = \pi_{A1}(R3)$

$A2 = FirstName, LastName, ActualDeparture, ActualArrival, DepartureAirportCode, Name2, City2, Timezone2, ArrivalAirportCode, Name1, City1, Timezone1$

$R6 = \pi_{A2}(R4)$

– All requested data for all passengers who traveled from one timezone to another in January 2026

$Result = R6 \cup R7$

Solution using the text notation.

(a) – Rename attributes to enable Cartesian product/theta join

$Airports1(Code1, Name1, City1, Latitude1, Longitude1, Timezone1) = Airports$

– Rename attributes to enable Cartesian product/theta join

$Airports2(Code2, Name2, City2, Latitude2, Longitude2, Timezone2) = Airports$

– Pairs of airports in different timezones. Note that the condition is $Code1 > Code2$ (i.e., using $>$, not using $<>$) in order to avoid having pairs where the difference is only the order of the airports in the pair. The decision to use $>$ instead of $<$ is arbitrary. $<$ would have worked equally well.

$R1 = Airports1 \text{ join_}\{Code1 > Code2 \text{ and } Timezone1 <> Timezone2\} Airports2$

– All flights that departed and arrived in January 2026

$C1 = ActualDeparture :: date \geq date'01/01/2026' \text{ and}$

$ActualDeparture :: date \leq date'01/31/2026' \text{ and}$

$ActualArrival :: date \geq date'01/01/2026' \text{ and}$

$ActualArrival :: date \leq date'01/31/2026'$

$R2 = select_ \{C1\} (Flights)$

– Passengers of the flights that departed and arrived in January 2026 and where the departure and arrival airport codes correspond to airports from different timezones. Note that we need to consider both flights from **Code1** to **Code2** and from **Code2** to **Code1**.

$C2 = DepartureAirportCode = Code1 \text{ and } ArrivalAirportCode = Code2$

$R3 = Passengers * Tickets * TicketFlights * R2 \text{ join_}\{C2\} R1$

$C3 = DepartureAirportCode = Code2 \text{ and } ArrivalAirportCode = Code1$

$R4 = Passengers * Tickets * TicketFlights * R2 \text{ join_}\{C3\} R1$

– While attributes are sets, so the order doesn't matter, here we want to ensure that departure airport info (code, name, city, timezone) is listed before the arrival airport info to avoid any mixing between the departure and the arrival airports.

$A1 = \text{FirstName, LastName, ActualDeparture, ActualArrival, DepartureAirportCode, Name1, City1, Timezone1, ArrivalAirportCode, Name2, City2, Timezone2}$

$R5 = \text{project}_{\{A1\}}(R3)$

$A2 = \text{FirstName, LastName, ActualDeparture, ActualArrival, DepartureAirportCode, Name2, City2, Timezone2, ArrivalAirportCode, Name1, City1, Timezone1}$

$R6 = \text{project}_{\{A2\}}(R4)$

– All requested data for all passengers who traveled from one timezone to another in January 2026

$\text{Result} = R5 \cup R6$

Answer:

$\rho_{\text{Airports1}(\text{Code1, Name1, City1, Latitude1, Longitude1, Timezone1})}(\text{Airports})$

$\rho_{\text{Airports2}(\text{Code2, Name2, City2, Latitude2, Longitude2, Timezone2})}(\text{Airports})$

$R1 = \text{Airports1} \bowtie_{\text{Code1} > \text{Code2 and Timezone1} < \text{Timezone2}} \text{Airports2}$

$C1 = \text{ActualDeparture} :: \text{date} \geq \text{date}'01/01/2026' \text{ and}$

$\text{ActualDeparture} :: \text{date} \leq \text{date}'01/31/2026' \text{ and}$

$\text{ActualArrival} :: \text{date} \geq \text{date}'01/01/2026' \text{ and}$

$\text{ActualArrival} :: \text{date} \leq \text{date}'01/31/2026'$

$R2 = \sigma_{C1}(\text{Flights})$

$C2 = \text{DepartureAirportCode} = \text{Code1 and ArrivalAirportCode} = \text{Code2}$

$R3 = \text{Passengers} \bowtie \text{Tickets} \bowtie \text{TicketFlights} \bowtie R2 \bowtie_{C2} R1$

$C3 = \text{DepartureAirportCode} = \text{Code2 and ArrivalAirportCode} = \text{Code1}$

$R4 = \text{Passengers} \bowtie \text{Tickets} \bowtie \text{TicketFlights} \bowtie R2 \bowtie_{C3} R1$

$A1 = \text{FirstName, LastName, ActualDeparture, ActualArrival, DepartureAirportCode, Name1, City1, Timezone1, ArrivalAirportCode, Name2, City2, Timezone2}$

$R5 = \pi_{A1}(R3)$

$A2 = \text{FirstName, LastName, ActualDeparture, ActualArrival, DepartureAirportCode, Name2, City2, Timezone2, ArrivalAirportCode, Name1, City1, Timezone1}$

$R6 = \pi_{A2}(R4)$

$\text{Result} = R5 \cup R6$