



EXERCISES — Agency Network - 1

version #



IT IS MY JOB TO MAKE SURE YOU DO YOURS.

Copyright

This document is for internal use at EPITA ([website](#)) only.

Copyright © 2021-2022 Assistants <assistants@tickets.assistants.epita.fr>

The use of this document must abide by the following rules:

- ▷ You downloaded it from the assistants' intranet.*
- ▷ This document is strictly personal and must **not** be passed onto someone else.
- ▷ Non-compliance with these rules can lead to severe sanctions.

Contents

1	Agency Network - 1	3
1.1	Goal	3

*<https://intra.assistants.epita.fr>

1 Agency Network - 1

Files to submit:

- agency_network_1/req01.sql
- agency_network_1/req02.sql
- agency_network_1/req03.sql
- agency_network_1/req04.sql
- agency_network_1/req05.sql
- agency_network_1/req06.sql
- agency_network_1/req07.sql
- agency_network_1/req08.sql
- agency_network_1/req09.sql
- agency_network_1/req10.sql
- agency_network_1/req11.sql

Provided files:

- agency_network_1/schema.sql
- agency_network_1/data.sql

Main function: required

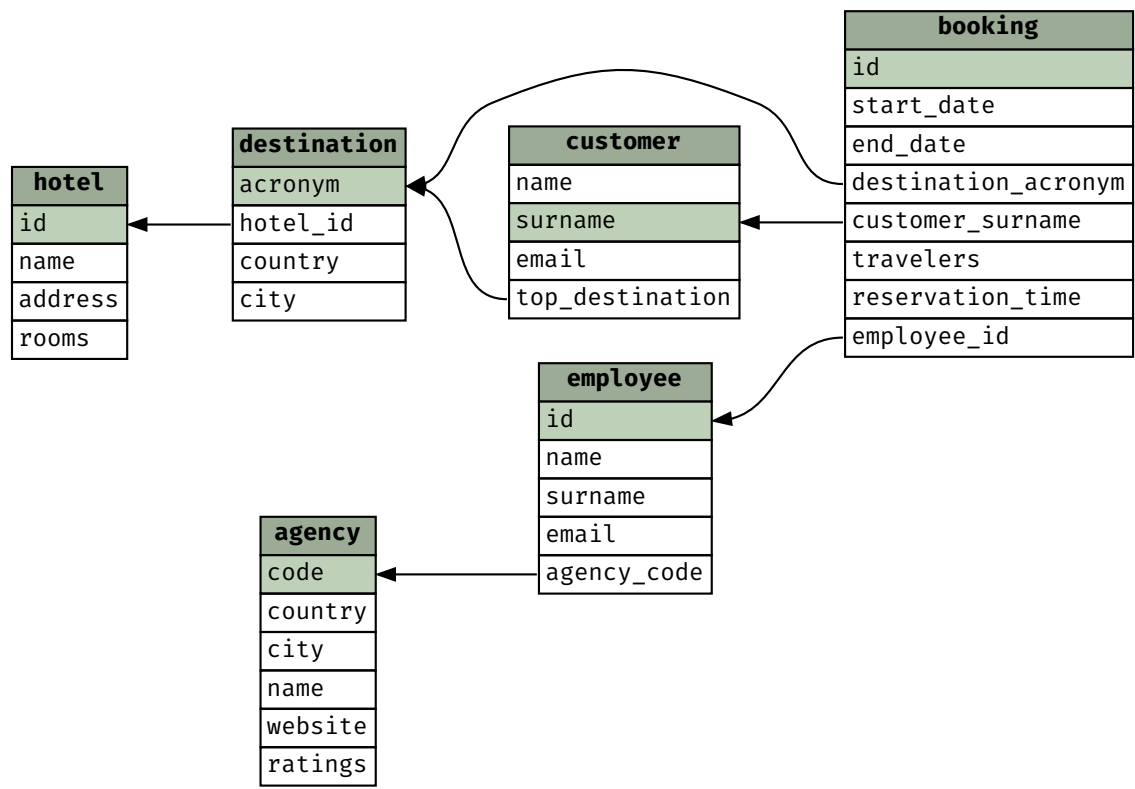
1.1 Goal

Write a few SQL queries to manage a travel agency.

Tables

Look at the `schema.sql` to fully understand what is expected of you.

Relationships



Exercises

Below is the list of exercises you must submit. If the type of ordering is not specified, then it must be ascending.

req01.sql:

You seem to have lost your mind and furiously want to list all bookings from your agencies, ordered by ascending *id*.

Example:

id	start_date	end_date	destination_acronym	customer_surname	travelers	reservation_time	employee_id
1	2019-06-15	2019-06-22	MU1	Lopez	2	2019-02-14 09:43:00	1
2	2019-08-15	2019-08-22	FR2	Latour	7	2019-01-10 10:32:10	2

(continues on next page)

(continued from previous page)

3		2019-12-22		2020-01-15		IN2		De Villardière		5		2019-09-
↪01 10:50:11				3								
4		2019-03-09		2019-03-19		IN1		Sam		2		2018-11-
↪10 16:45:00				4								
5		2019-01-15		2019-02-14		D01		Thepooth		2		2018-12-
↪23 08:10:01				5								
6		2019-02-12		2019-02-27		MU1		Anderson		3		2018-10-
↪17 15:13:27				1								
7		2019-09-09		2019-09-23		TR1		de Villardière		4		2019-05-
↪14 15:39:19				2								
8		2019-10-01		2019-10-22		TR1		Sam		1		2019-06-
↪22 11:34:35				3								
9		2019-01-22		2019-04-22		GP1		Lopez		6		2018-08-
↪17 11:59:59				4								
10		2019-03-15		2019-04-01		ES2		Anderson		5		2019-02-
↪10 02:46:01				5								
(10 rows)												

req02.sql:

In order to help customers choose a destination, the employees want to list all the destinations, specifying their *acronym*, *country* and *city*. Order the result by *acronym*, then by *country* and *city*.

Example:

acronym		country		city

D01		République Dominicaine		Punta Cana
ES1		Espagne		Madrid
ES2		Espagne		Calvià
FR1		France		Le Kremlin-Bicêtre
FR2		France		Le Kremlin-Bicêtre
FR3		France		Le Kremlin-Bicêtre
FR4		France		Marseille
FR5		France		Versailles
FR6		France		Pau
FR7		France		Tulle
FR8		France		Lille
FR9		France		Bordeaux
GP1		Guadeloupe		Le Gosier
GP2		Guadeloupe		Le Gosier
IN1		Inde		New Delhi
IN2		Inde		Rajasthan
MU1		Ile Maurice		Pointe aux Canonnières
RU1		Russie		Sochi
TR1		Turquie		Antalya
UA1		Ukraine		Mykolaivs'ka oblast
(20 rows)				

req03.sql:

In order to ease the reservations, the employees would like to list the *name*, the *address* and the number of rooms available of every hotel. Order by hotel names, in ascending order, then by the number of rooms, in descending order.

Example:

name	rooms	address
Balladins	24	Place de la Gare, 59800 Lille
Bosquet	11	Rue Valéry Meunier, 64000 Pau
Club Med	APT POSTAL 106	HIGUEY, Punta Cana, Dominican
Club Med	550	Tatil Koyu Tekerlektepe Mevkii, 07980 Antalya,
Club Med	405	La Pointe aux Canonniers, Grand Baie, Pointe aux
Club Med	400	Pointe de la Verdure, Le Gosier 97190
Club Med	175	7 Rue Roger Salengro, 94270 Le Kremlin-Bicêtre
Club Med	42	49 Route du Bas du Fort, Guadeloupe 97190,
Club Med	211	Ulitsa Ordzhonikidze, 11 , Sochi, Krasnodarskiy
Club Med	142	2 Boulevard du Général de Gaulle, 94270 Le
Club Med	200	20 Rue Voltaire, 94270 Le Kremlin-Bicêtre
Club Med	120	538 Chandan Hulla, Main, Chhattarpur Ext Rd,
Club Med	220	18 Quai du Port, 13002 Marseille
Club Med	120	2 bis Avenue de Paris, 78000 Versailles
Club Med	120	271 Avenue de Grande Bretagne, 31300 Toulouse
Club Med	40	35 Rue Esprit des Lois, 33000 Bordeaux
Club Med	240	9 Place de l'Église, 19460 Naves
Club Med	520	Plaza de Sta. Ana, 14, 28012 Madrid, Spain
Club Med	110	Av. Tomàs Blanes Tolosa, 4, 07181 Portals Nous,
Club Med	100	Pichola, Udaipur, Rajasthan 313001, India
Club Med	150	Central Ave, 57, Mykolaiv, Mykolaivs'ka oblast,
Club Med	60	

req04.sql:

Your deputy director would like to know the number of cities per destination, to do some statistics. Then, list the number of cities per country among all destinations. Order by the number of cities, in descending order, then by country.

Example:

count	country
7	France
2	Espagne
2	Inde
1	Guadeloupe
1	Ile Maurice
1	République Dominicaine
1	Russie
1	Turquie
1	Ukraine

(9 rows)

req05.sql:

The director is pleased with your previous work, and he wants to go further with your skills. You now have to display countries and cities, without the characters that are not contained between 'A' and 'Z' or 'a' and 'z', because of reasons. Results will be ordered by country and then by city. You may need to use some regex for this, look at the documentation.

Example:

country	city
Espagne	Calvi
Espagne	Madrid
France	Bordeaux
France	LeKremlinBictre
France	LeKremlinBictre
France	LeKremlinBictre
France	Lille
France	Marseille
France	Pau
France	Tulle
France	Versailles
Guadeloupe	LeGosier
Guadeloupe	LeGosier
IleMaurice	PointeauxCanonniere
Inde	NewDelhi
Inde	Rajasthan
RpubliqueDominicaine	PuntaCana
Russie	Sochi
Turquie	Antalya
Ukraine	Mykolaivskaoblast

(20 rows)

req06.sql:

Some customers want to get some information about their reservation. Then, for each reservation, list the number of days of the travel. You will display the following fields: - *destination*: the acronym of the destination - *customer*: the name of the reservation, that means the surname of the customer - *travelers*: the number of travelers for this reservation - *number_of_days*: the number of days the travelers will spend in this destination

Order by *number_of_days* in descending order, then by *destination* and *customer* in ascending order.

Example:

destination	customer	travelers	number_of_days
GP1	Lopez	6	90
DO1	Thepooth	2	30
IN2	De Villardièrè	5	24
TR1	Sam	1	21
ES2	Anderson	5	17
MU1	Anderson	3	15
TR1	de Villardièrè	4	14
IN1	Sam	2	10
FR2	Latour	7	7
MU1	Lopez	2	7

(10 rows)

req07.sql:

Display the name of agencies that do not have a website. Order by name.

Example:

name
Caribbee
STA Travel
Sunny Caraïbes Voyages
Travel Matters

(4 rows)

req08.sql:

Display the three customers who travel the most, ordered by the number of travels in descending order, and then by customer surname in descending order.

Example:

best customers	number of travels
Sam	2
Lopez	2
Anderson	2

(3 rows)

req09.sql:

Display the names of customers who were still travelling on March 11th 2019, ordered by name.

A name must not appear more than once. You can find a keyword for this purpose, we advise you to read the postgresSQL documentation.

Example:

customer_surname

(continues on next page)

(continued from previous page)

```
Lopez
Sam
(2 rows)
```

req10.sql:

Display the email of each customer whose domain is gmail, ordered by email.

Once again, an email must not appear more than once.

Example:

```
      email
-----
marc.latour@gmail.com
quitterle@gmail.com
(2 rows)
```

req11.sql:

You want to have a better overview of your customers trips. Display the id of the booking, followed by a *trip status*, ordered by trip status then by id. The trip can either be 'Booked', 'Ongoing', or 'Done'.

Example:

```
id | trip status
---+-----
 1 | Done
 2 | Done
 3 | Done
 4 | Done
 5 | Done
 6 | Done
 7 | Done
 8 | Done
 9 | Done
10 | Done
(10 rows)
```

It is my job to make sure you do yours.