Curriculum

SE Foundations

Average: **158.95%**

We're moving to Discord!

In a few days, we will be leaving Slack in favor of Discord 🎉

👆 **Click here for more information (/concepts/100033)**

0x0D. SQL - Introduction

SQL **MySQL**

By: Guillaume

Weight: 1

Project over - took place from Feb 14, 2023 6:00 AM to Feb 15, 2023 6:00 AM

☑ An auto review will be launched at the deadline

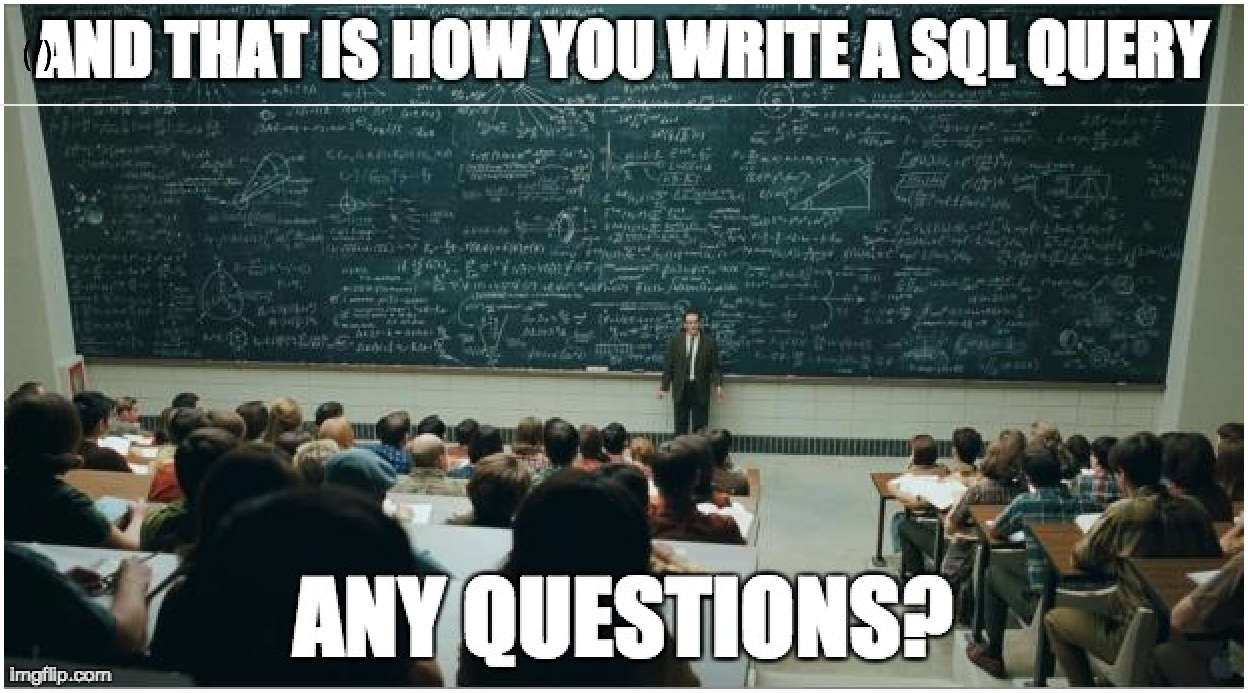
In a nutshell…

- **Auto QA review:** 104.0/104 mandatory & 24.0/24 optional
- **Altogether: 200.0%**
  - Mandatory: 100.0%
  - Optional: 100.0%
  - Calculation: 100.0% + (100.0% * 100.0%) == **200.0%**

Concepts

For this project, we expect you to look at these concepts:

- Databases (/concepts/37)
- Databases (/concepts/556)



Resources

Read or watch:

- What is Database & SQL? (/rltoken/yyRKTEdRkYEVlRgZPbasjw)
- A Basic MySQL Tutorial (/rltoken/sV2PtK5YfQsXWW1maIRZ5Q)
- Basic SQL statements: DDL and DML (/rltoken/IUKo4-UaRZSKPvXr5u9oBw) (*no need to read the chapter “Privileges”*)
- Basic queries: SQL and RA (/rltoken/rXKvu2u7vg1Hj6bnX7UgMg)
- SQL technique: functions (/rltoken/-Riv_dzSYsJyvy-LlaO6Mg)
- SQL technique: subqueries (/rltoken/QpIXoR--8eBIaidgSWYsBQ)
- What makes the big difference between a backtick and an apostrophe? (/rltoken/Gt0nFJPJRwW2Y0izzwbVrw)
- MySQL Cheat Sheet (/rltoken/1oU1LwCksQLXjs6fZYezrw)
- MySQL 8.0 SQL Statement Syntax (/rltoken/HmdmLiYBM0Q34iCYPWd9XQ)
- installing MySQL in Ubuntu 20.04 (/rltoken/IpYI9rgbwfjxOAQQgpHCmQ)

Learning Objectives

At the end of this project, you are expected to be able to explain to anyone (/rltoken/-zY4kpQMjYkkbqlEb9W37A), **without the help of Google**:

General

- What’s a database
- What’s a relational database
- What does SQL stand for
- What’s MySQL
- How to create a database in MySQL
- What does `DDL` and `DML` stand for
- How to `CREATE` or `ALTER` a table

- How to `SELECT` data from a table
- How to `INSERT` , `UPDATE` or `DELETE` data
- What are `subqueries`
- How to use MySQL functions

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`

Quiz questions

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

Tasks

0. List databases

mandatory

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

Write a script that lists all databases of your MySQL server.

```
guillaume@ubuntu:~/$ cat 0-list_databases.sql | mysql -hlocalhost -uroot -p
Enter password: 
Database
hbtn_0c_0
information_schema
mysql
performance_schema
sys
guillaume@ubuntu:~/$ 
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `0-list_databases.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

1. Create a database

mandatory

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

Write a script that creates the database `hbtn_0c_0` in your MySQL server.

- If the database `hbtn_0c_0` already exists, your script should not fail
- You are not allowed to use the `SELECT` or `SHOW` statements

```
guillaume@ubuntu:~/$ cat 1-create_database_if_missing.sql | mysql -hlocalhost -uroot
-p
Enter password:
guillaume@ubuntu:~/$ cat 0-list_databases.sql | mysql -hlocalhost -uroot -p
Enter password:
Database
information_schema
hbtn_0c_0
mysql
performance_schema
guillaume@ubuntu:~/$ cat 1-create_database_if_missing.sql | mysql -hlocalhost -uroot
-p
Enter password:
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `1-create_database_if_missing.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

2. Delete a database

mandatory

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

Write a script that deletes the database `hbtn_0c_0` in your MySQL server.

- If the database `hbtn_0c_0` doesn’t exist, your script should not fail
- You are not allowed to use the `SELECT` or `SHOW` statements

```
guillaume@ubuntu:~/$ cat 0-list_databases.sql | mysql -hlocalhost -uroot -p
Enter password:
Database
hbtn_0c_0
information_schema
mysql
performance_schema
sys
guillaume@ubuntu:~/$ cat 2-remove_database.sql | mysql -hlocalhost -uroot -p
Enter password:
guillaume@ubuntu:~/$ cat 0-list_databases.sql | mysql -hlocalhost -uroot -p
Enter password:
Database
information_schema
mysql
performance_schema
sys
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `2-remove_database.sql`

☑ Done! Help Check your code >_ Get a sandbox QA Review

3. List tables

mandatory

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

Write a script that lists all the tables of a database in your MySQL server.

- The database name will be passed as argument of `mysql` command (in the following example: `mysql` is the name of the database)

```
guillaume@ubuntu:~/$ cat 3-list_tables.sql | mysql -hlocalhost -uroot -p mysql
Enter password:
Tables_in_mysql
columns_priv
component
db
default_roles
engine_cost
func
general_log
global_grants
gtid_executed
help_category
help_keyword
help_relation
help_topic
innodb_index_stats
innodb_table_stats
password_history
plugin
procs_priv
proxies_priv
replication_asynchronous_connection_failover
replication_asynchronous_connection_failover_managed
role_edges
server_cost
servers
slave_master_info
slave_relay_log_info
slave_worker_info
slow_log
tables_priv
time_zone
time_zone_leap_second
time_zone_name
time_zone_transition
time_zone_transition_type
user
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `3-list_tables.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

4(/)First table

mandatory

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

Write a script that creates a table called `first_table` in the current database in your MySQL server.

- `first_table` description:
    - `id` INT
    - `name` VARCHAR(256)
- The database name will be passed as an argument of the `mysql` command
- If the table `first_table` already exists, your script should not fail
- You are not allowed to use the `SELECT` or `SHOW` statements

```
guillaume@ubuntu:~/$ cat 4-first_table.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
guillaume@ubuntu:~/$ cat 3-list_tables.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
Tables_in_hbtn_0c_0
first_table
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `4-first_table.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

5. Full description

mandatory

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

Write a script that prints the full description of the table `first_table` from the database `hbtn_0c_0` in your MySQL server.

- The database name will be passed as an argument of the `mysql` command
- You are not allowed to use the `DESCRIBE` or `EXPLAIN` statements

```
guillaume@ubuntu:~/$ cat 5-full_table.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
Table   Create Table
first_table     CREATE TABLE `first_table` (\n  `id` int DEFAULT NULL,\n  `name` var
char(256) DEFAULT NULL\n) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900
_ai_ci
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `5-full_table.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

6. List all in table

mandatory

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

Write a script that lists all rows of the table `first_table` from the database `hbtn_0c_0` in your MySQL server.

- All fields should be printed
- The database name will be passed as an argument of the `mysql` command

```
guillaume@ubuntu:~/$ cat 6-list_values.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `6-list_values.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

7. First add

mandatory

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

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

(/)

Write a script that inserts a new row in the table `first_table` (database `hbtn_0c_0`) in your MySQL server.

- New row:
  - `id = 89`
  - `name = Best School`
- The database name will be passed as an argument of the `mysql` command

```
guillaume@ubuntu:~/$ cat 7-insert_value.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
guillaume@ubuntu:~/$ cat 6-list_values.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
id  name
89  Best School
guillaume@ubuntu:~/$ cat 7-insert_value.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
guillaume@ubuntu:~/$ cat 7-insert_value.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
guillaume@ubuntu:~/$ cat 6-list_values.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
id  name
89  Best School
89  Best School
89  Best School
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `7-insert_value.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

8. Count 89

mandatory

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

Write a script that displays the number of records with `id = 89` in the table `first_table` of the database `hbtn_0c_0` in your MySQL server.

- The database name will be passed as an argument of the `mysql` command

```
guillaume@ubuntu:~/$ cat 8-count_89.sql | mysql -hlocalhost -uroot -p hbtn_0c_0 | tail -1
Enter password:
3
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `8-count_89.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

9. Full creation

mandatory

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

Write a script that creates a table `second_table` in the database `hbtn_0c_0` in your MySQL server and add multiples rows.

- `second_table` description:
  - `id` INT
  - `name` VARCHAR(256)
  - `score` INT
- The database name will be passed as an argument to the `mysql` command
- If the table `second_table` already exists, your script should not fail
- You are not allowed to use the `SELECT` and `SHOW` statements
- Your script should create these records:
  - `id` = 1, `name` = “John”, `score` = 10
  - `id` = 2, `name` = “Alex”, `score` = 3
  - `id` = 3, `name` = “Bob”, `score` = 14
  - `id` = 4, `name` = “George”, `score` = 8

```
guillaume@ubuntu:~/$ cat 9-full_creation.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `9-full_creation.sql`

☑ Done! Help Check your code Get a sandbox QA Review

10. List by best

mandatory

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

Write a script that lists all records of the table `second_table` of the database `hbtn_0c_0` in your MySQL server.

- Results should display both the score and the name (in this order)
- Records should be ordered by score (top first)
- The database name will be passed as an argument of the `mysql` command

```
guillaume@ubuntu:~/$ cat 10-top_score.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password: 
score   name
14  Bob
10  John
8   George
3   Alex
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `10-top_score.sql`

☑ Done! Help Check your code Get a sandbox QA Review

11. Select the best

mandatory

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

Write a script that lists all records with a `score >= 10` in the table `second_table` of the database `hbtn_0c_0` in your MySQL server.

- Results should display both the score and the name (in this order)
- Records should be ordered by score (top first)
- The database name will be passed as an argument of the `mysql` command

```
guillaume@ubuntu:~/$ cat 11-best_score.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
score   name
14  Bob
10  John
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `11-best_score.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

12. Cheating is bad

mandatory

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

Write a script that updates the score of Bob to `10` in the table `second_table`.

- You are not allowed to use Bob's id value, only the `name` field
- The database name will be passed as an argument of the `mysql` command

```
guillaume@ubuntu:~/$ cat 12-no_cheating.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
guillaume@ubuntu:~/$ cat 10-top_score.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
score   name
10  John
10  Bob
8   George
3   Alex
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `12-no_cheating.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

13. Score too low

mandatory

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

Write a script that removes all records with a `score <= 5` in the table `second_table` of the database `hbtn_0c_0` in your MySQL server.

- The database name will be passed as an argument of the `mysql` command

```
guillaume@ubuntu:~/$ cat 13-change_class.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password: 
guillaume@ubuntu:~/$ cat 10-top_score.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password: 
score   name
10  John
10  Bob
8   George
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `13-change_class.sql`

☑ Done! Help Check your code >_ Get a sandbox QA Review

14. Average

mandatory

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

Write a script that computes the score average of all records in the table `second_table` of the database `hbtn_0c_0` in your MySQL server.

- The result column name should be `average`
- The database name will be passed as an argument of the `mysql` command

```
guillaume@ubuntu:~/$ cat 14-average.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password: 
average
9.3333
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `14-average.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

15. Number by score

mandatory

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

Write a script that lists the number of records with the same score in the table `second_table` of the database `hbtn_0c_0` in your MySQL server.

- The result should display:
  - the `score`
  - the number of records for this `score` with the label `number`
- The list should be sorted by the number of records (descending)
- The database name will be passed as an argument to the `mysql` command

```
guillaume@ubuntu:~/$ cat 15-groups.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
score   number
10  2
8   1
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `15-groups.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

16. Say my name

mandatory

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

Write a script that lists all records of the table `second_table` of the database `hbtn_0c_0` in your MySQL server.

- Don’t list rows without a `name` value

(/)

- Results should display the score and the name (in this order)
- Records should be listed by descending score
- The database name will be passed as an argument to the `mysql` command

In this example, new data have been added to the table `second_table`.

```
guillaume@ubuntu:~/$ cat 16-no_link.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password: 
score   name
18  Aria
12  Aria
10  John
10  Bob
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `16-no_link.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

17. Go to UTF8

#advanced

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

Write a script that converts `hbtn_0c_0` database to UTF8 (`utf8mb4`, collate `utf8mb4_unicode_ci`) in your MySQL server.

You need to convert all of the following to `UTF8`:

- Database `hbtn_0c_0`
- Table `first_table`
- Field `name` in `first_table`

```
guillaume@ubuntu:~/$ cat 100-move_to_utf8.sql | mysql -hlocalhost -uroot -p
Enter password: 
guillaume@ubuntu:~/$ cat 5-full_table.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password: 
Table   Create Table
first_table CREATE TABLE `first_table` (\n  `id` int(11) DEFAULT NULL,\n  `name` var
char(256) COLLATE utf8mb4_unicode_ci DEFAULT NULL\n) ENGINE=InnoDB DEFAULT CHARSET=u
tf8mb4 COLLATE=utf8mb4_unicode_ci
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `100-move_to_utf8.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

18. Temperatures #0

#advanced

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

Import in `hbtn_0c_0` database this table dump: download (https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/272/temperatures.sql)

Write a script that displays the average temperature (Fahrenheit) by city ordered by temperature (descending).

```
guillaume@ubuntu:~/$ cat 101-avg_temperatures.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password: 
city    avg_temp
Chandler    72.8627
Gilbert 71.8088
Pismo beach 71.5147
San Francisco   71.4804
Sedona  70.7696
Phoenix 70.5882
Oakland 70.5637
Sunnyvale   70.5245
Chicago 70.4461
San Diego   70.1373
Glendale    70.1225
Sonoma  70.0392
Yuma    69.3873
San Jose    69.2990
Tucson  69.0245
Joliet  68.6716
Naperville  68.1029
Tempe   67.0441
Peoria  66.5392
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `101-avg_temperatures.sql`

☑ Done! Help Check your code Get a sandbox QA Review

19. Temperatures #1

#advanced

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

Import in `hbtn_0c_0` database this table dump: download (https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/272/temperatures.sql) (same as `Temperatures #0`)

Write a script that displays the top 3 of cities temperature during July and August ordered by temperature (descending)

```
guillaume@ubuntu:~/$ cat 102-top_city.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
city    avg_temp
Naperville  76.9412
San Diego   73.7941
Sunnyvale   73.2353
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `102-top_city.sql`

☑ Done! Help Check your code Get a sandbox QA Review

20. Temperatures #2

#advanced

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

Import in `hbtn_0c_0` database this table dump: download (https://s3.amazonaws.com/intranet-projects-files/holbertonschool-higher-level_programming+/272/temperatures.sql) (same as `Temperatures #0`)

Write a script that displays the max temperature of each state (ordered by State name).

```
guillaume@ubuntu:~/$ cat 103-max_state.sql | mysql -hlocalhost -uroot -p hbtn_0c_0
Enter password:
state   max_temp
AZ  110
CA  110
IL  110
guillaume@ubuntu:~/$
```

Repo:

- GitHub repository: `alx-higher_level_programming`
- Directory: `0x0D-SQL_introduction`
- File: `103-max_state.sql`

☑ Done! | Help | Check your code | >_ Get a sandbox | QA Review

Copyright © 2023 ALX, All rights reserved.