

TP4

Hotels-Resorts-Rooms-Customers-Bookings

Université Grenoble Alpes

28/03/2023

Bahareh Afshinpour

bahareh.afshinpour@univ-grenoble-alpes.fr

Download two files from <http://afshinpour.com/courses.html>

The screenshot shows the Oracle SQL Developer interface. The main window displays a table with the following columns: HN, RONO, TOTYP, and PRICE. The table contains 21 rows of data. A context menu is open over the table, and the 'SQL Scripts' option is selected. A sub-menu is also open, showing the 'Run SQL Script...' option.

HN	RONO	TOTYP	PRICE
15	9	DWC	82.5
15	10	S	73.5
15	11	S	83
15	12	S	78
15	13	D	81
15	14	S	79
15	15	DWC	78
15	16	DWC	77.5
15	17	S	66
15	18	S	76
15	19	S	66.5
15	20	S	73.5
15	21	S	71.5
15	22	S	79
15	23	S	73.5
15	24	S	73.5
15	25	DWC	72.5
15	26	S	82
15	27	D	83
15	28	D	66.5
15	29	D	71
15	30	D	66.5

- [Data_hotel2023.sql](#)
- [stations-hotels-create.sql](#)

Use these name for relational algebra

RESORTS : $\{\underline{NS}, \text{NomS}, \text{TypeS}, \text{CapCh}\}$ → La station de numéro NS, a pour nom NomS est de type TypeS (mer ou montagne) et a une capacité de CapCh chambres au total.

HOTELS : $\{\underline{NS}, \underline{NH}, \text{NomH}, \text{AdrH}, \text{TelH}, \text{CatH}, \text{Deb}, \text{Fin}, \text{NbCh}\}$ → L'hôtel de numéro NH, se trouvant dans la station de numéro NS, a pour nom NomH, a NbCh chambres, se trouve à l'adresse AdrH, pour téléphone TelH, est de catégorie CatH (nombre d'étoiles) et est ouvert du jour Deb au jour Fin (voir remarque dans la description de la relation **BOOKINGS**).

ROOMS : $\{\underline{NS}, \underline{NH}, \underline{NCH}, \text{TypCh}, \text{Prix}\}$ → La chambre numéro NCH de l'hôtel numéro NH de la station numéro NS est de type TypCh (S=Simple, D=douche, DWC=douche et WC, SDB=salle de bains) et coûte Prix euros.

GUESTS : $\{\underline{NCL}, \text{NomCl}, \text{AdrCl}, \text{TelCl}\}$ → Le client numéro NCL a pour nom NomCl et habite à l'adresse AdrCl avec pour téléphone TelCl.

BOOKINGS : $\{\underline{NS}, \underline{NH}, \underline{NCH}, \underline{\text{Jour}}, \text{NCL}\}$ → La chambre numéro NCH de l'hôtel numéro NH de la station numéro NS est occupée le jour Jour par le client numéro NCL. NCL = null signifie que la chambre est libre. Pour simplifier, on suppose que la relation **BOOKINGS** contient toutes les informations sur l'occupation des chambres pour l'année entière (du 1er janvier au 31 décembre). Jour donne le rang du jour dans l'année (34 dénote le 3 février, par exemple).

AFSHINPB.ROOMS	
PF*	RN NUMBER (*,0)
PF*	HN NUMBER (*,0)
P*	RONO NUMBER (*,0)
	TOTYP VARCHAR2 (5 BYTE)
	PRICE FLOAT (126)
ROOMS_PK (RN, HN, RONO)	
SYS_C0057032 (RN, HN)	

AFSHINPB.HOTELS	
PF*	RN NUMBER (*,0)
P*	HN NUMBER (*,0)
	NAMEH VARCHAR2 (80 BYTE)
	HADR VARCHAR2 (80 BYTE)
	HTEL CHAR (10 BYTE)
	HCAT NUMBER (*,0)
	BEG NUMBER (*,0)
	END NUMBER (*,0)
	RONB NUMBER (*,0)
HOTELS_PK (RN, HN)	
SYS_C0057027 (RN)	

AFSHINPB.RESORTS	
P*	RN NUMBER (*,0)
	NAME VARCHAR2 (30 BYTE)
	TYPE VARCHAR2 (20 BYTE)
	ROCAP NUMBER (*,0)
RESORTS_PK (RN)	

AFSHINPB.BOOKINGS	
PF*	RN NUMBER (*,0)
PF*	HN NUMBER (*,0)
PF*	RONO NUMBER (*,0)
P*	DAY NUMBER (*,0)
F	CUNO NUMBER (*,0)
BOOKINGS_PK (RN, HN, RONO, DAY)	
SYS_C0057040 (RN, HN, RONO)	
SYS_C0057041 (CUNO)	

AFSHINPB.ROOMS	
PF*	RN NUMBER (*,0)
PF*	HN NUMBER (*,0)
P*	RONO NUMBER (*,0)
	TOTYP VARCHAR2 (5 BYTE)
	PRICE FLOAT (126)
ROOMS_PK (RN, HN, RONO)	
SYS_C0057032 (RN, HN)	

AFSHINPB.CUSTOMERS	
P*	CUNO NUMBER (*,0)
	CUN VARCHAR2 (30 BYTE)
	CUADD VARCHAR2 (80 BYTE)
	CUTEL CHAR (10 BYTE)
CUSTOMERS_PK (CUNO)	

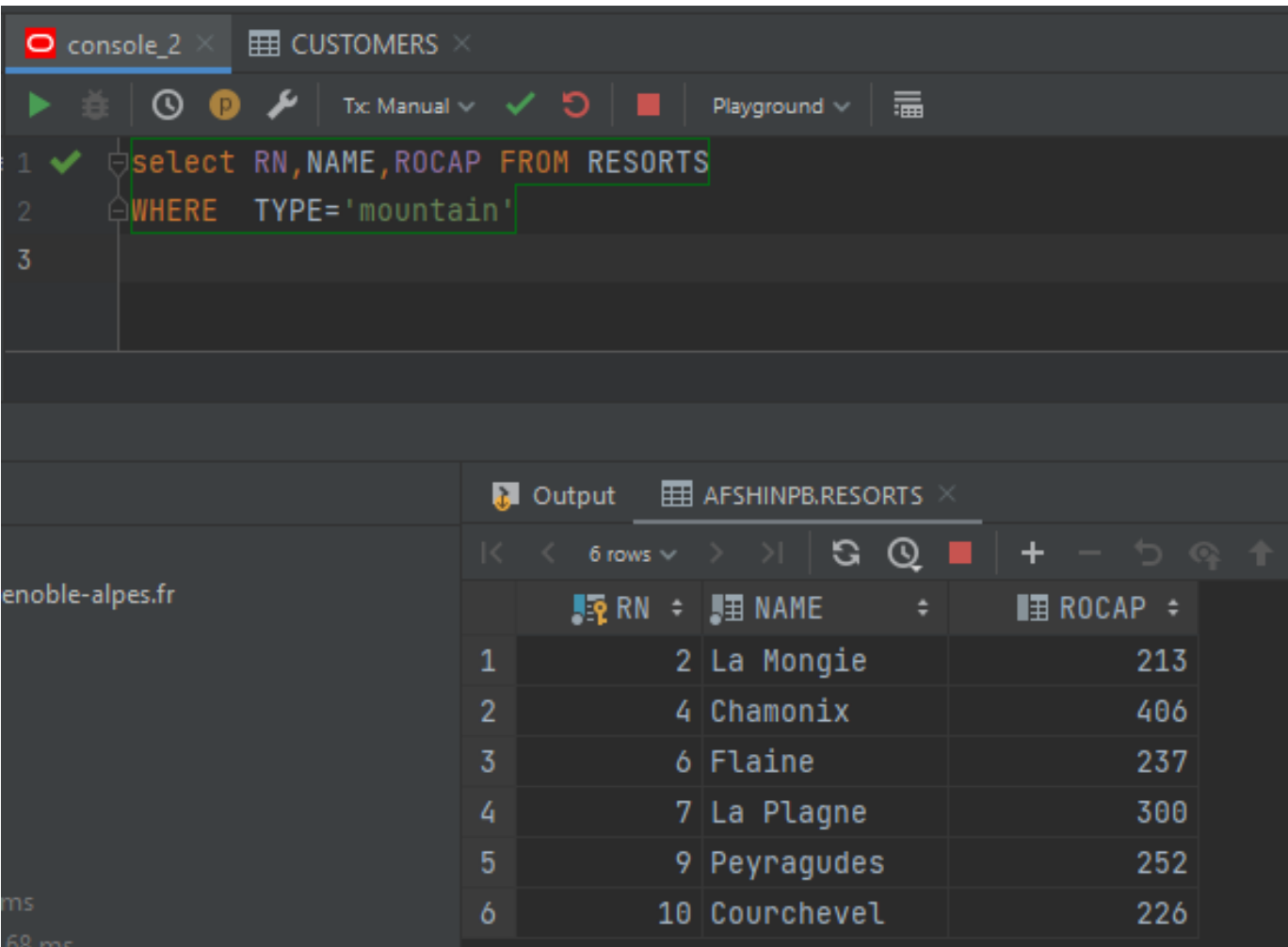
AFSHINPB.BOOKINGS	
PF*	RN NUMBER (*,0)
PF*	HN NUMBER (*,0)
PF*	RONO NUMBER (*,0)
P*	DAY NUMBER (*,0)
F	CUNO NUMBER (*,0)
BOOKINGS_PK (RN, HN, RONO, DAY)	
SYS_C0057040 (RN, HN, RONO)	
SYS_C0057041 (CUNO)	

AFSHINPB.CUSTOMERS	
P*	CUNO NUMBER (*,0)
	CUN VARCHAR2 (30 BYTE)
	CUADD VARCHAR2 (80 BYTE)
	CUTEL CHAR (10 BYTE)
CUSTOMERS_PK (CUNO)	

List of mountain resorts with their number of rooms.

Algèbre :

$$\pi_{\{NS, \text{NomS}, \text{CapCH}\}} \left(\sigma_{\text{TypeS}=\text{montagne}}(\text{RESORTS}) \right)$$



The screenshot shows a SQL playground interface. At the top, there are tabs for 'console_2' and 'CUSTOMERS'. Below the tabs is a toolbar with various icons. The main area contains a SQL query:

```
1 select RN, NAME, ROCAP FROM RESORTS
2 WHERE TYPE='mountain'
3
```

Below the query, there is an 'Output' section showing the results of the query. The output is a table with 6 rows and 3 columns: RN, NAME, and ROCAP. The results are:

	RN	NAME	ROCAP
1	2	La Mongie	213
2	4	Chamonix	406
3	6	Flaine	237
4	7	La Plagne	300
5	9	Peyragudes	252
6	10	Courchevel	226

Answer : 6 rows

Select **Name, Rocap**
from resorts
Where type='mountain'

List of hotels in a seaside resort with their address, telephone, category.

The screenshot shows a database console interface with a SQL query and its results. The query is:

```
1  
2 ✓ Select hn,nameh,Hadr,Htel from HOTELS,RESORTS  
3 where HOTELS.RN= RESORTS.RN and RESORTS.type='seaside'
```

The results are displayed in a table with 37 rows. The table has the following columns: HN, NAMEH, HADR, and HTEL. The first 12 rows are shown below:

	HN	NAMEH	HADR	HTEL
1	1	Hotel Bon SÃ©jour	SÃ©jour@imag.fr	0686117786
2	2	Hotel de la cÃ¢te	cÃ¢te@alice.fr	0951970087
3	3	Hotel de la gare	gare@free.fr	0939828225
4	4	Hotel de la mer	mer@alice.fr	0368703933
5	5	Hotel de la plage	plage@alice.fr	0123225707
6	6	Sea Hotel	Hotel@neuf.fr	0912826060
7	7	Hotel du front de mer	mer@gmail.com	0461913402
8	8	Hotel du rivage	rivage@free.fr	0494516516
9	9	Hotel Terminus	Terminus@gmail.com	0812546583
10	10	Tahiti Hotel	Hotel@free.fr	0373240581
11	1	Hotel Bon SÃ©jour	SÃ©jour@neuf.fr	0395898121
12	4	Hotel de la mer	mer@yahoo.fr	0751139771

37 rows

List of seaside resorts with 4-star hotels.

Algèbre :

$$\pi_{\{NomS\}} ((\sigma_{CatH=4}(\text{HOTELS})) \bowtie_{\text{HOTELS.NS}=\text{RESORTS.NS}} (\sigma_{\text{TypeS}=\text{mer}}(\text{RESORTS})))$$

The screenshot shows a SQL playground interface with a dark theme. At the top, there are tabs for 'console_2' and 'BOOKINGS'. Below the tabs is a toolbar with various icons for execution, refresh, and manual transaction control. The main area contains a SQL query:

```
1 select name from HOTELS, RESORTS
2 where hotels.rn=RESORTS.RN and RESORTS.TYPE='seaside' and HOTELS.hcat=4
```

The query is highlighted with a green border. Below the query editor, there is an 'Output' section with a tab for 'AFSHINPB.RESORTS'. The output shows a table with 4 rows and 1 column named 'NAME':

	NAME
1	Biarritz
2	Cannes
3	Dunkerque
4	Dunkerque

Names and addresses of customers who booked at the mountain.

Algèbre :

$$\pi_{\{NomCl,AdrCl\}} \left(\left(\sigma_{TypeS=montagne}(\text{RESORTS}) \right) \bowtie_{\text{RESORTS.NS=BOOKINGS.NS}} \text{BOOKINGS} \right. \\ \left. \bowtie_{\text{BOOKINGS.NCL=GUESTS.NCL}} \text{GUESTS} \right)$$

```
console_2 x BOOKINGS x HOTELS x RESORTS x CUSTOMERS x
Tx Manual ✓ Playground
1
2 Select distinct CUSTOMERS.CUNO, CUSTOMERS.CUN ,customers.CUADD
3 from CUSTOMERS,BOOKINGS,RESORTS
4 where CUSTOMERS.CUNO= BOOKINGS.CUNO and BOOKINGS.RN=RESORTS.RN and RESORTS.TYPE='mountain';

Output AFSHINPB.CUSTOMERS x
enoble-alpes.fr
501 1230323180 RATCEJAJ Adrienne Adrienne.RATCEJAJ@yahoo.fr
502 1257693383 AMIDOE Carle Carle.AMIDOE@alice.fr
503 1312175297 TEDCUPTUU JÃ@rÃ'me JÃ@rÃ'me.TEDCUPTUU@gmail.com
504 1521381329 CURUSQOQE Achille Achille.CURUSQOQE@alice.fr
505 1566082381 DENIX Arlette Arlette.DENIX@imag.fr
506 1708500468 NELONINO ClÃ@mence ClÃ@mence.NELONINO@neuf.fr
507 1765531312 LNETE Jeannot Jeannot.LNETE@neuf.fr
```

19916 rows

Rooms in a 2-star hotel in a mountain resort with a price lower than € 50.

Algèbre :

$$\pi_{\{NS,NH,NCH\}} \left(\left(\sigma_{\text{Prix} \leq 50}(\text{ROOMS}) \right) \bowtie_{\text{ROOMS.NS}=\text{HOTELS.NS} \wedge \text{ROOMS.NH}=\text{HOTELS.NH}} \left(\sigma_{\text{CatH}=4}(\text{HOTELS}) \right) \bowtie_{\text{HOTELS.NS}=\text{RESORTS.NS}} \left(\sigma_{\text{TypeS}=\text{montagne}}(\text{RESORTS}) \right) \right)$$

The screenshot shows a SQL IDE interface with a query editor and an output window. The query editor contains the following SQL query:

```
1  
2 ✓ Select ROOMS.RN, ROOMS.HN, ROOMS.PRICE  
3   from ROOMS, HOTELS, RESORTS  
4   where ROOMS.HN=HOTELS.HN and rooms.RN=HOTELS.RN and HOTELS.RN=RESORTS.RN  
5   and HOTELS.HCAT=2 AND RESORTS.TYPE='mountain' and ROOMS.PRICE<50;
```

The output window displays the results of the query, showing 148 rows. The columns are labeled RN, HN, and PRICE. The first few rows are:

RN	HN	PRICE
129	6	20
130	6	20
131	6	20
132	6	19
133	6	19
134	6	19
135	6	19
136	6	19
137	6	19

148 rows

```
CREATE TABLE rooms (  
  rn integer NOT NULL,  
  hn integer NOT NULL,  
  rono integer NOT NULL,  
  totyp varchar2(5),  
  price double precision,  
  PRIMARY KEY (rn, hn, rono),  
  FOREIGN KEY (rn, hn) REFERENCES hotels(rn, hn)  
);
```

List of 4-star hotels and their addresses that have only rooms with a bathroom.

Algèbre :

$\pi_{\{\text{NomCl}\}} (\text{GUESTS} \bowtie_{\text{GUESTS.AdrCl}=\text{HOTELS.AdrH}} \text{HOTELS})$

The screenshot shows a SQL IDE interface with a query editor and an output window. The query editor contains the following SQL code:

```
1 SELECT DISTINCT CUSTOMERS.CUNO , CUSTOMERS.CUN
2 FROM HOTELS,CUSTOMERS
3 WHERE CUSTOMERS.CUADD=HOTELS.HADR
```

The output window displays the results of the query, showing two rows of data:

	CUNO	CUN
1	962097349	MARIANA Argine
2	547610084	DIIV Sylvie

2 rows

Customers who booked a room with shower in a seaside resort hotel.

Algèbre :

$$\pi_{\{NomCI\}} (\mathbf{GUESTS} \bowtie_{\mathbf{GUESTS.NCL=BOOKINGS.NCL}} \mathbf{BOOKINGS} \\ \bowtie_{\mathbf{BOOKINGS.NS=ROOMS.NS \wedge BOOKINGS.NH=ROOMS.NH \wedge BOOKINGS.NCH=ROOMS.NCH}} \\ (\sigma_{\mathbf{TypeCh=D}}(\mathbf{ROOMS})) \bowtie_{\mathbf{ROOMS.NS=RESORTS.NS}} (\sigma_{\mathbf{TypeS=mer}}(\mathbf{RESORTS})))$$

```
console_2 x ROOMS x BOOKINGS x HOTELS x RESORTS x CUSTOMERS x
Txc Manual v [check] [refresh] [play] Playground v
1 ✓ SELECT DISTINCT CUSTOMERS.CUNO, CUSTOMERS.CUN
2 FROM BOOKINGS, ROOMS, CUSTOMERS, RESORTS
3 WHERE BOOKINGS.RN=ROOMS.RN and BOOKINGS.HN=ROOMS.HN and BOOKINGS.RONO=ROOMS.RONO
4 and CUSTOMERS.CUNO=BOOKINGS.CUNO and RESORTS.RN=ROOMS.RN and RESORTS.TYPE='seaside'
5 and ROOMS.TOTYP in ('D', 'DWC');
```

```
CREATE TABLE bookings (
  rn integer NOT NULL,
  hn integer NOT NULL,
  rono integer NOT NULL,
  day integer NOT NULL,
  cuno integer,
  PRIMARY KEY (rn, hn, rono, day),
  FOREIGN KEY (rn, hn, rono) REFERENCES rooms(rn, hn, rono),
  FOREIGN KEY (cuno) REFERENCES customers(cuno)
);
```

```
CREATE TABLE rooms (
  rn integer NOT NULL,
  hn integer NOT NULL,
  rono integer NOT NULL,
  totyp varchar2(5),
  price double precision,
  PRIMARY KEY (rn, hn, rono),
  FOREIGN KEY (rn, hn) REFERENCES hotels(rn, hn)
);
```