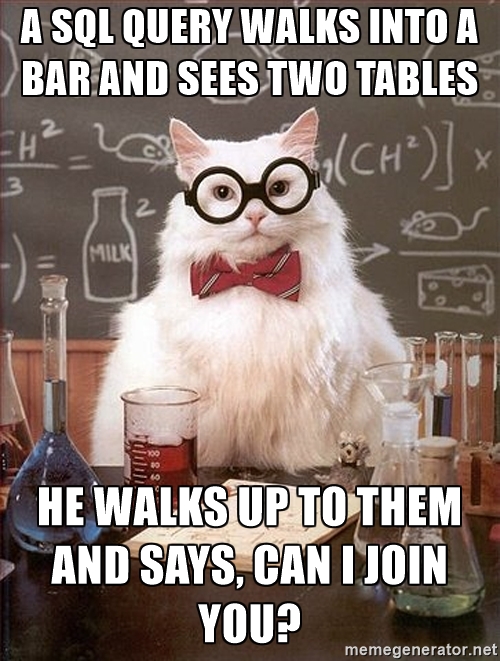
0x0E. SQL - More queries

**SQLMySQL**

* By: Guillaume
* Weight: 1
* Project over - took place from Oct 18, 2023 5:00 AM to Oct 19, 2023 5:00 AM
* An auto review will be launched at the deadline

In a nutshell…

* **Auto QA review:** 57.2/104 mandatory & 15.6/24 optional
* **Altogether:**  **90.75%**
  + Mandatory: 55.0%
  + Optional: 65.0%
  + Calculation:  55.0% + (55.0% \* 65.0%)  == **90.75%**



Resources

**Read or watch**:

* [How To Create a New User and Grant Permissions in MySQL](https://intranet.alxswe.com/rltoken/RniBKj48bnIN8xpXhGl1yA)
* [How To Use MySQL GRANT Statement To Grant Privileges To a User](https://intranet.alxswe.com/rltoken/FIiEIvA6IN_hSKM5TvgyxQ)
* [MySQL constraints](https://intranet.alxswe.com/rltoken/LrovGa6N-OE2ID_tpWZRaQ)
* [SQL technique: subqueries](https://intranet.alxswe.com/rltoken/kR71h5zjkPtx4kBoVf7q0g)
* [Basic query operation: the join](https://intranet.alxswe.com/rltoken/rNMJeQ1jbNTCljbvCSjf6w)
* [SQL technique: multiple joins and the distinct keyword](https://intranet.alxswe.com/rltoken/HhZ6TJ1q5S0aR4lhfpKdOQ)
* [SQL technique: join types](https://intranet.alxswe.com/rltoken/T6FZUQdsMzr8hgNInBzudA)
* [SQL technique: union and minus](https://intranet.alxswe.com/rltoken/Nd-sdM8QUpf0YKIlXzVv4w)
* [MySQL Cheat Sheet](https://intranet.alxswe.com/rltoken/iSNyinU6SPWTGDUWMmcRkg)
* [The Seven Types of SQL Joins](https://intranet.alxswe.com/rltoken/-plhBsra0N7BOuFoEg--zg)
* [MySQL Tutorial](https://intranet.alxswe.com/rltoken/I4Lws_eQrIrNTbkZvvk-oQ)
* [SQL Style Guide](https://intranet.alxswe.com/rltoken/051eAEP_rePBU7jeh879GA)
* [MySQL 8.0 SQL Statement Syntax](https://intranet.alxswe.com/rltoken/YavbYiraYFr8oTukT_N6eQ)

Extra resources around relational database model design:

* [Design](https://intranet.alxswe.com/rltoken/EWLRPeqr5sQ9AqfoG_KXxw)
* [Normalization](https://intranet.alxswe.com/rltoken/mqBhYoSYbhH5ZZrhDcY0kA)
* [ER Modeling](https://intranet.alxswe.com/rltoken/R0exkJmf-2ddKjGfa8D0dA)

Learning Objectives

At the end of this project, you are expected to be able to [explain to anyone](https://intranet.alxswe.com/rltoken/0qci3VdIVdKJXldEZ6zAjA), **without the help of Google**:

General

* How to create a new MySQL user
* How to manage privileges for a user to a database or table
* What’s a PRIMARY KEY
* What’s a FOREIGN KEY
* How to use NOT NULL and UNIQUE constraints
* How to retrieve datas from multiple tables in one request
* What are subqueries
* What are JOIN and UNION

Copyright - Plagiarism

* You are tasked to come up with solutions for the tasks below yourself to meet with the above learning objectives.
* You will not be able to meet the objectives of this or any following project by copying and pasting someone else’s work.
* You are not allowed to publish any content of this project.
* Any form of plagiarism is strictly forbidden and will result in removal from the program.

Requirements

General

* Allowed editors: vi, vim, emacs
* All your files will be executed on Ubuntu 20.04 LTS using MySQL 8.0 (version 8.0.25)
* All your files should end with a new line
* All your SQL queries should have a comment just before (i.e. syntax above)
* All your files should start by a comment describing the task
* All SQL keywords should be in uppercase (SELECT, WHERE…)
* A README.md file, at the root of the folder of the project, is mandatory
* The length of your files will be tested using wc

More Info

Comments for your SQL file:

$ cat my\_script.sql

-- 3 first students in the Batch ID=3

-- because Batch 3 is the best!

SELECT id, name FROM students WHERE batch\_id = 3 ORDER BY created\_at DESC LIMIT 3;

$

Install MySQL 8.0 on Ubuntu 20.04 LTS

$ sudo apt update

$ sudo apt install mysql-server

...

$ mysql --version

mysql Ver 8.0.25-0ubuntu0.20.04.1 for Linux on x86\_64 ((Ubuntu))

$

Connect to your MySQL server:

$ sudo mysql

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 11

Server version: 8.0.25-0ubuntu0.20.04.1 (Ubuntu)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its

affiliates. Other names may be trademarks of their respective

owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

mysql> quit

Bye

$

Use “container-on-demand” to run MySQL

**In the container, credentials are root/root**

* Ask for container Ubuntu 20.04
* Connect via SSH
* OR connect via the Web terminal
* In the container, you should start MySQL before playing with it:

$ service mysql start

\* Starting MySQL database server mysqld

$

$ cat 0-list\_databases.sql | mysql -uroot -p

Database

information\_schema

mysql

performance\_schema

sys

$

**In the container, credentials are root/root**

How to import a SQL dump

$ echo "CREATE DATABASE hbtn\_0d\_tvshows;" | mysql -uroot -p

Enter password:

$ curl "https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level\_programming+/274/hbtn\_0d\_tvshows.sql" -s | mysql -uroot -p hbtn\_0d\_tvshows

Enter password:

$ echo "SELECT \* FROM tv\_genres" | mysql -uroot -p hbtn\_0d\_tvshows

Enter password:

id name

1 Drama

2 Mystery

3 Adventure

4 Fantasy

5 Comedy

6 Crime

7 Suspense

8 Thriller

$

Quiz questions

**Great!** You've completed the quiz successfully! Keep going! (Show quiz)

Tasks

0. My privileges!

**mandatory**

Score: 32.5% (*Checks completed: 50.0%*)

Write a script that lists all privileges of the MySQL users user\_0d\_1 and user\_0d\_2 on your server (in localhost).

guillaume@ubuntu:~/$ cat 0-privileges.sql | mysql -hlocalhost -uroot -p

Enter password:

ERROR 1141 (42000) at line 3: There is no such grant defined for user 'user\_0d\_1' on host 'localhost'

guillaume@ubuntu:~/$

guillaume@ubuntu:~/$ echo "CREATE USER 'user\_0d\_1'@'localhost';" | mysql -hlocalhost -uroot -p

Enter password:

guillaume@ubuntu:~/$ echo "GRANT ALL PRIVILEGES ON \*.\* TO 'user\_0d\_1'@'localhost';" | mysql -hlocalhost -uroot -p

Enter password:

guillaume@ubuntu:~/$ cat 0-privileges.sql | mysql -hlocalhost -uroot -p

Enter password:

Grants for user\_0d\_1@localhost

GRANT SELECT, INSERT, UPDA..., DROP ROLE ON \*.\* TO `user\_0d\_1`@`localhost`

GRANT APPLICATION\_PASSWORD\_ADMIN,AUDIT...,XA\_RECOVER\_ADMIN ON \*.\* TO `user\_0d\_1`@`localhost`

ERROR 1141 (42000) at line 4: There is no such grant defined for user 'user\_0d\_2' on host 'localhost'

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 0-privileges.sql

 Done? Help Check your code Ask for a new correction Get a sandbox QA Review

1. Root user

**mandatory**

Score: 0.0% (*Checks completed: 0.0%*)

Write a script that creates the MySQL server user user\_0d\_1.

* user\_0d\_1 should have all privileges on your MySQL server
* The user\_0d\_1 password should be set to user\_0d\_1\_pwd
* If the user user\_0d\_1 already exists, your script should not fail

guillaume@ubuntu:~/$ cat 1-create\_user.sql | mysql -hlocalhost -uroot -p

Enter password:

guillaume@ubuntu:~/$ cat 0-privileges.sql | mysql -hlocalhost -uroot -p

Enter password:

Grants for user\_0d\_1@localhost

GRANT SELECT, INSERT..., DROP ROLE ON \*.\* TO `user\_0d\_1`@`localhost`

GRANT APPLICATION\_PASSWORD\_ADMIN,...,XA\_RECOVER\_ADMIN ON \*.\* TO `user\_0d\_1`@`localhost`

ERROR 1141 (42000) at line 4: There is no such grant defined for user 'user\_0d\_2' on host 'localhost'

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 1-create\_user.sql

 Done? Help Check your code Ask for a new correction Get a sandbox QA Review

2. Read user

**mandatory**

Score: 0.0% (*Checks completed: 0.0%*)

Write a script that creates the database hbtn\_0d\_2 and the user user\_0d\_2.

* user\_0d\_2 should have only SELECT privilege in the database hbtn\_0d\_2
* The user\_0d\_2 password should be set to user\_0d\_2\_pwd
* If the database hbtn\_0d\_2 already exists, your script should not fail
* If the user user\_0d\_2 already exists, your script should not fail

guillaume@ubuntu:~/$ cat 2-create\_read\_user.sql | mysql -hlocalhost -uroot -p

Enter password:

guillaume@ubuntu:~/$ cat 0-privileges.sql | mysql -hlocalhost -uroot -p

Enter password:

Grants for user\_0d\_1@localhost

GRANT SELECT, ..., DROP ROLE ON \*.\* TO `user\_0d\_1`@`localhost`

GRANT APPLICATION\_PASSWORD\_ADMIN,...,XA\_RECOVER\_ADMIN ON \*.\* TO `user\_0d\_1`@`localhost`

Grants for user\_0d\_2@localhost

GRANT USAGE ON \*.\* TO `user\_0d\_2`@`localhost`

GRANT SELECT ON `hbtn\_0d\_2`.\* TO `user\_0d\_2`@`localhost`

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 2-create\_read\_user.sql

 Done? Help Check your code Ask for a new correction Get a sandbox QA Review

3. Always a name

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Write a script that creates the table force\_name on your MySQL server.

* force\_name description:
  + id INT
  + name VARCHAR(256) can’t be null
* The database name will be passed as an argument of the mysql command
* If the table force\_name already exists, your script should not fail

guillaume@ubuntu:~/$ cat 3-force\_name.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

guillaume@ubuntu:~/$ echo 'INSERT INTO force\_name (id, name) VALUES (89, "Best School");' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

guillaume@ubuntu:~/$ echo 'SELECT \* FROM force\_name;' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

id name

89 Best School

guillaume@ubuntu:~/$ echo 'INSERT INTO force\_name (id) VALUES (333);' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

ERROR 1364 (HY000) at line 1: Field 'name' doesn't have a default value

guillaume@ubuntu:~/$ echo 'SELECT \* FROM force\_name;' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

id name

89 Best School

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 3-force\_name.sql

 Done! Help Check your code Get a sandbox QA Review

4. ID can't be null

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Write a script that creates the table id\_not\_null on your MySQL server.

* id\_not\_null description:
  + id INT with the default value 1
  + name VARCHAR(256)
* The database name will be passed as an argument of the mysql command
* If the table id\_not\_null already exists, your script should not fail

guillaume@ubuntu:~/$ cat 4-never\_empty.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

guillaume@ubuntu:~/$ echo 'INSERT INTO id\_not\_null (id, name) VALUES (89, "Best School");' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

guillaume@ubuntu:~/$ echo 'SELECT \* FROM id\_not\_null;' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

id name

89 Best School

guillaume@ubuntu:~/$ echo 'INSERT INTO id\_not\_null (name) VALUES ("Best");' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

guillaume@ubuntu:~/$ echo 'SELECT \* FROM id\_not\_null;' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

id name

89 Best School

1 Best

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 4-never\_empty.sql

 Done! Help Check your code Get a sandbox QA Review

5. Unique ID

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Write a script that creates the table unique\_id on your MySQL server.

* unique\_id description:
  + id INT with the default value 1 and must be unique
  + name VARCHAR(256)
* The database name will be passed as an argument of the mysql command
* If the table unique\_id already exists, your script should not fail

guillaume@ubuntu:~/$ cat 5-unique\_id.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

guillaume@ubuntu:~/$ echo 'INSERT INTO unique\_id (id, name) VALUES (89, "Best School");' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

guillaume@ubuntu:~/$ echo 'SELECT \* FROM unique\_id;' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

id name

89 Best School

guillaume@ubuntu:~/$ echo 'INSERT INTO unique\_id (id, name) VALUES (89, "Best");' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

ERROR 1062 (23000) at line 1: Duplicate entry '89' for key 'unique\_id.id'

guillaume@ubuntu:~/$ echo 'SELECT \* FROM unique\_id;' | mysql -hlocalhost -uroot -p hbtn\_0d\_2

Enter password:

id name

89 Best School

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 5-unique\_id.sql

 Done! Help Check your code Get a sandbox QA Review

6. States table

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Write a script that creates the database hbtn\_0d\_usa and the table states (in the database hbtn\_0d\_usa) on your MySQL server.

* states description:
  + id INT unique, auto generated, can’t be null and is a primary key
  + name VARCHAR(256) can’t be null
* If the database hbtn\_0d\_usa already exists, your script should not fail
* If the table states already exists, your script should not fail

guillaume@ubuntu:~/$ cat 6-states.sql | mysql -hlocalhost -uroot -p

Enter password:

guillaume@ubuntu:~/$ echo 'INSERT INTO states (name) VALUES ("California"), ("Arizona"), ("Texas");' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

guillaume@ubuntu:~/$ echo 'SELECT \* FROM states;' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id name

1 California

2 Arizona

3 Texas

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 6-states.sql

 Done? Help Check your code Get a sandbox QA Review

7. Cities table

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Write a script that creates the database hbtn\_0d\_usa and the table cities (in the database hbtn\_0d\_usa) on your MySQL server.

* cities description:
  + id INT unique, auto generated, can’t be null and is a primary key
  + state\_id INT, can’t be null and must be a FOREIGN KEY that references to id of the states table
  + name VARCHAR(256) can’t be null
* If the database hbtn\_0d\_usa already exists, your script should not fail
* If the table cities already exists, your script should not fail

guillaume@ubuntu:~/$ cat 7-cities.sql | mysql -hlocalhost -uroot -p

Enter password:

guillaume@ubuntu:~/$ echo 'INSERT INTO cities (state\_id, name) VALUES (1, "San Francisco");' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

guillaume@ubuntu:~/$ echo 'SELECT \* FROM cities;' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id state\_id name

1 1 San Francisco

guillaume@ubuntu:~/$ echo 'INSERT INTO cities (state\_id, name) VALUES (10, "Paris");' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

ERROR 1452 (23000) at line 1: Cannot add or update a child row: a foreign key constraint fails (`hbtn\_0d\_usa`.`cities`, CONSTRAINT `cities\_ibfk\_1` FOREIGN KEY (`state\_id`) REFERENCES `states` (`id`))

guillaume@ubuntu:~/$ echo 'SELECT \* FROM cities;' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id state\_id name

1 1 San Francisco

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 7-cities.sql

 Done? Help Check your code Get a sandbox QA Review

8. Cities of California

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Write a script that lists all the cities of California that can be found in the database hbtn\_0d\_usa.

* The states table contains only one record where name = California (but the id can be different, as per the example)
* Results must be sorted in ascending order by cities.id
* You are not allowed to use the JOIN keyword
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ echo 'SELECT \* FROM states;' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id name

1 California

2 Arizona

3 Texas

4 Utah

guillaume@ubuntu:~/$ echo 'SELECT \* FROM cities;' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id state\_id name

1 1 San Francisco

2 1 San Jose

4 2 Page

6 3 Paris

7 3 Houston

8 3 Dallas

guillaume@ubuntu:~/$ cat 8-cities\_of\_california\_subquery.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id name

1 San Francisco

2 San Jose

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 8-cities\_of\_california\_subquery.sql

 Done? Help Check your code Get a sandbox QA Review

9. Cities by States

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Write a script that lists all cities contained in the database hbtn\_0d\_usa.

* Each record should display: cities.id - cities.name - states.name
* Results must be sorted in ascending order by cities.id
* You can use only one SELECT statement
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ echo 'SELECT \* FROM states;' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id name

1 California

2 Arizona

3 Texas

4 Utah

guillaume@ubuntu:~/$ echo 'SELECT \* FROM cities;' | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id state\_id name

1 1 San Francisco

2 1 San Jose

4 2 Page

6 3 Paris

7 3 Houston

8 3 Dallas

guillaume@ubuntu:~/$ cat 9-cities\_by\_state\_join.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_usa

Enter password:

id name name

1 San Francisco California

2 San Jose California

4 Page Arizona

6 Paris Texas

7 Houston Texas

8 Dallas Texas

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 9-cities\_by\_state\_join.sql

 Done? Help Check your code Get a sandbox QA Review

10. Genre ID by show

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Import the database dump from hbtn\_0d\_tvshows to your MySQL server: [download](https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/274/hbtn_0d_tvshows.sql)

Write a script that lists all shows contained in hbtn\_0d\_tvshows that have at least one genre linked.

* Each record should display: tv\_shows.title - tv\_show\_genres.genre\_id
* Results must be sorted in ascending order by tv\_shows.title and tv\_show\_genres.genre\_id
* You can use only one SELECT statement
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ cat 10-genre\_id\_by\_show.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_tvshows

Enter password:

title genre\_id

Breaking Bad 1

Breaking Bad 6

Breaking Bad 7

Breaking Bad 8

Dexter 1

Dexter 2

Dexter 6

Dexter 7

Dexter 8

Game of Thrones 1

Game of Thrones 3

Game of Thrones 4

House 1

House 2

New Girl 5

Silicon Valley 5

The Big Bang Theory 5

The Last Man on Earth 1

The Last Man on Earth 5

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 10-genre\_id\_by\_show.sql

 Done? Help Check your code Get a sandbox QA Review

11. Genre ID for all shows

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Import the database dump of hbtn\_0d\_tvshows to your MySQL server: [download](https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/274/hbtn_0d_tvshows.sql) (same as 10-genre\_id\_by\_show.sql)

Write a script that lists all shows contained in the database hbtn\_0d\_tvshows.

* Each record should display: tv\_shows.title - tv\_show\_genres.genre\_id
* Results must be sorted in ascending order by tv\_shows.title and tv\_show\_genres.genre\_id
* If a show doesn’t have a genre, display NULL
* You can use only one SELECT statement
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ cat 11-genre\_id\_all\_shows.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_tvshows

Enter password:

title genre\_id

Better Call Saul NULL

Breaking Bad 1

Breaking Bad 6

Breaking Bad 7

Breaking Bad 8

Dexter 1

Dexter 2

Dexter 6

Dexter 7

Dexter 8

Game of Thrones 1

Game of Thrones 3

Game of Thrones 4

Homeland NULL

House 1

House 2

New Girl 5

Silicon Valley 5

The Big Bang Theory 5

The Last Man on Earth 1

The Last Man on Earth 5

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 11-genre\_id\_all\_shows.sql

 Done? Help Check your code Get a sandbox QA Review

12. No genre

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Import the database dump from hbtn\_0d\_tvshows to your MySQL server: [download](https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/274/hbtn_0d_tvshows.sql) (same as 11-genre\_id\_all\_shows.sql)

Write a script that lists all shows contained in hbtn\_0d\_tvshows without a genre linked.

* Each record should display: tv\_shows.title - tv\_show\_genres.genre\_id
* Results must be sorted in ascending order by tv\_shows.title and tv\_show\_genres.genre\_id
* You can use only one SELECT statement
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ cat 12-no\_genre.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_tvshows

Enter password:

title genre\_id

Better Call Saul NULL

Homeland NULL

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 12-no\_genre.sql

 Done? Help Check your code Get a sandbox QA Review

13. Number of shows by genre

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Import the database dump from hbtn\_0d\_tvshows to your MySQL server: [download](https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/274/hbtn_0d_tvshows.sql) (same as 12-no\_genre.sql)

Write a script that lists all genres from hbtn\_0d\_tvshows and displays the number of shows linked to each.

* Each record should display: <TV Show genre> - <Number of shows linked to this genre>
* First column must be called genre
* Second column must be called number\_of\_shows
* Don’t display a genre that doesn’t have any shows linked
* Results must be sorted in descending order by the number of shows linked
* You can use only one SELECT statement
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ cat 13-count\_shows\_by\_genre.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_tvshows

Enter password:

genre number\_of\_shows

Drama 5

Comedy 4

Mystery 2

Crime 2

Suspense 2

Thriller 2

Adventure 1

Fantasy 1

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 13-count\_shows\_by\_genre.sql

 Done? Help Check your code Get a sandbox QA Review

14. My genres

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Import the database dump from hbtn\_0d\_tvshows to your MySQL server: [download](https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/274/hbtn_0d_tvshows.sql) (same as 13-count\_shows\_by\_genre.sql)

Write a script that uses the hbtn\_0d\_tvshows database to lists all genres of the show Dexter.

* The tv\_shows table contains only one record where title = Dexter (but the id can be different)
* Each record should display: tv\_genres.name
* Results must be sorted in ascending order by the genre name
* You can use only one SELECT statement
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ cat 14-my\_genres.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_tvshows

Enter password:

name

Crime

Drama

Mystery

Suspense

Thriller

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 14-my\_genres.sql

 Done? Help Check your code Get a sandbox QA Review

15. Only Comedy

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Import the database dump from hbtn\_0d\_tvshows to your MySQL server: [download](https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/274/hbtn_0d_tvshows.sql) (same as 14-my\_genres.sql)

Write a script that lists all Comedy shows in the database hbtn\_0d\_tvshows.

* The tv\_genres table contains only one record where name = Comedy (but the id can be different)
* Each record should display: tv\_shows.title
* Results must be sorted in ascending order by the show title
* You can use only one SELECT statement
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ cat 15-comedy\_only.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_tvshows

Enter password:

title

New Girl

Silicon Valley

The Big Bang Theory

The Last Man on Earth

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 15-comedy\_only.sql

 Done? Help Check your code Get a sandbox QA Review

16. List shows and genres

**mandatory**

Score: 65.0% (*Checks completed: 100.0%*)

Import the database dump from hbtn\_0d\_tvshows to your MySQL server: [download](https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/274/hbtn_0d_tvshows.sql) (same as 15-comedy\_only.sql)

Write a script that lists all shows, and all genres linked to that show, from the database hbtn\_0d\_tvshows.

* If a show doesn’t have a genre, display NULL in the genre column
* Each record should display: tv\_shows.title - tv\_genres.name
* Results must be sorted in ascending order by the show title and genre name
* You can use only one SELECT statement
* The database name will be passed as an argument of the mysql command

guillaume@ubuntu:~/$ cat 16-shows\_by\_genre.sql | mysql -hlocalhost -uroot -p hbtn\_0d\_tvshows

Enter password:

title name

Better Call Saul NULL

Breaking Bad Crime

Breaking Bad Drama

Breaking Bad Suspense

Breaking Bad Thriller

Dexter Crime

Dexter Drama

Dexter Mystery

Dexter Suspense

Dexter Thriller

Game of Thrones Adventure

Game of Thrones Drama

Game of Thrones Fantasy

Homeland NULL

House Drama

House Mystery

New Girl Comedy

Silicon Valley Comedy

The Big Bang Theory Comedy

The Last Man on Earth Comedy

The Last Man on Earth Drama

guillaume@ubuntu:~/$

**Repo:**

* GitHub repository: alx-higher\_level\_programming
* Directory: 0x0E-SQL\_more\_queries
* File: 16-shows\_by\_genre.sql

 Done? Help Check your code Get a sandbox QA Review

Top of Form



Bottom of Form

Copyright © 2023 ALX, All rights reserved.