T - Web Development Seminar

T-WEB-x00

Day 12

SQL

1.25

Day 12

repository name: web_seminar_day12_$ACADEMICYEAR
repository rights: ramassage-tek
language: SQL

- Your repository must contain the totality of your source files, but no useless files (binary, temp files, obj files,...).

You should take note that we will not be testing with the same DB as you.
That means that you shall only use general request (and no request like "id = 42" for example).

EXERCISE 1 — 1PT

Turn in: ./ex_01/ex_01.sql

Write a query that displays the **number** of members and their **average age**, rounded to the nearest integer.
The columns must be named "Number of members" and "Average age".

A man of 55 years and 11 months is considered to be only 55 years old.

EXERCISE 2 — 1PT

Turn in: ./ex_02/ex_02.sql

Write a query that displays only the **zip code** where there is more than one individual; display them in ascending order.
The column must be named "Zip codes".

EXERCISE 3 — 1PT

Turn in: ./ex_03/ex_03.sql

Write a query that displays for each floor its **number**, the **total number of seats** and the **total number of rooms**, sorted by ascending number of seats.
The columns must be named "Floor number", "Total number of seats" and "Total number of rooms".

EXERCISE 4 — 1PT

Turn in: ./ex_04/ex_04.sql

Write a query that displays the **first 92 characters of the summary** of movies whose id is odd and between 42 and 84.
The column must be named "Summaries".

EXERCISE 5 — 1PT

Turn in: ./ex_05/ex_05.sql

Write a query that displays the **email addresses** of the members in the table profiles, replacing the string "machin.com" by "cap-gemini.fr".
The whole list must be sorted by reverse alphabetical order.
The column must be named "New email addresses".

EXERCISE 6 — 1PT

Turn in: ./ex_06/ex_06.sql

Write a query that displays for each movie their **title** and the **number of days** since they were released.
The release date must be defined.
The columns must be named: "Movie title" and "Number of days passed".

EXERCISE 7 — 1PT

Turn in: ./ex_07/ex_07.sql

Write a query that displays the **title** of movies whose first letter of their title is between 'O' and 'T' included.
The whole list has to be sorted in alphabetical order.
The column must be named "Movie title".

EXERCISE 8 — 1PT

Turn in: ./ex_08/ex_08.sql

Write a query that displays the **name** of the table genres whose id is not between 6 and 12.

Those numbers should not be included in the final result.

EXERCISE 9 — 1PT

Turn in: ./ex_09/ex_09.sql

Write a query that displays the **title** and **min_duration** of all the movies.
The result has to be sorted by descending length of title and then sorted by ascending movie duration.

EXERCISE 10 — 1PT

Turn in: ./ex_10/ex_10.sql

Write a query that must sum the **prod_year** of all the movies in a column "Sum prod_year".
Each year must be summed only once.

EXERCISE 11 — 1PT

Turn in: ./ex_11/ex_11.sql

Write a query that **adds a new entry** in the subscription table.
Its name must be "Premium", its summary "For the privileged", its price 80 and the duration of subscription 126.

EXERCISE 12 — 1PT

Turn in: ./ex_12/ex_12.sql

Go back to the query of the previous exercise and add the **premium subscription** 5 times.

You must only use one INSERT INTO statement.

EXERCISE 13 — 1PT

Turn in: ./ex_13/ex_13.sql

Delete the **last 4 subscriptions** of the subscription table.

EXERCISE 14 — 1PT

Turn in: ./ex_14/ex_14.sql

Update the name of the last subscription to "Premium++".

EXERCISE 15 — 1PT

Turn in: ./ex_15/ex_15.sql

Update the **email addresses** of members by replacing the string "machin.com" by "cap-gemini.fr".

EXERCISE 16 — 1PT

Turn in: ./ex_16/ex_16.sql

Delete the movies whose **prod_year** is equal to 0.

EXERCISE 17 — 1PT

Turn in: ./ex_17/ex_17.sql

Update the field **producer_id** in the movies table.
When the field is not defined, set it to the id of the producer who has the less movies. Moreover the producer name must finish by "film".

EXERCISE 18 — 1PT

Turn in: ./ex_18/ex_18.sql

Delete all movies in the movie table that were released in 1990.

EXERCISE 19 — 1PT

Turn in: ./ex_19/ex_19.sql

Add in the job table the **list of jobs** given to you in the .csv (the file must be in the same format and must be in */tmp/jobs.csv*).

EXERCISE 20 — 1PT

Turn in: ./ex_20/ex_20.sql

Export the **content** of the movie table with format CSV in a file in */tmp/movies.csv*.