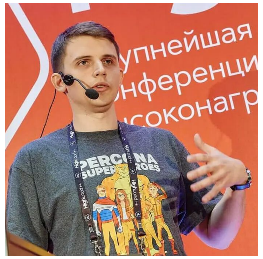
MySQL Up and Running in 30 minutes

Sergey Kuzmichev
Vinicius Grippa

Who we are



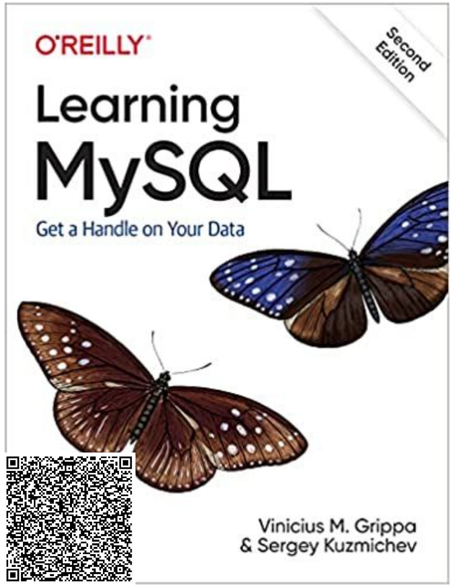
Sergey Kuzmichev

Databases, performance, reliability. Infrastructure and Performance Architect at Investing.com. We are hiring.

Linkedin: https://www.linkedin.com/in/skuzmichev
Telegram: @arronax
GitHub: https://github.com/arronax

Vinicius Grippa

Senior Database Engineer at Percona.
Linkedin: https://www.linkedin.com/in/vinicius-grippa/


O'REILLY
Second Edition
Learning MySQL
Get a Handle on Your Data
Vinicius M. Grippa & Sergey Kuzmichev

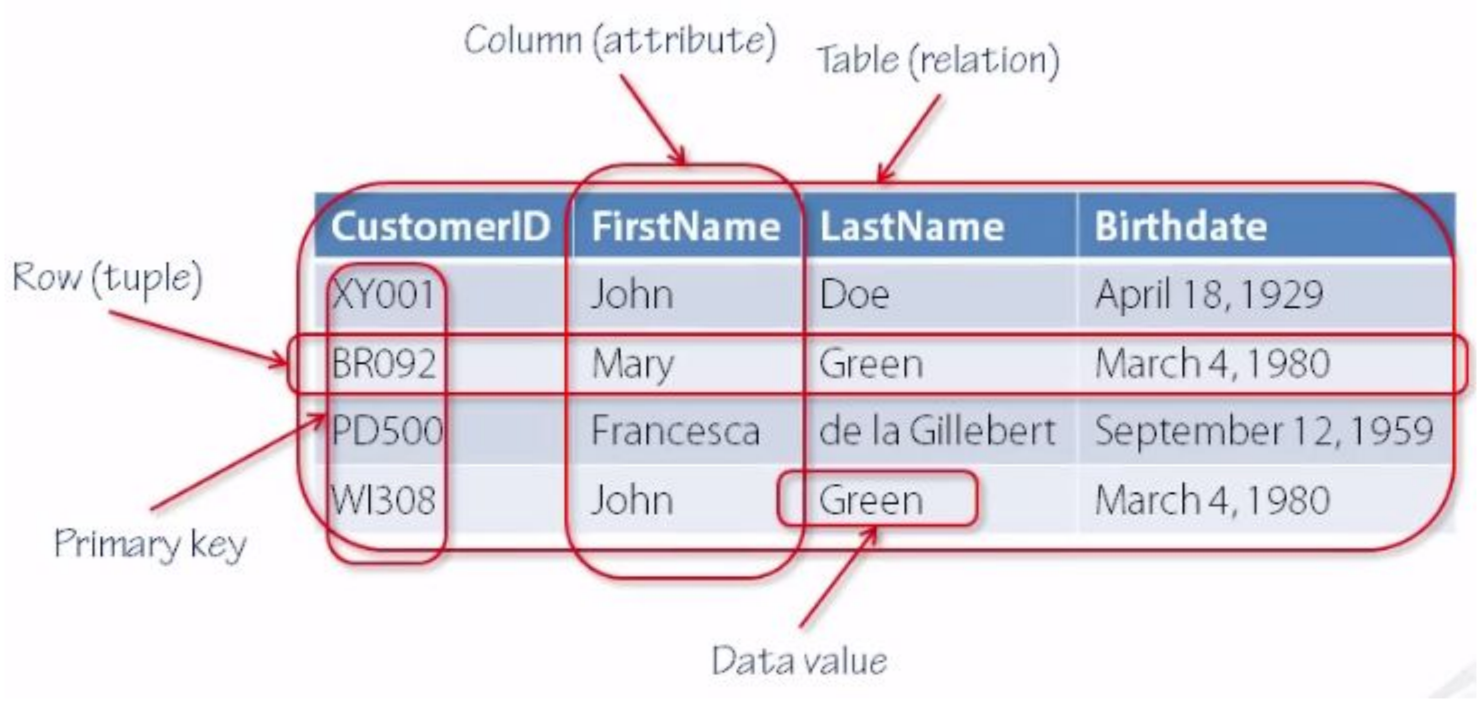
Agenda

- What is a relational database management system (RDBMS)? How different it is from a NoSQL database?
- Is SQL useful for Data Science?
- Deploying a MySQL instance
- MySQL GUI tools
- Dataset examples

What is a relational database management system (RDBMS)?

- Data is organized in tables, where columns represent attributes and rows represent records.
- Relation in RDBMS is a table, no the FK or another constraint.
- Schema is strict and omnipresent. Each table is defined to hold specific data.
- Data is traditionally accessed mainly via the SQL.
- SQL stands for Structured Query Language, a language used to interact with a relational database. It can be used to read and write (create, update, delete) data in a Relational Database Management System (RDBMS).

What is a relational database management system (RDBMS)?



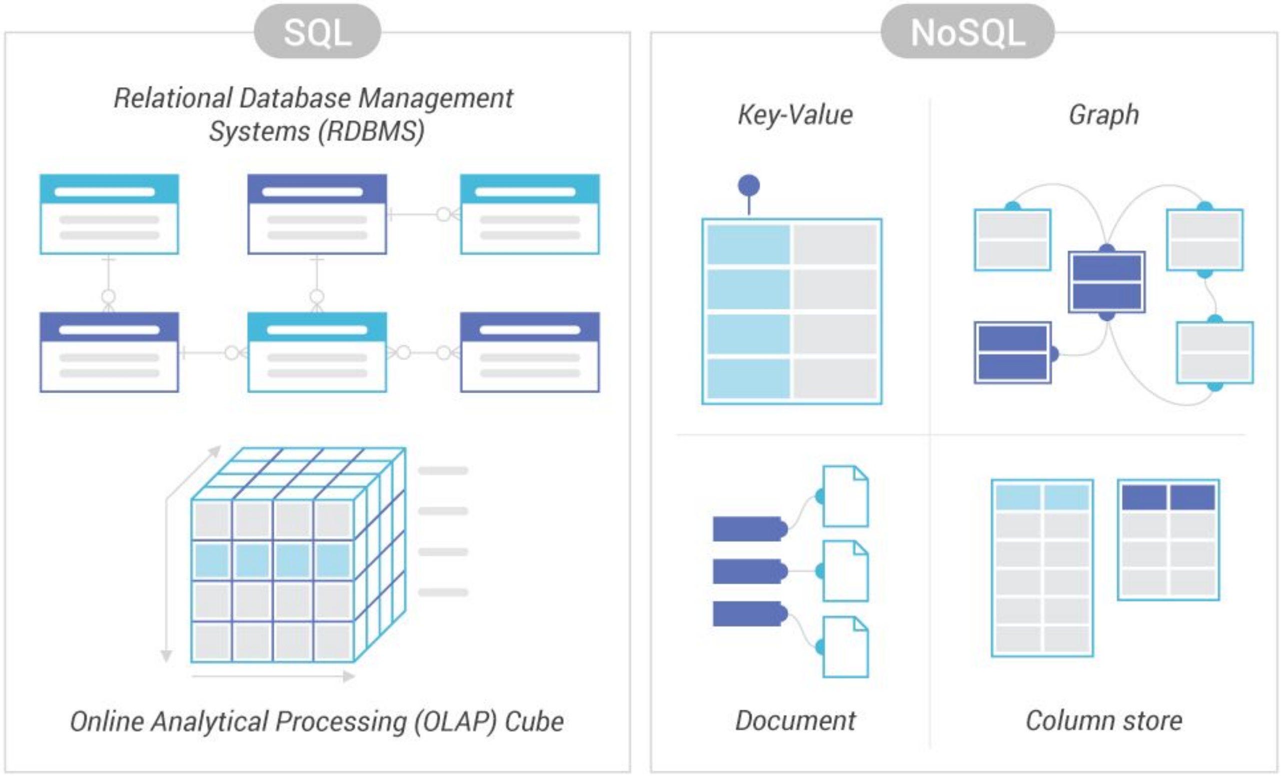
CustomerID	FirstName	LastName	Birthdate
XY001	John	Doe	April 18, 1929
BR092	Mary	Green	March 4, 1980
PD500	Francesca	de la Gillebert	September 12, 1959
WI308	John	Green	March 4, 1980

Source: *https://medium.com/@oliverknocklein/visualizing-sql-a-beginners-guide-to-relational-databases-c2dcfda79ea4*

How different it is from a NoSQL database?

- While not a mandatory trait, most NoSQL databases will have no schema or a relaxed schema.
- Constraints and table dependencies come naturally in an RDBMS, but they do not define the RDBMS.
- NoSQL can store different types of data.

How different it is from a NoSQL database?

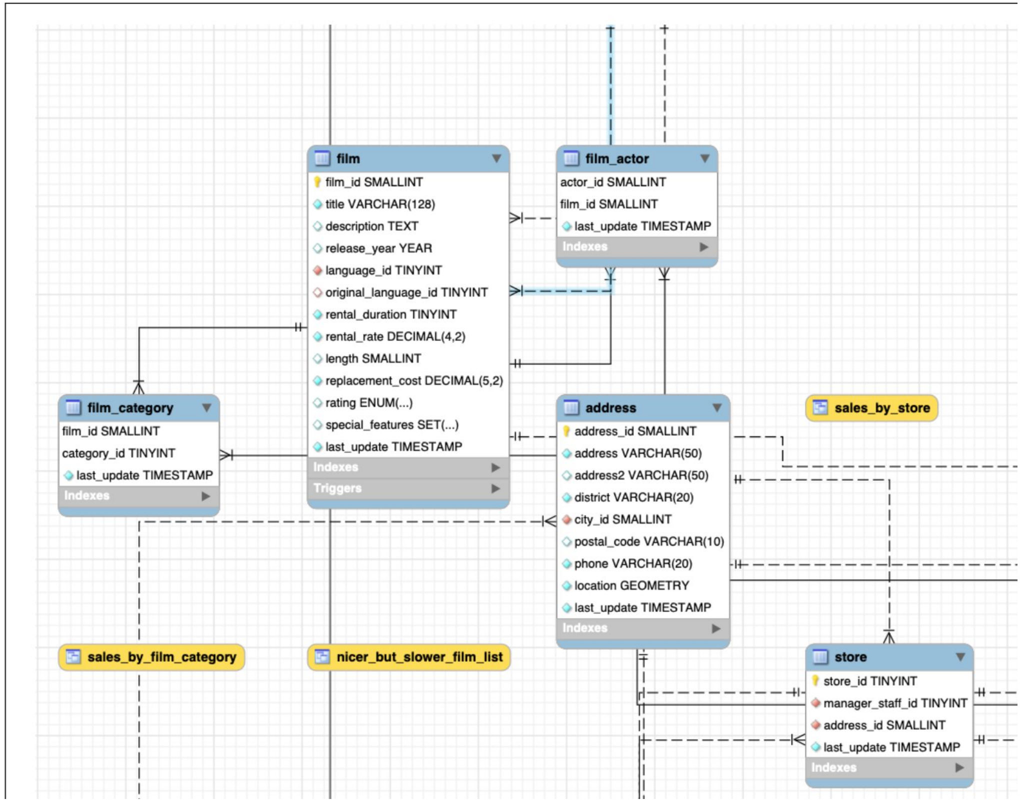

SQL
Relational Database Management Systems (RDBMS)
Online Analytical Processing (OLAP) Cube
NoSQL
Key-Value
Graph
Document
Column store

Source: https://www.scylladb.com/learn/nosql/nosql-vs-sql/

Is SQL useful for Data Science?

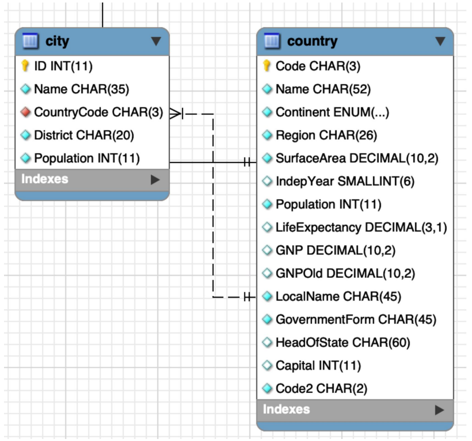
- SQL remains the ideal choice for many CRM, business intelligence tools, finance and in office operations.
- A Data Scientist needs SQL in order to handle structured data (extract, manipulate and transform). This structured data is stored in relational databases.

Is SQL useful for Data Science?



datasciencedojo
data science for everyone

Structured Query Language (SQL)



```
SELECT Name FROM country;

SELECT Name FROM city;

SELECT city.Name, country.Name FROM country INNER JOIN city ON
CountryCode=Code;
```

Deploying a MySQL instance

datasciencedojo
data science for everyone

Deploying a MySQL instance

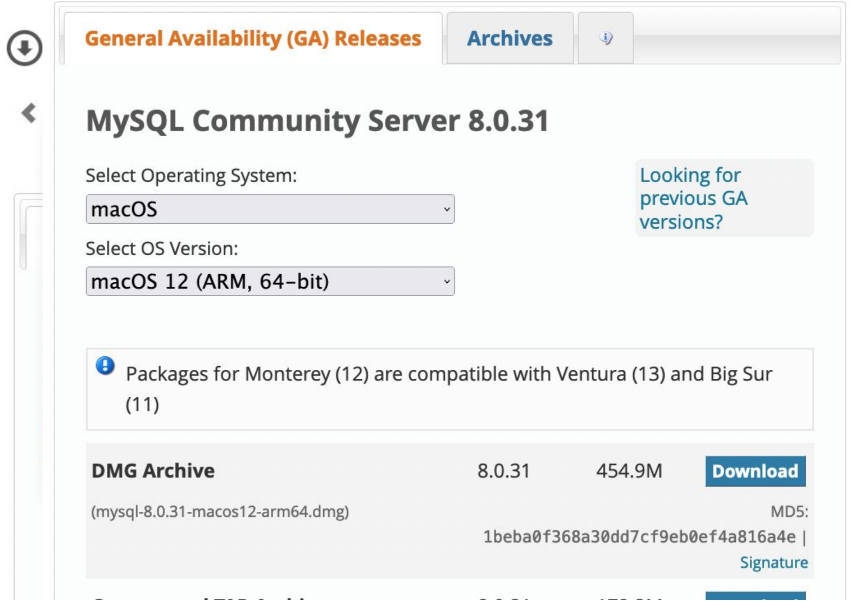
- There are many ways to deploy or install a database.
- It is possible to install in a server, virtualized instances (EC2, Compute Engine, Virtualbox).
- Or more recently ways as docker, LXC, Kubernetes, podman.
- In both options, you can automate the deployment (Terraform, Ansible, Vagrant, …)
- And more recently, DBaaS (database-as-a-service).

Deploying a MySQL instance

DEMO

Deploying MySQL locally

- MySQL Community Server installers or packages are available for most OSes
- MacOS and Windows have graphical installers



MySQL Community Server 8.0.31

Select Operating System: macOS

Select OS Version: macOS 12 (ARM, 64-bit)

Looking for previous GA versions?

Packages for Monterey (12) are compatible with Ventura (13) and Big Sur (11)

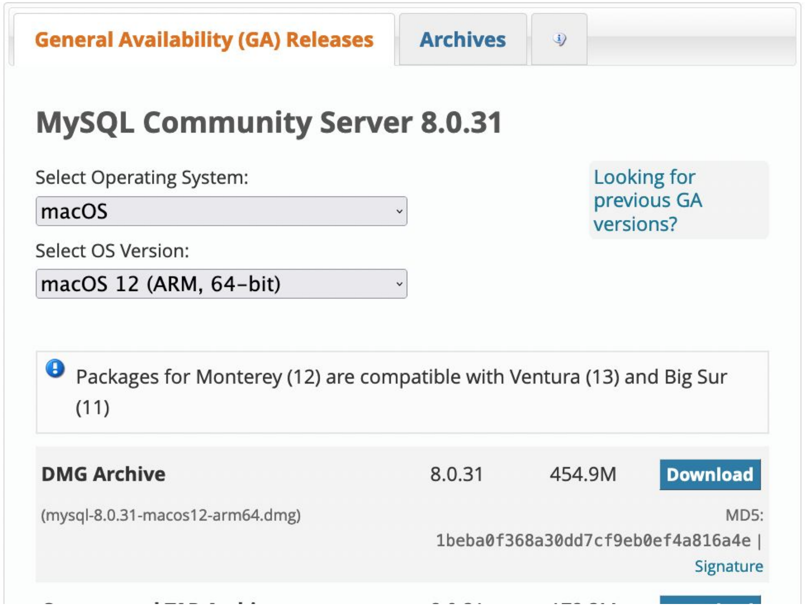
DMG Archive	8.0.31	454.9M	Download
(mysql-8.0.31-macos12-arm64.dmg)			MD5: 1beba0f368a30dd7cf9eb0ef4a816a4e | Signature

Deploying MySQL locally – MacOS

- Windows installation is similar to one outlined here
- Dedicated installer for Windows

Deploying MySQL locally – macOS

- Pick macOS as the OS
- Pick your CPU (Intel is x86, Apple Silicon is ARM)
- Get DMG


General Availability (GA) Releases
Archives
Looking for previous GA versions?
Download
Signature

MySQL Community Server 8.0.31

Select Operating System: macOS

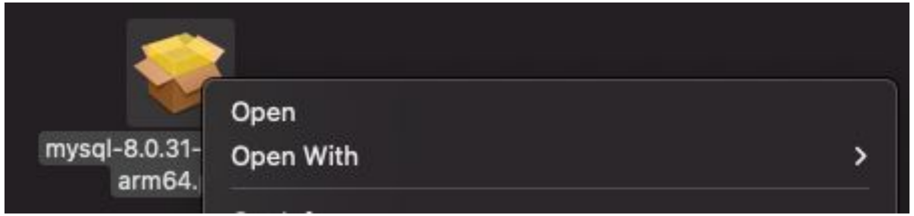
Select OS Version: macOS 12 (ARM, 64-bit)

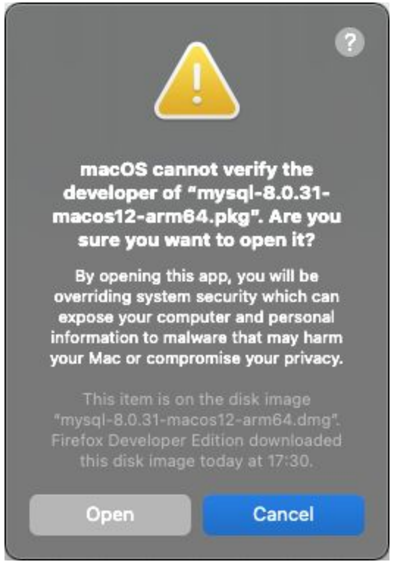
Packages for Monterey (12) are compatible with Ventura (13) and Big Sur (11)

DMG Archive	8.0.31	454.9M	Download
(mysql-8.0.31-macos12-arm64.dmg)			MD5: 1beba0f368a30dd7cf9eb0ef4a816a4e | Signature

Deploying MySQL locally – MacOS

- Bypass developer verification using forbidden knowledge
- Right mouse click (double-click on touchpad, or ctrl+click a single mouse button)





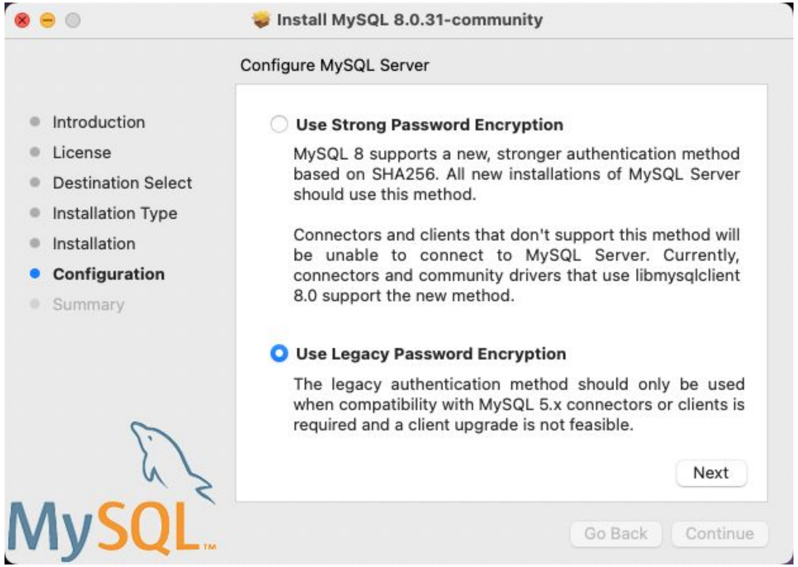
macOS cannot verify the developer of "mysql-8.0.31-macos12-arm64.pkg". Are you sure you want to open it?

By opening this app, you will be overriding system security which can expose your computer and personal information to malware that may harm your Mac or compromise your privacy.

This item is on the disk image "mysql-8.0.31-macos12-arm64.dmg". Firefox Developer Edition downloaded this disk image today at 17:30.

Deploying MySQL locally – MacOS

- Press next and continue everywhere, it's that simple
- We recommend using Legacy Password (only for local envs, not prod!)


Install MySQL 8.0.31-community

Configure MySQL Server

( ) **Use Strong Password Encryption**

MySQL 8 supports a new, stronger authentication method based on SHA256. All new installations of MySQL Server should use this method.

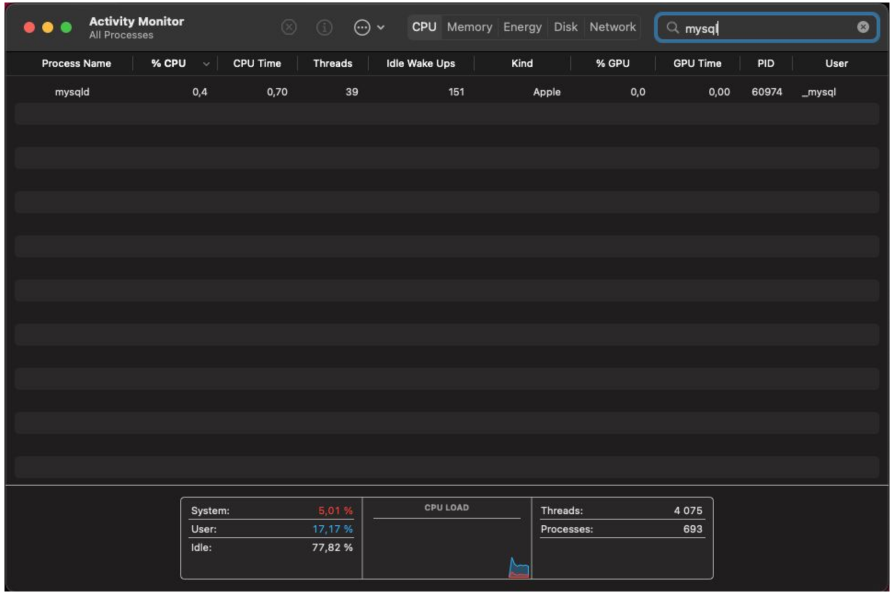
Connectors and clients that don't support this method will be unable to connect to MySQL Server. Currently, connectors and community drivers that use libmysqlclient 8.0 support the new method.

[✓ **Use Legacy Password Encryption**]

The legacy authentication method should only be used when compatibility with MySQL 5.x connectors or clients is required and a client upgrade is not feasible.

Deploying MySQL locally – MacOS

- mysqld process – MySQL Daemon – is running

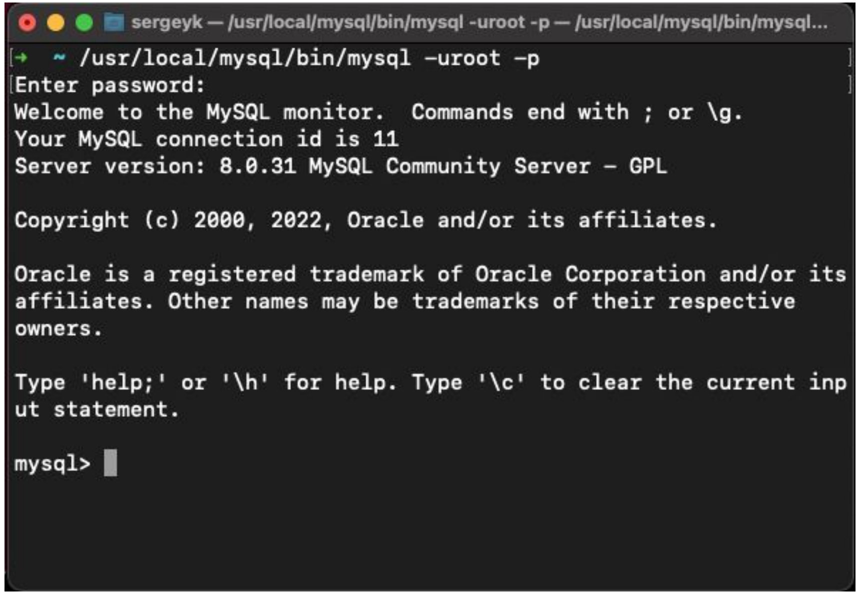


Process Name	% CPU	CPU Time	Threads	Idle Wake Ups	Kind	% GPU	GPU Time	PID	User
mysqld	0,4	0,70	39	151	Apple	0,0	0,00	60974	_mysql

System:	5,01 %
User:	17,17 %
Idle:	77,82 %
Threads:	4 075
Processes:	693

Deploying MySQL locally – MacOS

- MySQL CLI can be used to verify the installation

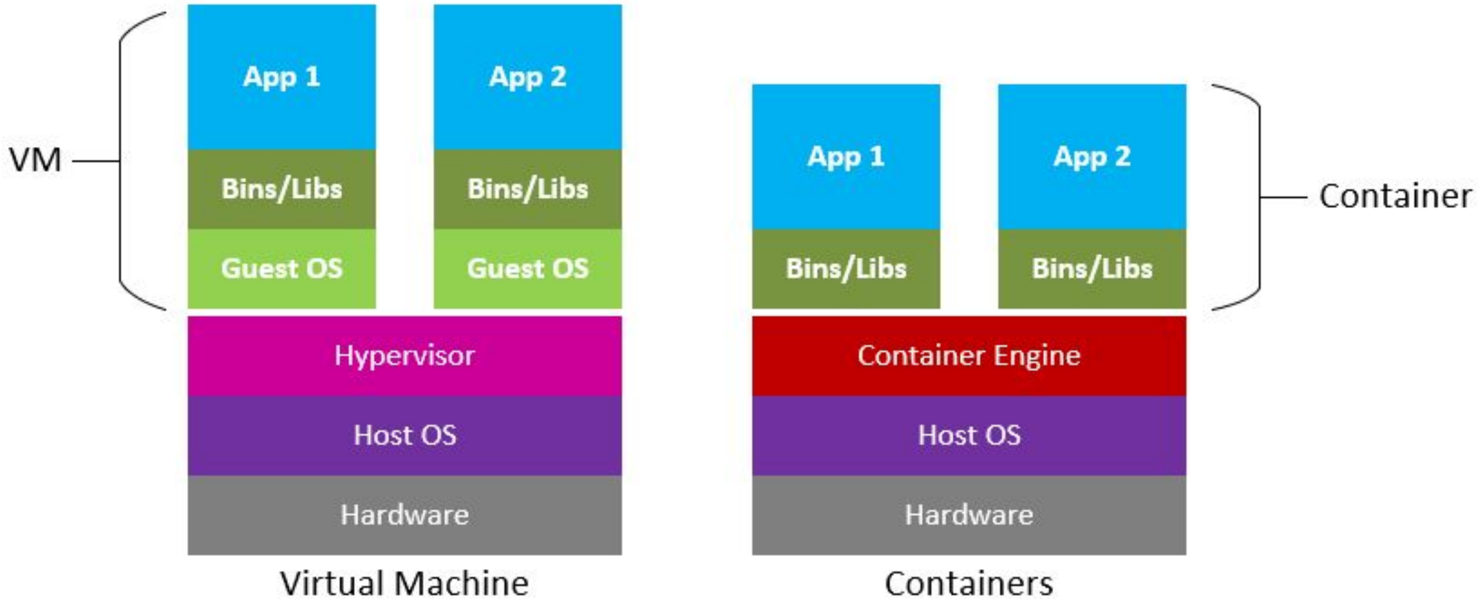


sergeyk — /usr/local/mysql/bin/mysql -uroot -p — /usr/local/mysql/bin/mysql...

```
→  ~ /usr/local/mysql/bin/mysql -uroot -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 11
Server version: 8.0.31 MySQL Community Server - GPL

Copyright (c) 2000, 2022, 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 inp
ut statement.

mysql> 
```

Deploying MySQL with Docker

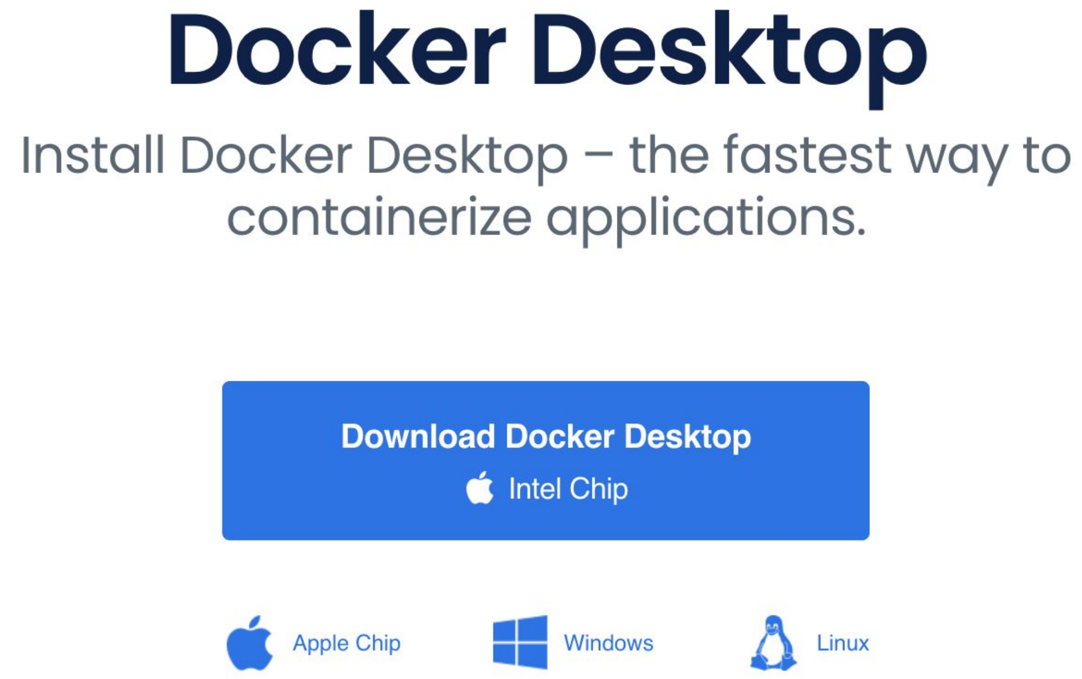
- Docker uses OS-level virtualization to deliver software in containers. It is possible to have many containers running in the same host. It is usually faster to deploy than a VM.



Deploying MySQL with Docker

DEMO

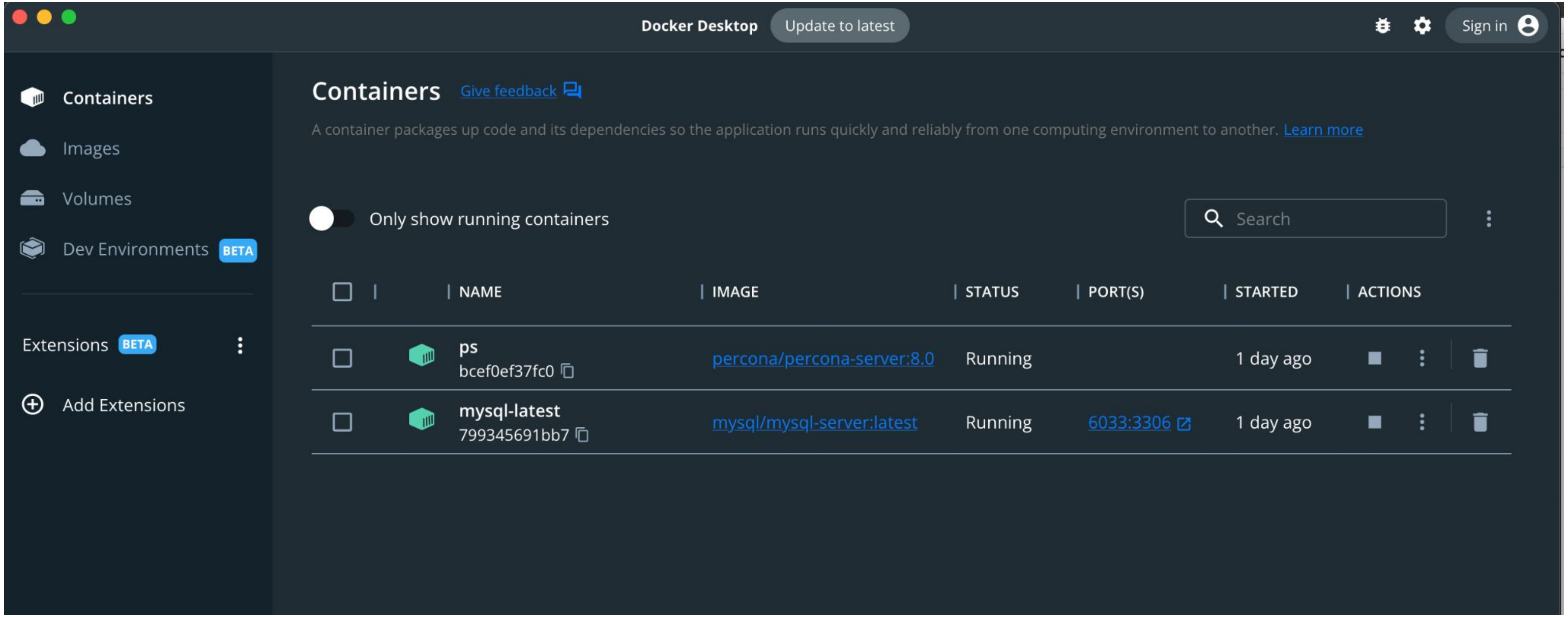
Deploying MySQL with Docker



Docker Desktop

Install Docker Desktop – the fastest way to containerize applications.

Deploying MySQL with Docker



Docker Desktop Update to latest Sign in

Containers Give feedback

A container packages up code and its dependencies so the application runs quickly and reliably from one computing environment to another. Learn more

Only show running containers

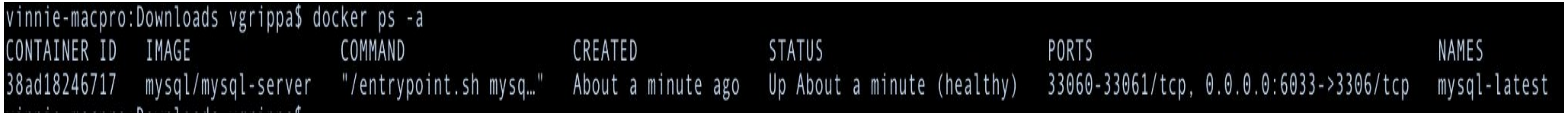
Search

NAME	IMAGE	STATUS	PORT(S)	STARTED	ACTIONS
ps bcef0ef37fc0	percona/percona-server:8.0	Running		1 day ago	
mysql-latest 799345691bb7	mysql/mysql-server:latest	Running	6033:3306	1 day ago	

Deploying MySQL with Docker

```
$ docker run --name mysql-latest -p 6033:3306 -e
MYSQL_ROOT_PASSWORD='learning_mysql' -d mysql/mysql-server:latest

$ docker exec -ti mysql-latest mysql -uroot -plearning_mysql
```

Deploying MySQL with Docker

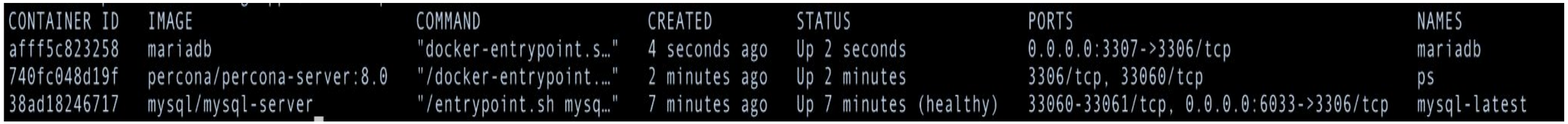


```
vinnie-macpro:Downloads vgrippa$ docker ps -a
CONTAINER ID   IMAGE               COMMAND                  CREATED              STATUS                        PORTS                                NAMES
38ad18246717   mysql/mysql-server  "/entrypoint.sh mysq…"   About a minute ago   Up About a minute (healthy)   33060-33061/tcp, 0.0.0.0:6033->3306/tcp   mysql-latest
```

Deploying Percona Server/MariaDB with Docker

```
$ docker run -d --name ps -e MYSQL_ROOT_PASSWORD=root   percona/percona-server:8.0

$ docker run --name mariadb -p 3307:3306  -e MYSQL_ROOT_PASSWORD=password -d mariadb
```

Deploying MySQL with Docker



```
CONTAINER ID   IMAGE                        COMMAND                   CREATED          STATUS                    PORTS                                         NAMES
afff5c823258   mariadb                      "docker-entrypoint.s…"   4 seconds ago    Up 2 seconds              0.0.0.0:3307->3306/tcp                        mariadb
740fc048d19f   percona/percona-server:8.0   "/docker-entrypoint.…"   2 minutes ago    Up 2 minutes              3306/tcp, 33060/tcp                           ps
38ad18246717   mysql/mysql-server           "/entrypoint.sh mysq…"   7 minutes ago    Up 7 minutes (healthy)    33060-33061/tcp, 0.0.0.0:6033->3306/tcp       mysql-latest
```

Interfacing with MySQL – CLI

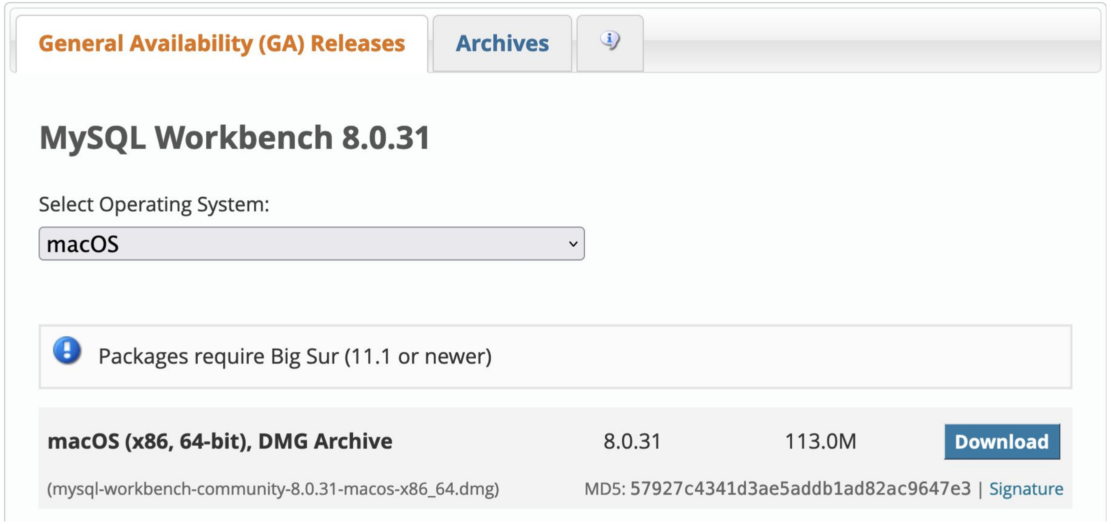
- Convenient on a server
- Probably not the best way to learn

Interfacing with MySQL – CLI

DEMO

Interfacing with MySQL – MySQL Workbench

- Similar to the MySQL Server itself, Oracle provides Workbench installers
- x86 app works on ARM Macs



MySQL Workbench 8.0.31

Select Operating System:
macOS

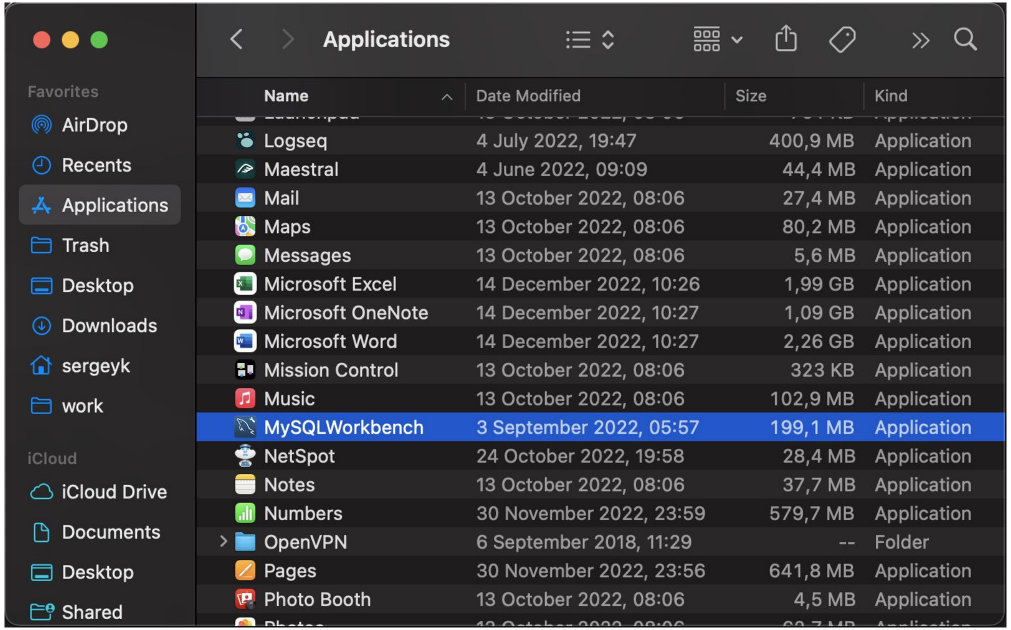
Packages require Big Sur (11.1 or newer)

macOS (x86, 64-bit), DMG Archive	8.0.31	113.0M	Download
(mysql-workbench-community-8.0.31-macos-x86_64.dmg)	MD5: 57927c4341d3ae5addb1ad82ac9647e3 | Signature		

datasciencedojo
data science for everyone

Interfacing with MySQL – MySQL Workbench

- After following the standard dmg installation, find Workbench in Apps list

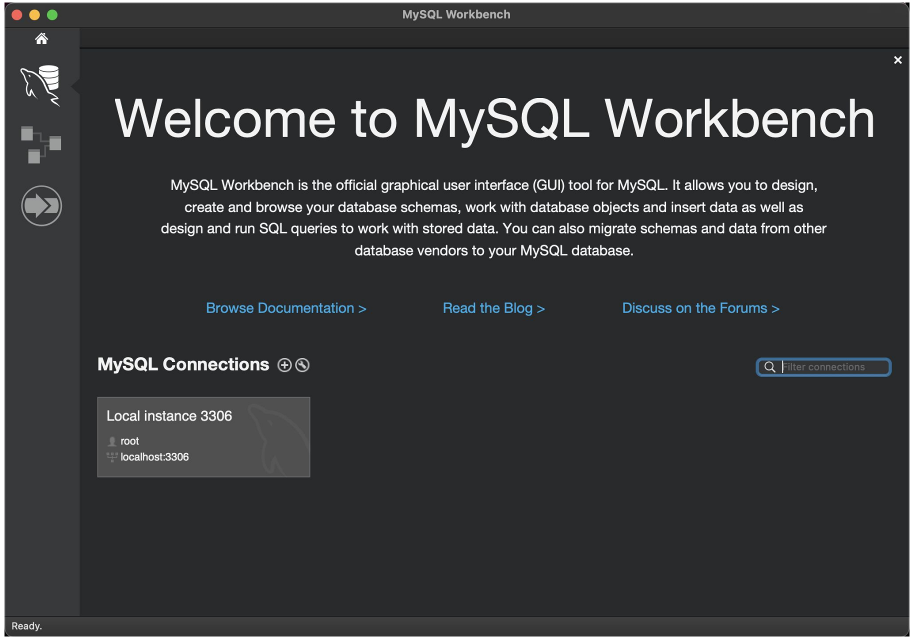


Name	Date Modified	Size	Kind
Logseq	4 July 2022, 19:47	400,9 MB	Application
Maestral	4 June 2022, 09:09	44,4 MB	Application
Mail	13 October 2022, 08:06	27,4 MB	Application
Maps	13 October 2022, 08:06	80,2 MB	Application
Messages	13 October 2022, 08:06	5,6 MB	Application
Microsoft Excel	14 December 2022, 10:26	1,99 GB	Application
Microsoft OneNote	14 December 2022, 10:27	1,09 GB	Application
Microsoft Word	14 December 2022, 10:27	2,26 GB	Application
Mission Control	13 October 2022, 08:06	323 KB	Application
Music	13 October 2022, 08:06	102,9 MB	Application
MySQLWorkbench	3 September 2022, 05:57	199,1 MB	Application
NetSpot	24 October 2022, 19:58	28,4 MB	Application
Notes	13 October 2022, 08:06	37,7 MB	Application
Numbers	30 November 2022, 23:59	579,7 MB	Application
OpenVPN	6 September 2018, 11:29	--	Folder
Pages	30 November 2022, 23:56	641,8 MB	Application
Photo Booth	13 October 2022, 08:06	4,5 MB	Application

datasciencedojo
data science for everyone

Interfacing with MySQL – MySQL Workbench

- Default Workbench startup screen has a Local connection


MySQL Workbench

Welcome to MySQL Workbench

MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.

Browse Documentation > Read the Blog > Discuss on the Forums >

MySQL Connections

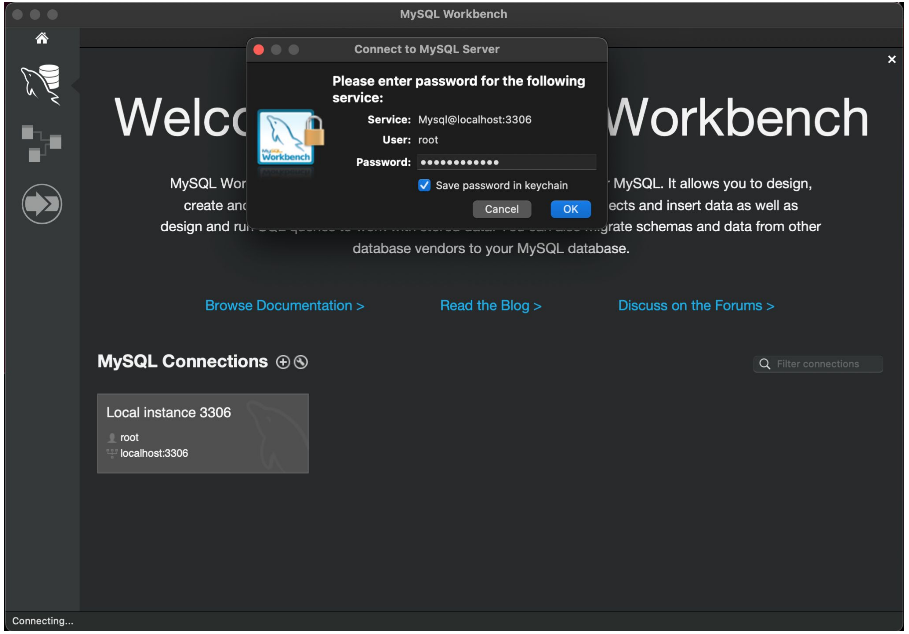
Filter connections

Local instance 3306
root
localhost:3306

Ready.

Interfacing with MySQL – MySQL Workbench

- Hopefully, you remember the password you set when installing MySQL



MySQL Workbench

Connect to MySQL Server

Please enter password for the following service:

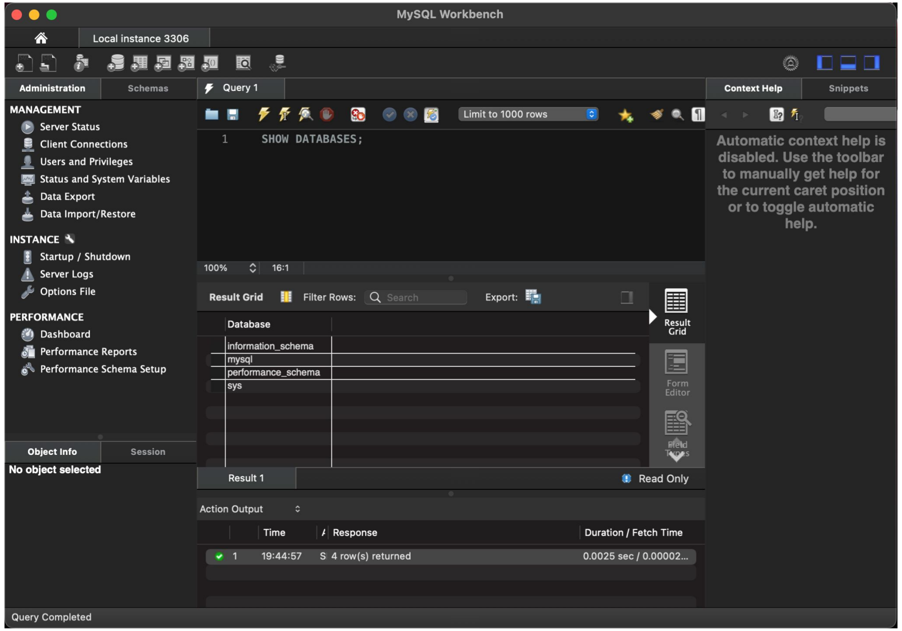
Service: Mysql@localhost:3306
User: root
Password: ●●●●●●●●●●●●

☑ Save password in keychain

datasciencedojo
data science for everyone

Interfacing with MySQL – MySQL Workbench

- By default, a fresh MySQL installation has no data



```
SHOW DATABASES;
```

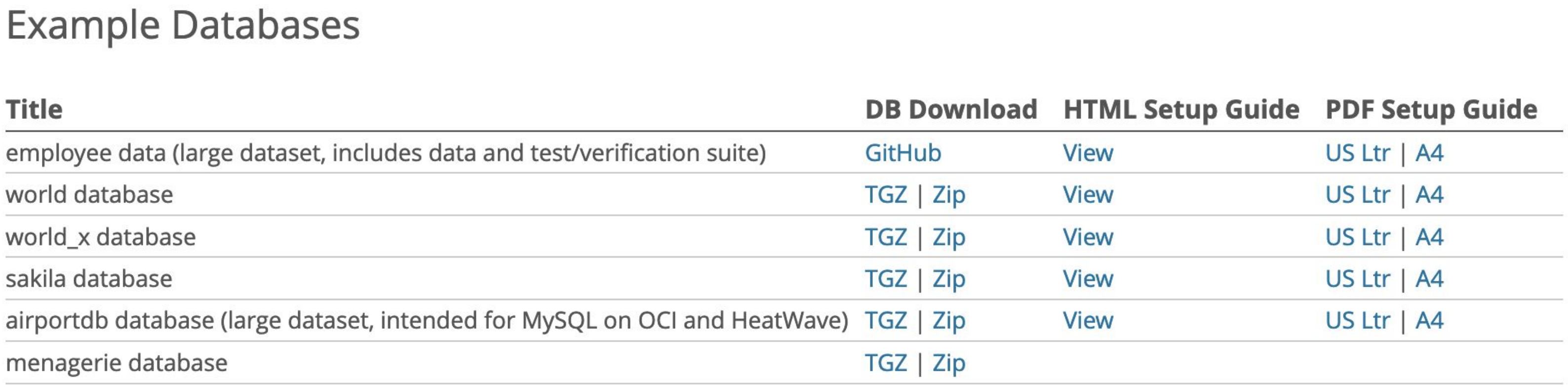
```
Database
information_schema
mysql
performance_schema
sys
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```
   Time      Response           Duration / Fetch Time
1  19:44:57  4 row(s) returned  0.0025 sec / 0.00002...
```

Demo Datasets and where to find them

- By default, a fresh MySQL installation has no data
- There are many datasets available online
- Here are some from Oracle itself
- We will show Sakila and Employee databases



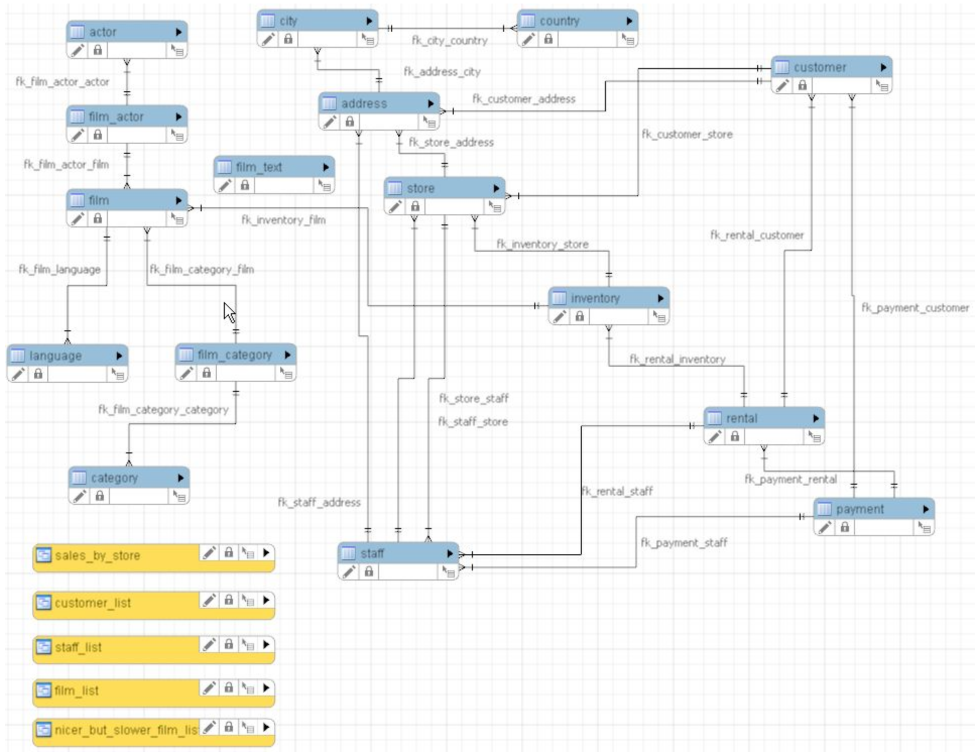
Example Databases

Title	DB Download	HTML Setup Guide	PDF Setup Guide
employee data (large dataset, includes data and test/verification suite)	GitHub	View	US Ltr | A4
world database	TGZ | Zip	View	US Ltr | A4
world_x database	TGZ | Zip	View	US Ltr | A4
sakila database	TGZ | Zip	View	US Ltr | A4
airportdb database (large dataset, intended for MySQL on OCI and HeatWave)	TGZ | Zip	View	US Ltr | A4
menagerie database	TGZ | Zip		

Sakila database

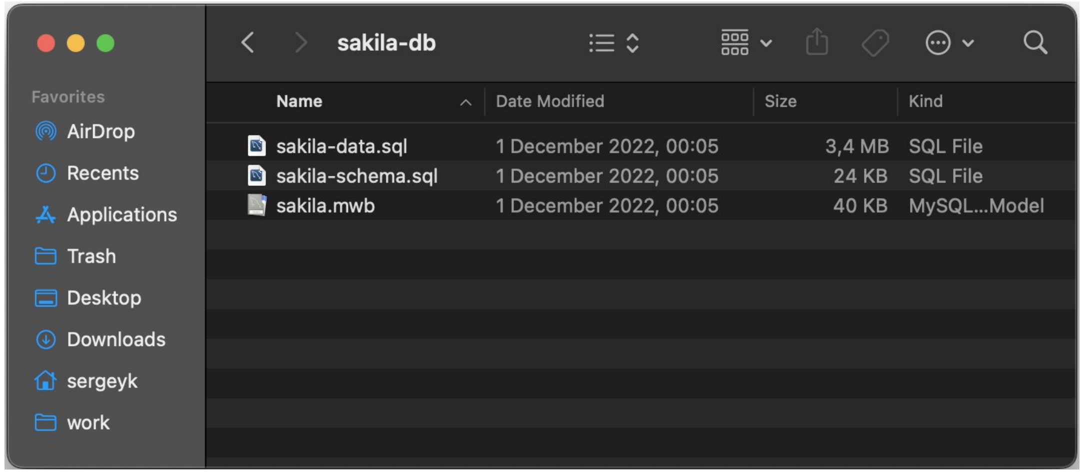
- Sakila is a classic example database
- Simple and small, yet allows for some complicated queries
- Rental company data

Sakila database



Sakila database – installation

- Download from mysql.com
- Unpack (both macOS and Windows support zip files by default
- Navigate to the unpacked directory



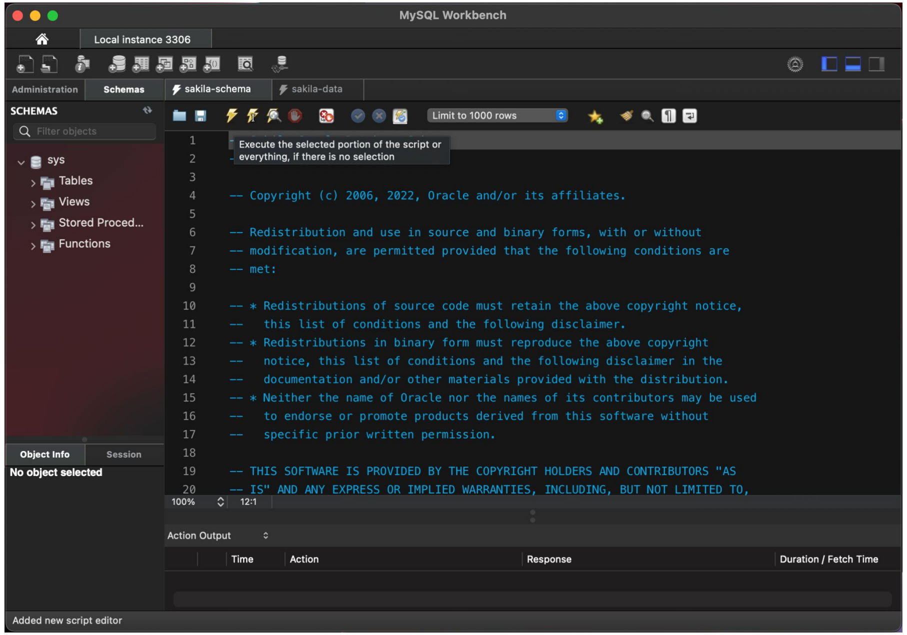
Name	Date Modified	Size	Kind
sakila-data.sql	1 December 2022, 00:05	3,4 MB	SQL File
sakila-schema.sql	1 December 2022, 00:05	24 KB	SQL File
sakila.mwb	1 December 2022, 00:05	40 KB	MySQL...Model

Sakila database – installation

- We will use MySQL Workbench to install the data
- Open two scripts
- Either from Workbench
- Or just doubleclick from the OS (Explorer or Finder)

Sakila database – installation

- Execute sakila-schema



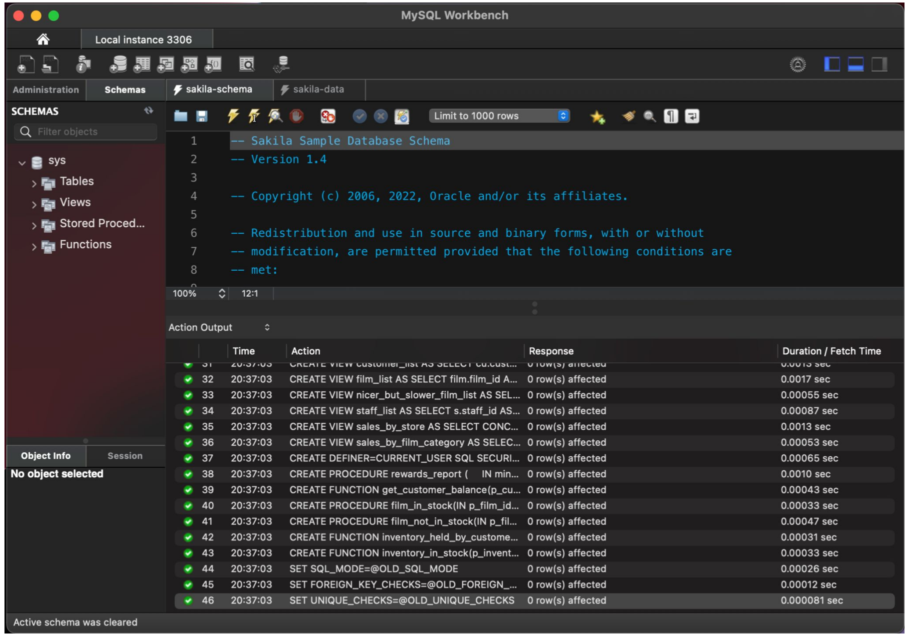
Execute the selected portion of the script or everything, if there is no selection

```
-- Copyright (c) 2006, 2022, Oracle and/or its affiliates.

-- Redistribution and use in source and binary forms, with or without
-- modification, are permitted provided that the following conditions are
-- met:

-- * Redistributions of source code must retain the above copyright notice,
--   this list of conditions and the following disclaimer.
-- * Redistributions in binary form must reproduce the above copyright
--   notice, this list of conditions and the following disclaimer in the
--   documentation and/or other materials provided with the distribution.
-- * Neither the name of Oracle nor the names of its contributors may be used
--   to endorse or promote products derived from this software without
--   specific prior written permission.

-- THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
-- IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
```

Added new script editor

Sakila database – installation

- All action output items (but one warning) should be green



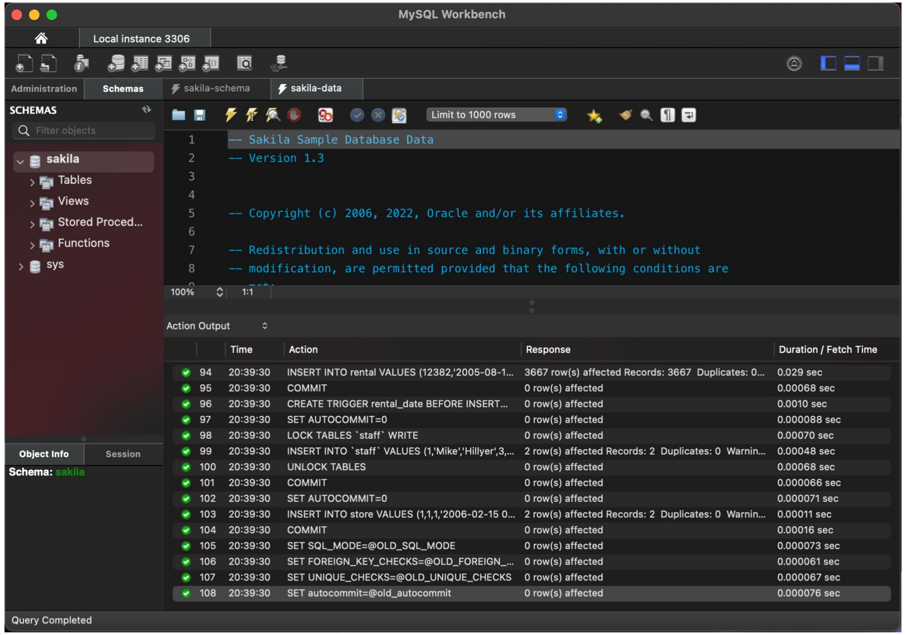
```
-- Sakila Sample Database Schema
-- Version 1.4

-- Copyright (c) 2006, 2022, Oracle and/or its affiliates.

-- Redistribution and use in source and binary forms, with or without
-- modification, are permitted provided that the following conditions are
-- met:
```

```
    Time      Action                                       Response            Duration / Fetch Time
31  20:37:03  CREATE VIEW customer_list AS SELECT cu.cust... 0 row(s) affected  0.0013 sec
32  20:37:03  CREATE VIEW film_list AS SELECT film.film_id A... 0 row(s) affected  0.0017 sec
33  20:37:03  CREATE VIEW nicer_but_slower_film_list AS SEL... 0 row(s) affected  0.00055 sec
34  20:37:03  CREATE VIEW staff_list AS SELECT s.staff_id AS... 0 row(s) affected  0.00087 sec
35  20:37:03  CREATE VIEW sales_by_store AS SELECT CONC... 0 row(s) affected  0.0013 sec
36  20:37:03  CREATE VIEW sales_by_film_category AS SELEC... 0 row(s) affected  0.00053 sec
37  20:37:03  CREATE DEFINER=CURRENT_USER SQL SECURI... 0 row(s) affected  0.00065 sec
38  20:37:03  CREATE PROCEDURE rewards_report (    IN min... 0 row(s) affected  0.0010 sec
39  20:37:03  CREATE FUNCTION get_customer_balance(p_cu... 0 row(s) affected  0.00043 sec
40  20:37:03  CREATE PROCEDURE film_in_stock(IN p_film_id... 0 row(s) affected  0.00033 sec
41  20:37:03  CREATE PROCEDURE film_not_in_stock(IN p_fil... 0 row(s) affected  0.00047 sec
42  20:37:03  CREATE FUNCTION inventory_held_by_custome... 0 row(s) affected  0.00031 sec
43  20:37:03  CREATE FUNCTION inventory_in_stock(p_invent... 0 row(s) affected  0.00033 sec
44  20:37:03  SET SQL_MODE=@OLD_SQL_MODE                   0 row(s) affected  0.00026 sec
45  20:37:03  SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_...      0 row(s) affected  0.00012 sec
46  20:37:03  SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS         0 row(s) affected  0.000081 sec
```

datasciencedojo
data science for everyone

Sakila database – installation

- Execute sakila-data and observe the output

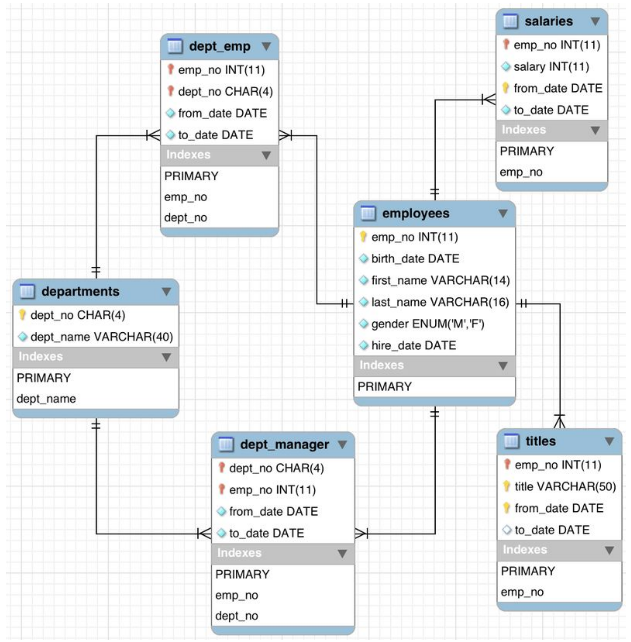


```
-- Sakila Sample Database Data
-- Version 1.3


-- Copyright (c) 2006, 2022, Oracle and/or its affiliates.

-- Redistribution and use in source and binary forms, with or without
-- modification, are permitted provided that the following conditions are
```

	Time	Action	Response	Duration / Fetch Time
✓ 94	20:39:30	INSERT INTO rental VALUES (12382,'2005-08-1...	3667 row(s) affected Records: 3667 Duplicates: 0...	0.029 sec
✓ 95	20:39:30	COMMIT	0 row(s) affected	0.00068 sec
✓ 96	20:39:30	CREATE TRIGGER rental_date BEFORE INSERT...	0 row(s) affected	0.0010 sec
✓ 97	20:39:30	SET AUTOCOMMIT=0	0 row(s) affected	0.000088 sec
✓ 98	20:39:30	LOCK TABLES `staff` WRITE	0 row(s) affected	0.00070 sec
✓ 99	20:39:30	INSERT INTO `staff` VALUES (1,'Mike','Hillyer',3,...	2 row(s) affected Records: 2 Duplicates: 0 Warnin...	0.00048 sec
✓ 100	20:39:30	UNLOCK TABLES	0 row(s) affected	0.00068 sec
✓ 101	20:39:30	COMMIT	0 row(s) affected	0.000066 sec
✓ 102	20:39:30	SET AUTOCOMMIT=0	0 row(s) affected	0.000071 sec
✓ 103	20:39:30	INSERT INTO store VALUES (1,1,1,'2006-02-15 0...	2 row(s) affected Records: 2 Duplicates: 0 Warnin...	0.00011 sec
✓ 104	20:39:30	COMMIT	0 row(s) affected	0.00016 sec
✓ 105	20:39:30	SET SQL_MODE=@OLD_SQL_MODE	0 row(s) affected	0.000073 sec
✓ 106	20:39:30	SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_...	0 row(s) affected	0.000061 sec
✓ 107	20:39:30	SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS	0 row(s) affected	0.000067 sec
✓ 108	20:39:30	SET autocommit=@old_autocommit	0 row(s) affected	0.000076 sec

Employee database

- Another great sample database
- Larger size than Sakila, though still on a small side (~150MiB)
- Simpler schema structure



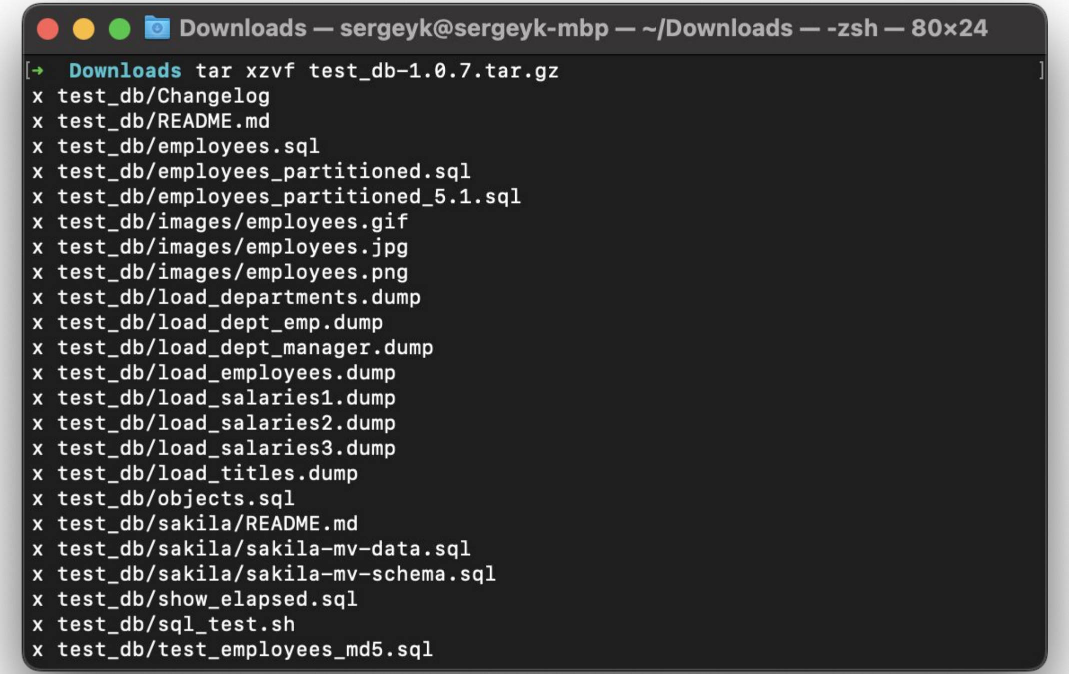
datasciencedojo
data science for everyone

Employee database – installation

- We will need to use a terminal and MySQL CLI
- Download Employee database from github
- Package is a tar.gz file, not native for Windows or macOS
- Both OSes (Win from version 10) support tar, however
- Command for Windows and macOS is going to be the same

Employee database – installation

- `tar xzvf test_db-1.0.7.tar.gz`

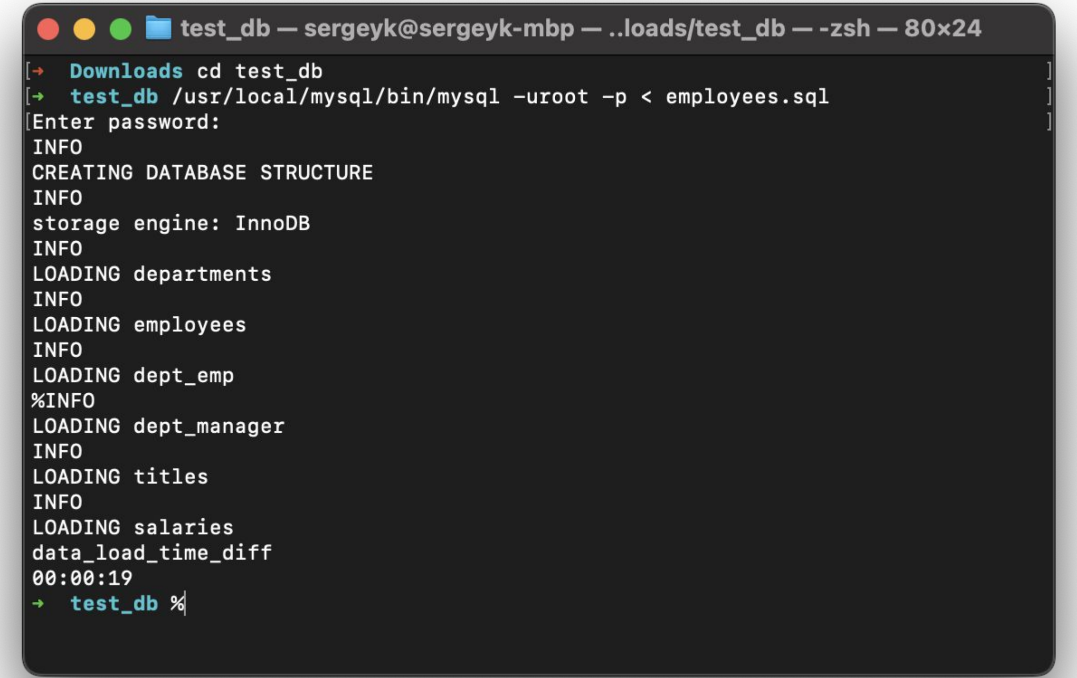


Downloads — sergeyk@sergeyk-mbp — ~/Downloads — -zsh — 80×24

```
➜  Downloads tar xzvf test_db-1.0.7.tar.gz
x test_db/Changelog
x test_db/README.md
x test_db/employees.sql
x test_db/employees_partitioned.sql
x test_db/employees_partitioned_5.1.sql
x test_db/images/employees.gif
x test_db/images/employees.jpg
x test_db/images/employees.png
x test_db/load_departments.dump
x test_db/load_dept_emp.dump
x test_db/load_dept_manager.dump
x test_db/load_employees.dump
x test_db/load_salaries1.dump
x test_db/load_salaries2.dump
x test_db/load_salaries3.dump
x test_db/load_titles.dump
x test_db/objects.sql
x test_db/sakila/README.md
x test_db/sakila/sakila-mv-data.sql
x test_db/sakila/sakila-mv-schema.sql
x test_db/show_elapsed.sql
x test_db/sql_test.sh
x test_db/test_employees_md5.sql
```

Employee database – installation

- `mysql -u root -p < employees.sql`


test_db — sergeyk@sergeyk-mbp — ..loads/test_db — -zsh — 80×24

```
➜  Downloads cd test_db
➜  test_db /usr/local/mysql/bin/mysql -uroot -p < employees.sql
Enter password:
INFO
CREATING DATABASE STRUCTURE
INFO
storage engine: InnoDB
INFO
LOADING departments
INFO
LOADING employees
INFO
LOADING dept_emp
%INFO
LOADING dept_manager
INFO
LOADING titles
INFO
LOADING salaries
data_load_time_diff
00:00:19
➜  test_db %
```

Interfacing with MySQL – MySQL Workbench

- Now that we have sample data
- Let’s explore and run some queries
- It’s super easy

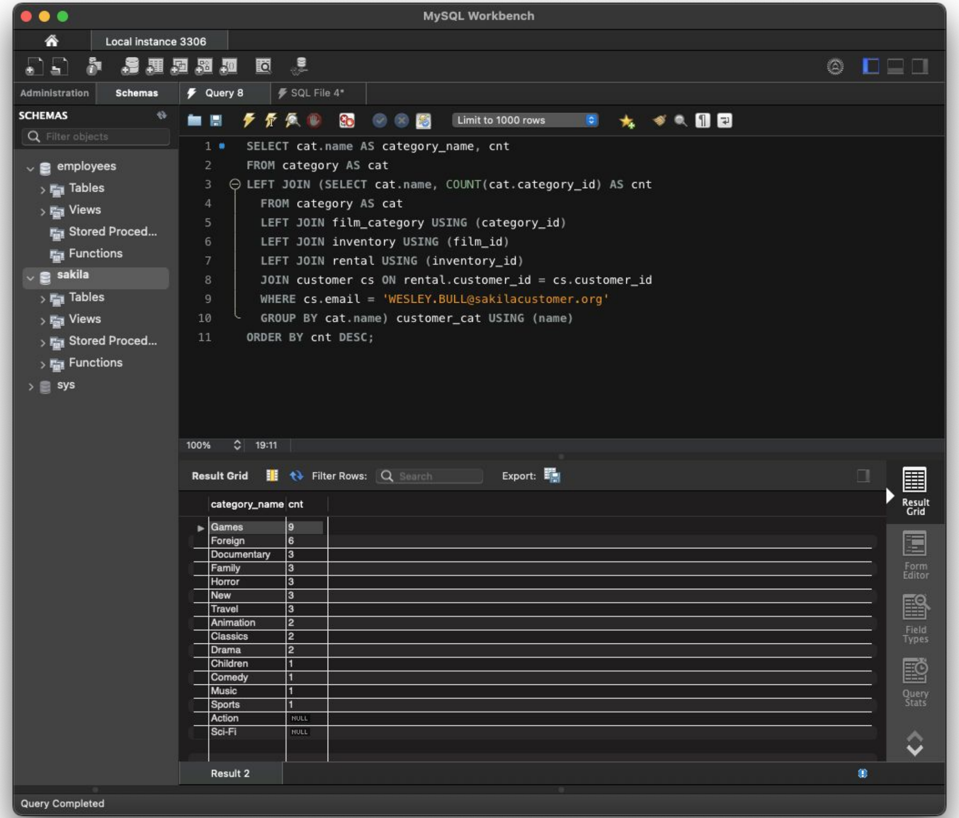
Interfacing with MySQL – Sakila sample query

- Even though Sakila is simple, it allows for fairly complicated queries
- Break down rented films by category for a specific customer
- Include 0 or NULL where they rented no films
- All basic SQL but might not be as straightforward to write initially

Interfacing with MySQL – Sakila sample query

```sql
SELECT cat.name AS category_name , cnt
FROM category AS cat
LEFT JOIN (SELECT cat.name, COUNT(cat.category_id) AS cnt
  FROM category AS cat
  LEFT JOIN film_category USING (category_id)
  LEFT JOIN inventory USING (film_id)
  LEFT JOIN rental USING (inventory_id)
  JOIN customer cs ON rental.customer_id = cs.customer_id
  WHERE cs.email = 'WESLEY.BULL@sakilacustomer.org'
  GROUP BY cat.name) customer_cat USING (name)
ORDER BY cnt DESC;
```

Interfacing with MySQL – Sakila sample query



```sql
SELECT cat.name AS category_name, cnt
FROM category AS cat
LEFT JOIN (SELECT cat.name, COUNT(cat.category_id) AS cnt
  FROM category AS cat
  LEFT JOIN film_category USING (category_id)
  LEFT JOIN inventory USING (film_id)
  LEFT JOIN rental USING (inventory_id)
  JOIN customer cs ON rental.customer_id = cs.customer_id
  WHERE cs.email = 'WESLEY.BULL@sakilacustomer.org'
  GROUP BY cat.name) customer_cat USING (name)
ORDER BY cnt DESC;
```

category_name	cnt
Games	9
Foreign	6
Documentary	3
Family	3
Horror	3
New	3
Travel	3
Animation	2
Classics	2
Drama	2
Children	1
Comedy	1
Music	1
Sports	1
Action	NULL
Sci-Fi	NULL

Interfacing with MySQL – Employees sample query

- Employees allows playing around with more complicated queries
- Still really small and simple
- Let’s try some window functions
- And CTEs
- Show minimum and maximum salaries per department with each employee
- Find percentile bucket where an employee’s salary falls

Interfacing with MySQL – Employees sample query

- Employees allows playing around with more complicated queries
- Still really small and simple
- Let’s try some window functions
- And CTEs

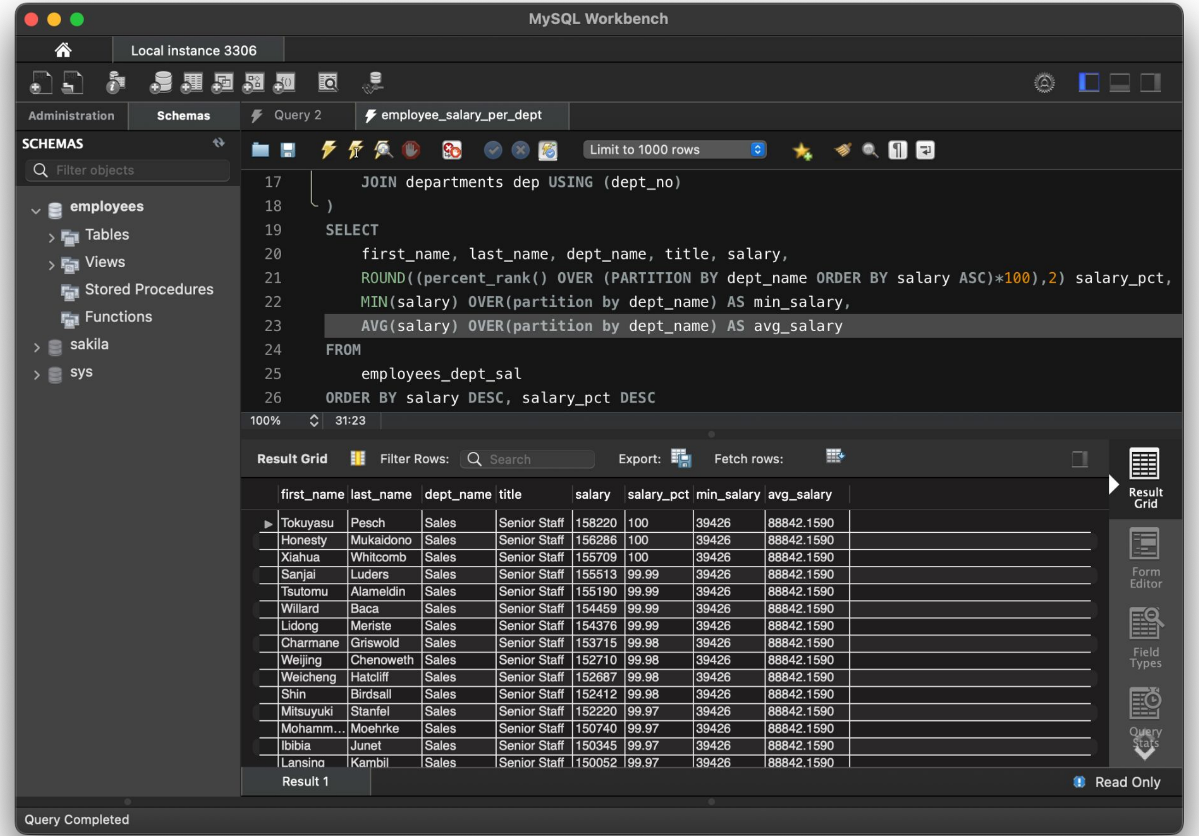
Interfacing with MySQL – Employees sample query

- Employees allows playing around with more complicated queries
- Still really small and simple
- Let’s try some window functions
- And CTEs
- Rank salaries per department and show alongside min and avg salaries

Interfacing with MySQL – Employees sample query

```
WITH current_salaries AS (
   SELECT emp_no, salary
   FROM salaries
           WHERE to_date = '9999-01-01'
),
current_titles AS (
   SELECT emp_no, title
           FROM titles
           WHERE to_date = '9999-01-01'
),
employees_dept_sal AS (
   SELECT emp.first_name, emp.last_name, dep.dept_name, ct.title, cs.salary
   FROM employees emp
           JOIN current_salaries cs USING (emp_no)
           JOIN current_titles ct USING (emp_no)
           JOIN dept_emp de USING (emp_no)
           JOIN departments dep USING (dept_no)
)
SELECT
   first_name, last_name, dept_name, title, salary,
           ROUND((percent_rank() OVER (PARTITION BY dept_name ORDER BY salary ASC)*100),2) salary_pct,
           MIN(salary) OVER(partition by dept_name) AS min_salary,
           AVG(salary) OVER(partition by dept_name) AS avg_salary
FROM
   employees_dept_sal
ORDER BY salary DESC, salary_pct DESC
```

Interfacing with MySQL – Employees sample query



```
    JOIN departments dep USING (dept_no)
)
SELECT
    first_name, last_name, dept_name, title, salary,
    ROUND((percent_rank() OVER (PARTITION BY dept_name ORDER BY salary ASC)*100),2) salary_pct,
    MIN(salary) OVER(partition by dept_name) AS min_salary,
    AVG(salary) OVER(partition by dept_name) AS avg_salary
FROM
    employees_dept_sal
ORDER BY salary DESC, salary_pct DESC
```

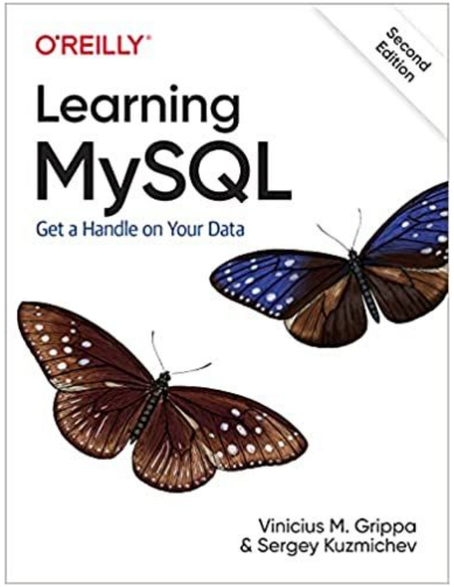
first_name	last_name	dept_name	title	salary	salary_pct	min_salary	avg_salary
Tokuyasu	Pesch	Sales	Senior Staff	158220	100	39426	88842.1590
Honesty	Mukaidono	Sales	Senior Staff	156286	100	39426	88842.1590
Xiahua	Whitcomb	Sales	Senior Staff	155709	100	39426	88842.1590
Sanjai	Luders	Sales	Senior Staff	155513	99.99	39426	88842.1590
Tsutomu	Alameldin	Sales	Senior Staff	155190	99.99	39426	88842.1590
Willard	Baca	Sales	Senior Staff	154459	99.99	39426	88842.1590
Lidong	Meriste	Sales	Senior Staff	154376	99.99	39426	88842.1590
Charmane	Griswold	Sales	Senior Staff	153715	99.98	39426	88842.1590
Weijing	Chenoweth	Sales	Senior Staff	152710	99.98	39426	88842.1590
Weicheng	Hatcliff	Sales	Senior Staff	152687	99.98	39426	88842.1590
Shin	Birdsall	Sales	Senior Staff	152412	99.98	39426	88842.1590
Mitsuyuki	Stanfel	Sales	Senior Staff	152220	99.97	39426	88842.1590
Mohamm…	Moehrke	Sales	Senior Staff	150740	99.97	39426	88842.1590
Ibibia	Junet	Sales	Senior Staff	150345	99.97	39426	88842.1590
Lansing	Kambil	Sales	Senior Staff	150052	99.97	39426	88842.1590

QUESTIONS?

datasciencedojo
data science for everyone

You can get our book “Learning MySQL” using one of the following URLs:

- O’Reilly Learning portal
- Amazon book page


O'REILLY
Second Edition
Learning MySQL
Get a Handle on Your Data
Vinicius M. Grippa & Sergey Kuzmichev

PERCONA

We're hiring

Join our team!

OPEN POSITIONS

#RemoteWork

APPLY NOW: percona.com/careers

Investing.com Careers

Investing.com is a financial markets platform providing real-time data, quotes, charts, financial tools, breaking news, and analysis across 300 exchanges around the world in 43 language editions. With over 300,000 financial instruments covered, Investing.com offers unlimited access to cutting-edge financial market tools such as customized portfolios, personal alerts, calendars, calculators, and financial insights, completely free of charge.

Explore our Open Positions

谢谢

Thank you

Grazie

Obrigado

Gracias