Lab Session 2

In each question, the desired output is pictured below the query specification.

Some Oracle SQL functions:

Environment

From your account at Paris Sud:
user_name=C##shortlogin_a
password=shortlogin_a

From outside, connecting through ssh:
fulllogin@tp-ssh1.dep-informatique.u-psud.fr

Reminders

- DESC: describe schema of table
- USER_TABLES: describes the relational tables owned by the current user. Interesting field: TABLE_NAME.
- In Oracle, the DUAL table, with a single row and column, used to query constant or pseudocolumn expressions.

DUAL table

```
SELECT 1+2, SYSDATE FROM DUAL;
```

```
       1+2 SYSDATE
---------- ---------
         3 25-SEP-15
```

- loading file1.sql in sqlplus: `@file1;` *--prefix file1 with access path if located in subdirectory*
- display options:

```
set linesize 100; --number of characters in a line
set pagesize 200; -- how many lines before repeating table header
column mycol1 format 99.9;
column mycol2 format a3;
SELECT TO_CHAR(1+2.1,'99') TC,1/3 mycol1,'abcde' mycol2 FROM DUAL;
```

```
TC  MYCOL1 MYC
--- ------ ---
  3     .3 abc
           de
```

Useful functions

- Time manipulation functions:

ADD_MONTHS, EXTRACT( )

```
SELECT EXTRACT(day FROM TO_DATE('21/08/2006','dd/MM/yyyy')) d,
  ADD_MONTHS(
    TO_DATE('09/08/2006', 'dd/MM/yyyy'),
    2)  Deadline    -- adds 2 months to input date
FROM DUAL;
```

```
  D DEADLINE
--- ---------
 21 09-OCT-06
```

- search ordering, and cycle detection in hierarchical query:

BREADTH/DEPTH FIRST, CYCLE

```
WITH t1(id, parent_id, lvl, root_id, path) AS (
...
)
SEARCH DEPTH FIRST BY id SET order1
CYCLE id SET cyclecolumn TO 1 DEFAULT 0
SELECT ..., cyclecolumn
FROM t1
ORDER BY order1;
```

1 First steps

Lab. Ex 1.1
Download the file Oracle_NW.sql from https://www.lri.fr/~groz/documents/Oracle_NW.sql. Load this file in sqlplus. A schema of this Northwind database is provided in Figure 3 below, however the database you have installed does not contain all tables from Figure 3. Using the appropriate system tables, identify the irrelevant tables in the schema (attributes in the database correspond to those in Figure 3, but separated with "_").
This file is allegedly a port of the "Northwind traders" demo database designed by Microsoft to showcase Access.

Lab. Ex 1.2
Which is the fact table? Is there any recursive dimension?

2 OLAP aggregation extensions

Remark: unless specified otherwise, country... means shipping country (incidentally equal to customer country).

Lab. Ex 1.3
Write SQL queries to compute the following expressions. You are not allowed to use UNION.

1. number of customers per country
2. number of orders per country,(country and city), and in total. Results are ordered by alphabetical order on country then city.

SHIP_COUNTRY	SHIP_CITY	NBORDERS
Argentina	Buenos Aires	16
Argentina		16
Austria	Graz	30
Austria	Salzburg	10
Austria		40
Belgium	Bruxelles	7
...		

Ex1.3.2 (left)

C_COUNTRY	S_COUNTRY	QUANTITY	NBORDER
Argentina	Australia	13	2
Argentina	Canada	10	1
Argentina	Denmark	3	1
...			

Ex1.3.3 (above)

3. number of orders and quantity of items shipped (according to order details) for each pair of Customer country and Supplier country. Order result by customer country first, then supplier country.
4. number of orders and quantity for all the cube levels when we only consider Customer and Supplier geography, at the top and country levels only. (i.e., same as before but add totals over each kind of country, and grand total).

C_COUNTRY	S_COUNTRY	QUANTITY	NBORDER
Argentina	Australia	13	2
...			
Argentina		339	16
...			
	USA	6828	244
		51317	830

5. total price (Quantity* UnitPrice) of orders with french suppliers, for each country, region and city. The country must be displayed whenever the region is, and likewise the region whenever the city is. No grand total should be displayed. Propose 2 solutions; each based on a different function to extend GROUP BY.

SHIP_COUNTRY	SHIP_REGION	SHIP_CITY	PRICE
UK		London	4604
UK			4604
UK	Essex	Colchester	510
...			
Switzerland			3557

Ex1.3.5 (above)

S_COUNTRY	S_CITY	NBORDERS
Argentina	Buenos Aires	16
Argentina	whole country	16
Austria	Graz	30
...		
Venezuela	whole country	46
		830

Ex1.3.6 (right)

6. modify your query from question 2 so that the string 'whole country' is displayed instead of NULL on every row that aggregates all cities of a *single* country.

3 Windowing aggregates

Lab. Ex 1.4
Write SQL queries to compute the following expressions.

1. number of orders per country and city on one column, together with total number of orders for the country, and maximal number of orders over the cities in this country on other columns.

```
SHIP_COUNTRY    SHIP_CITY         NBORDERS   NBORDCTY NBORMAXCTY
--------------- --------------- ---------- ---------- ----------
Argentina       Buenos Aires            16         16         16
Austria         Graz                    30         40         30
Austria         Salzburg                10         40         30
...
70 rows selected.
```

2. cities ranked within countries by number of orders, displaying number of orders and rank. There should not be any gaps in the ranks even in presence of ties.

```
SHIP_COUNTRY    SHIP_CITY         NBORDERS       RANK
--------------- --------------- ---------- ----------
Argentina       Buenos Aires            16          1
Austria         Salzburg                10          1
Austria         Graz                    30          2
Belgium         Bruxelles                7          1
Belgium         Charleroi               12          2
...
70 rows selected.
```

3. add to query 2 the percentage of the total number of orders reached by each city within a country.

```
SHIP_COUNTRY    SHIP_CITY         NBORDERS       RANK PERCENTG
--------------- --------------- ---------- ---------- --------
Argentina       Buenos Aires            16          1     1.00
Austria         Salzburg                10          1      .25
Austria         Graz                    30          2      .75
Belgium         Bruxelles                7          1      .37
...
```

Ex. 1.4.3(left)

```
  ORDER_ID      PRICE
---------- ----------
     10250       1813
     ...
     11071        510
     11073        300
     11074      244.3
```

Ex. 1.4.4(right)

4. total price for each order, filtering out all the orders whose price is higher than 110% of the preceding order (defining the preceding OrderID as the largest among smaller OrderID) (you may nest queries).

5. product sold in highest quantity per year, with the quantity. Propose one answer using windowing functions and another that does not (you may nest queries).

```
      YEAR PRODUCT_NAME                                   QTITY
---------- ---------------------------------------- ----------
      1998 Konbu                                           659
      1997 Gnocchi di nonna Alice                          971
      1996 Gorgonzola Telino                               444
```

4 Recursive queries

Lab. Ex 1.5 (warmup)
Use a hierarchical query on the DUAL table to create a table listing integers from 1 to 60.

Lab. Ex 1.6
Generate the list of the next 30 months (format: MON-YY) starting from today.

Lab. Ex 1.7
Write a query to display the supervisor of each employee and the employee's distance to its topmost manager. Modify the query to compute results through depth first search. Modify the data to introduce a cycle in the supervisor relationship. Evaluate the inpact of such cycles on the query.

Lab. Ex 1.8 (Recursive queries beyond hierarchies)

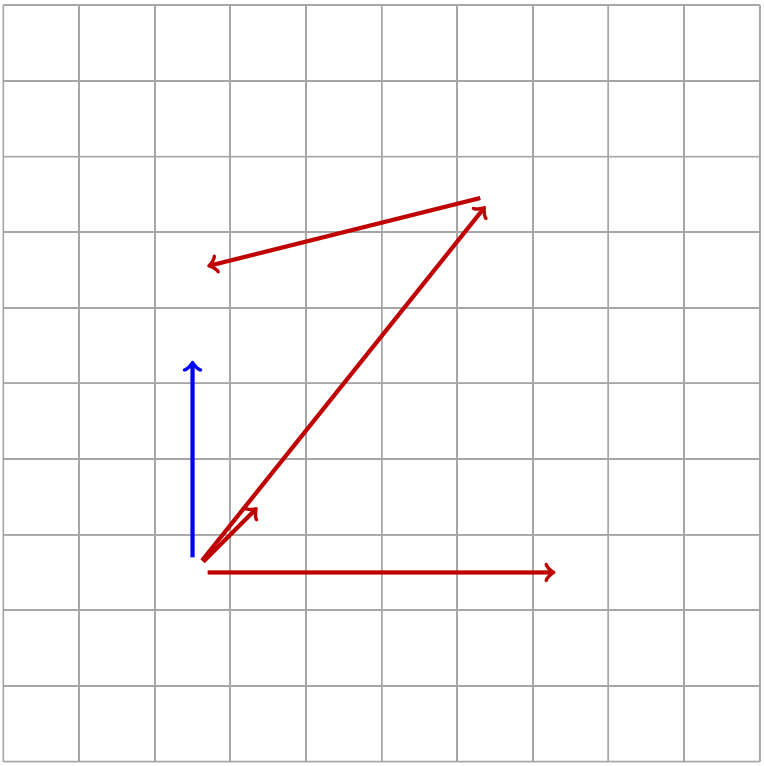

Figure 1: A set of possible moves: (2,2,2,5)...

Below, please find two sample recursive queries. Both are somewhat artifical, but we discussed more realistic examples (dealing with hierarchies) in the lectures. You may find the second question a bit harder as it combines `OVER()` clause and CTEs.

1. (rather trivial, more like a warmup) Give u_{50}, for the Syracuse sequence $(u_n)_{n\geq 0}$ defined as follows: $u_0 = 127$, $\begin{cases} u_{n+1} = & u_n/2 \text{ if } u_n \text{ is even} \\ u_{n+1} = & 3 * u_n/2 + 1 \text{ if } u_n \text{ is odd} \end{cases}$

   Hint: `MOD(5,2)=1, TRUNC(3.5,0)=2`

2. We are given a chessboard of some dimension. We first put an object at (2,2). Each turn, the object is moved by choosing one of its possible moves at random (uniform distribution). The relation `D` provided in `deplacements.csv` tells you which moves are authorized from each position. A possible move is represented as a sequence of 4 integers: `x_origin, y_origin, x_destination, y_destination`.

   (a) Is it possible to reach (0,9) in *exactly* two steps? in exactly 4? within 4 steps at most?
   You will use a query that could be generalized to an arbitrary large number of steps (theoretically: the DBMS may not be able to evaluate it efficiently) so a query joining 2 or 4 copies of `D` explicitly would not do.

   (b) Compute the probability to reach each position after (exactly) 5 steps

   (c) To check your results, export the preceding query as a csv then usethe python script `plotting-probas-heatmap.py` to produce Figure 2 (once again, use `ipython` as the script relies on the `pandas` library).

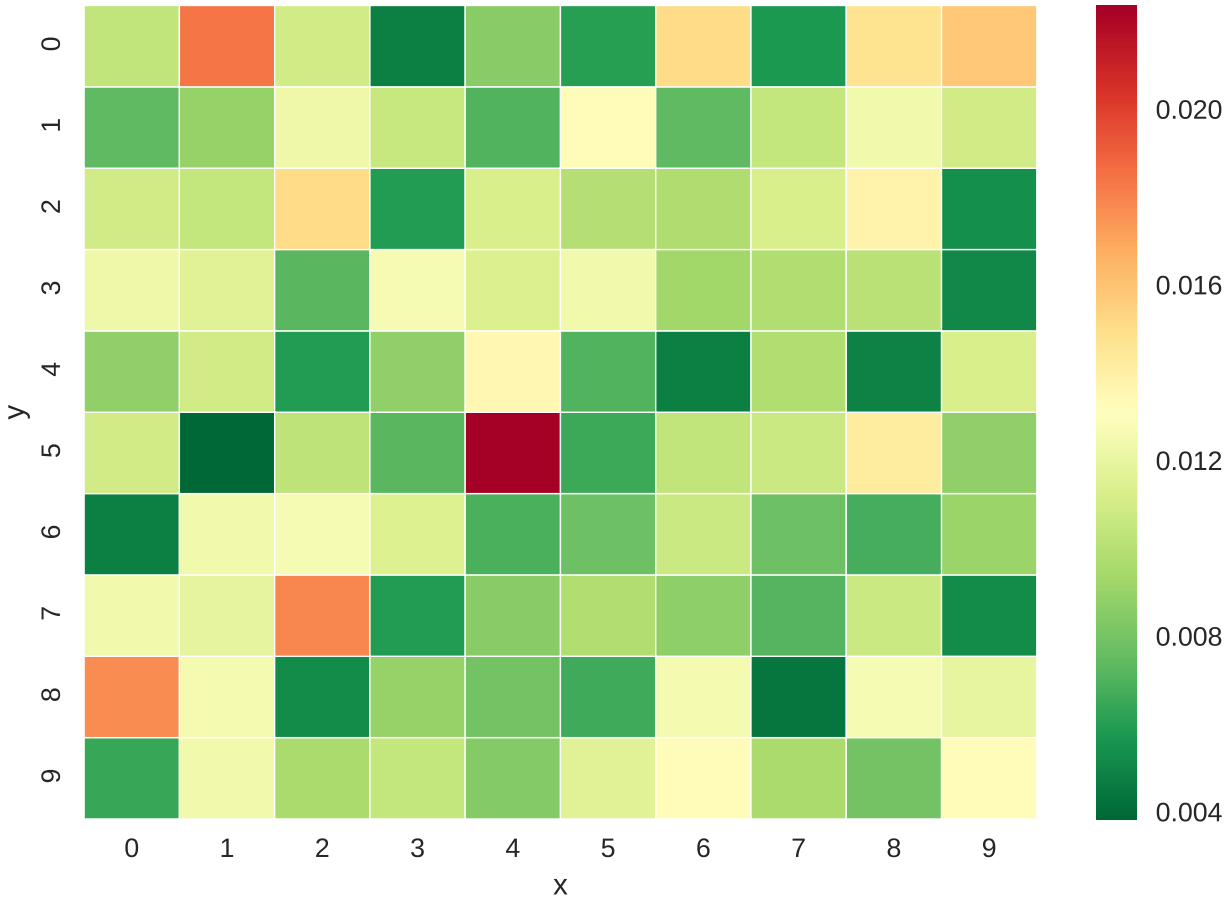

Figure 2: Probabilities after 5 Steps

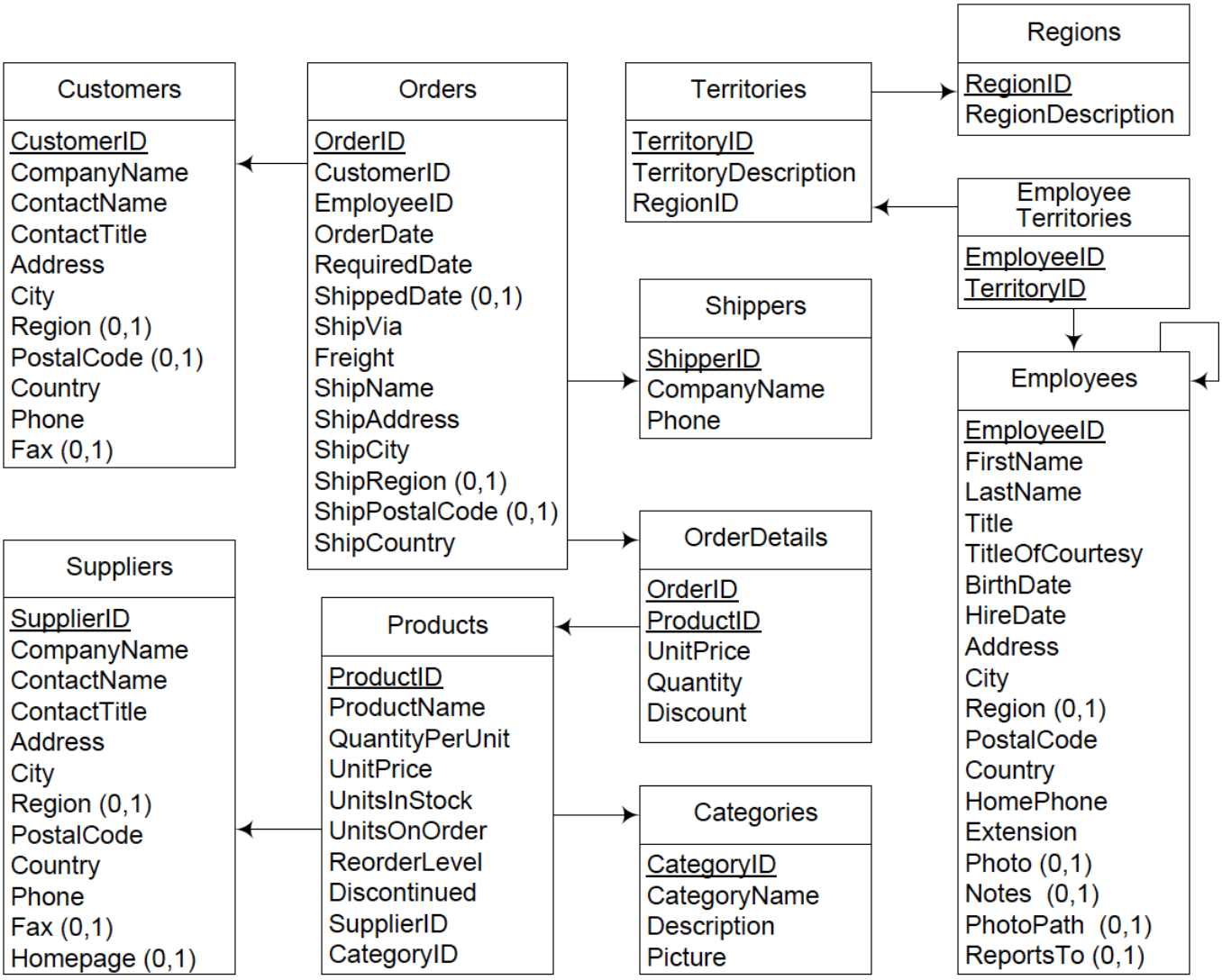

Figure 3: (Approximative) Relational schema for the Northwind database