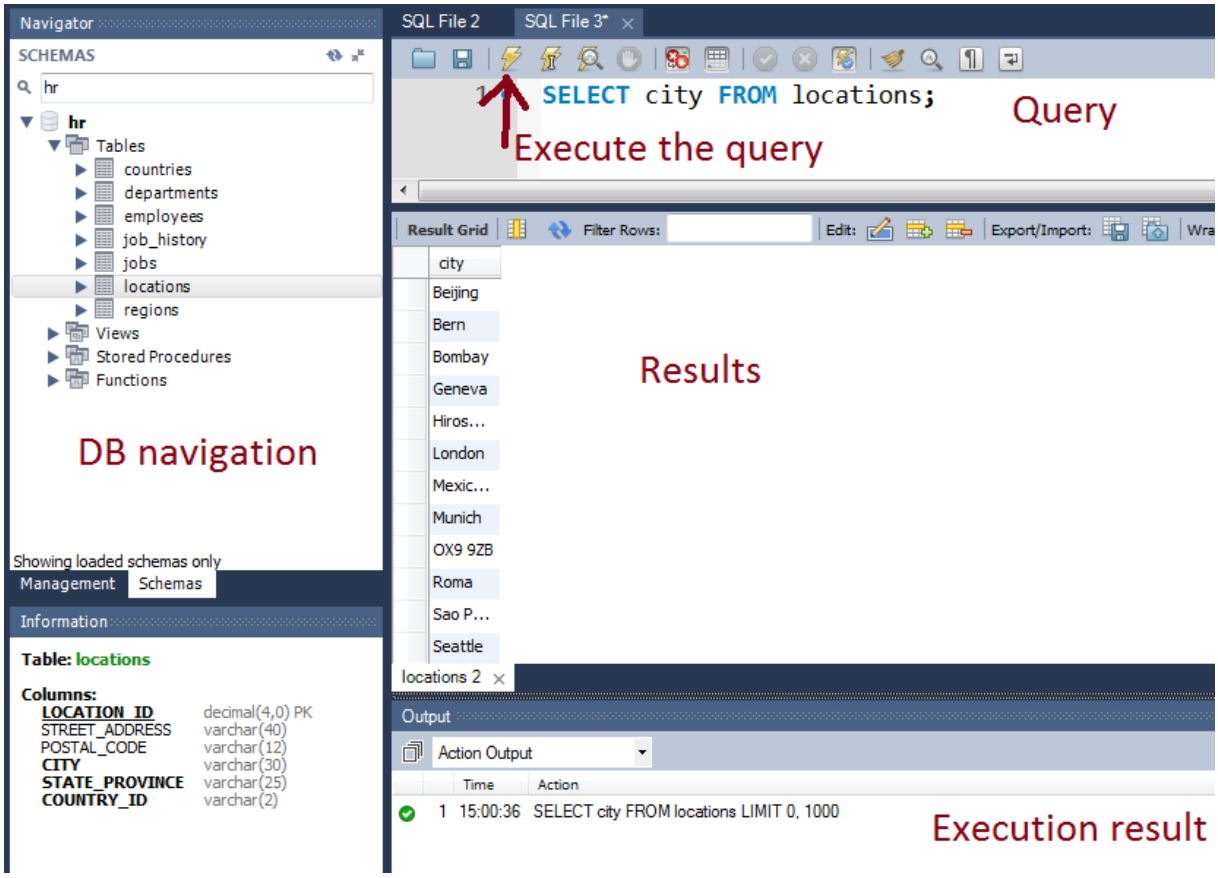
Lab #5 - Create/Insert/Load - 1h30

Intro

Connection to the database : Click the *MySQL Workbench* icon on your desktop, then select the local connection. The database server is already installed on your virtual machine and the *humanresources* database has already been imported.



Part 1 – Creating a database

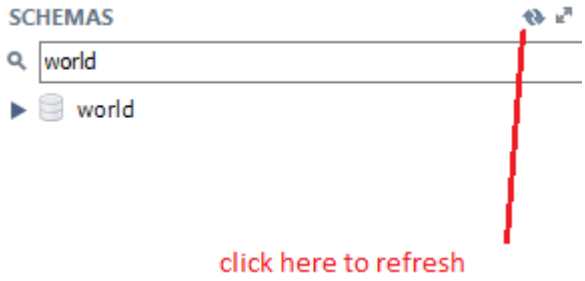
Today's syntax

```
CREATE DATABASE db_name

INSERT [LOW_PRIORITY | DELAYED | HIGH_PRIORITY] [IGNORE]
    [INTO] tbl_name
        [(col_name,...)]
    {VALUES | VALUE} ({expr | DEFAULT},...),(...),...
    [ ON DUPLICATE KEY UPDATE
      col_name=expr
        [, col_name=expr] ... ]
```

Syntax for create table :https:/dev.mysql.com/doc/refman/5.7/en/create-table.html

1. Create a database called « world »



2. Write a SQL statement to create a simple table countries including columns country_id,country_name and region_id, where the country is made of 1 or 3 characters, the name cannot exceed 255 characters and the region ID is an integer.

3. Write a query to insert the country « France", identified by the id « FRA », with region id '1'
4. Write a query to insert the country « Mexico », identified by the id « MEX », no region specified
5. Alter the table countries to add the column country_id to the primary key (Google is your friend). You can use command line or the menus of the Workbench.

6. Try to insert the country France as in question 3
7. Alter the table countries to set the *country_name* as *not null*
8. Create a table « region » made of two columns, id (an auto increment integer) and a mandatory column region_name as a string of maximum 60 caracters.

   https://dev.mysql.com/doc/refman/5.7/en/example-auto-increment.html

9. Alter the table countries to add a foreign key reference from countries.region_id to region.id.

   Why isn't it possible ? Find a way.

10. Write a query to select the country names and their region names

	REGION_ID	COUNTRY_ID	COUNTRY_NAME	REGION_NAME
	1	FRA	France	Europe
	2	MEX	Mexico	Central America

11. Complete manually the regions to obtain the following table content

REGION_ID	REGION_NAME
1	Europe
2	Central America
3	North America
4	South America
5	Asia
6	Oceania
7	Africa
NULL	NULL

12. Clear the table countries, i.e. delete all rows

13. Load the countries from the CSV File : countries.csv that is on Mootse (see here for documentation – otherwise use Google)

COUNTRY_ID	COUNTRY_NAME	REGION_ID
ABW	Aruba	2
AFG	Afghanistan	5
AGO	Angola	7
AIA	Anguilla	2
ALB	Albania	1
AND	Andorra	1
ANT	Netherlands Antilles	2
ARE	United Arab Emirates	5
ARG	Argentina	4

14. Load the SQL file cities.sql using the Workbench interface Tools->Import

```
1  SELECT * FROM world.city;
```

Result Grid | Filter Rows: | Edit:

ID	Name	CountryCode	District	Population
1	Kabul	AFG	Kabol	1780000
2	Qandahar	AFG	Qandahar	237500
3	Herat	AFG	Herat	186800
4	Mazar-e...	AFG	Balkh	127800
5	Amster...	NLD	Noord-H...	731200
6	Rotterdam	NLD	Zuid-Holl...	593321
7	Haag	NLD	Zuid-Holl...	440900
8	Utrecht	NLD	Utrecht	234323

Advanced – Index usage (will be corrected during lecture)

15. Create a query to select the pair of cities of the same country, so that the smallest city has at least 1M (million) population and the largest has at least 1M more population than the smallest one

16. How long was the query ? Use the keyword EXPLAIN before your query to get information on the query plan. Take a screenshot of it

17. Alter the table city to add an index on population then reset the query cache with the command « RESET QUERY CACHE. Execute the query 15 again, what do you observe on the execution time ? Use explain to get more information.