

# Cursus Databanken

Docenten: Eric Juchtmans, Wim Goedertier

*Laatste wijziging: 22 sep 2024*

## *04-MySQL-Shell-gebruiken.docx*

### Inhoud

MySQL Shell vs. MySQL Command Line Client .....	2
CLI-clients vs. GUI-clients vs. Libraries .....	2
Opstarten en verbinden .....	3
Afsluiten .....	4
Auto-completion (de TAB-toets) .....	5
History van instructies .....	5
Help.....	5
Ctrl-C.....	5
Kopiëren en plakken.....	5
A. Tekst kopiëren .....	5
B. Tekst plakken.....	6
Oorzaken van verwarring binnen een CLI.....	6
A. Tekst selecteren BEVRIEST het scherm.....	6
B. Muis gebruiken heeft geen zin .....	6
C. Grootte van lettertype of venster aanpassen, brengt de CLI in de war .....	6
Nog enkele shortcuts.....	7
Ctrl-L: Scherm leeg maken .....	7
Ctrl-A (of HOME): naar het begin van de lijn gaan .....	8
Ctrl-E (of END): naar het einde van de lijn gaan .....	8
Configuratie aanpassen.....	8
A. Automatisch naar SQL-mode.....	8
B. Automatisch connectie maken .....	8
C. Prompt aanpassen.....	8

## MySQL Shell vs. MySQL Command Line Client

In “02-MySQL-installeren” hebben we de “Command Line Client” gebruikt die wordt meegeleverd met “MySQL Server 8.4” om de installatie te testen. Deze client is super basic en slechts enkele MB groot, maar hij doet wat hij moet doen.

In “03-DBs-importeren” en in de komende weken gaan we de “MySQL Shell” gebruiken. Dit pakket moet apart geïnstalleerd worden (maar dat deden we al in “02-MySQL-installeren”) en is enkele 100 MB groot. De “Shell” werkt **zeer** gelijkaardig als de “Command Line Client”.

Het grootste voordeel van de Shell (ten opzichte van de Command Line Client) is de **auto-completion**. Dat betekent dat we de meeste tekst niet volledig zullen moeten intikken. (Zie verder)

We kunnen de configuratie van de MySQL Shell ook gemakkelijk naar onze hand zetten. Zodat hij automatisch connectie maakt met de juiste server. We kunnen ook de “prompt” aanpassen. (Zie verder)

## CLI-clients vs. GUI-clients vs. Libraries

De “Shell” en de “Command Line Client” zijn allebei “text based” of werken allebei volgens het principe van de CLI (command line interface): Je tikt iets in, als je op ENTER drukt, wordt de lijn verwerkt. Het resultaat of het antwoord verschijnt als tekst op de volgende lijn.

Tijdens de eerste 3 à 4 weken gaan we *enkel* de CLI-clients gebruiken. Daar zijn enkele goede redenen voor:

- Bij evaluaties (op punten) zal je, voor dit vak, **enkel “MySQL Shell” mogen gebruiken**. Het is dus belangrijk dat je hier voldoende mee oefent, zodat je die omgeving gewoon bent en weet hoe je bepaalde problemen oplost.
- In de eerste weken gaan we ons vooral focussen op de **syntax** van SQL en het begrijpen van **foutmeldingen**. Met een CLI-client zijn de mogelijkheden beperkt en is het dus gemakkelijker om enkel daar op te focussen.
- Als je op een bestaande database-server iets moet doen, dan zal je favoriete tool of client niet altijd beschikbaar zijn en soms zal je geen rechten hebben om die te installeren. Het is dan goed om te kunnen terugvallen op een eenvoudige CLI-client (die steeds beschikbaar zal zijn en niet moet geconfigureerd worden).
- Op Linux-servers wordt er voornamelijk met CLI-tools gewerkt.
- Als je dingen wil automatiseren via scripts (bvb. met PowerShell of met Bash vanuit Linux), dan zal je ook voornamelijk CLI-tools moeten gebruiken.

Na enkele weken gaan we ook kennis maken met database-clients (tools) met een GUI (grafische user interface). Deze GUI-clients hebben dezelfde mogelijkheden als de CLI-clients, maar ze zijn meestal wat gebruiksvriendelijker en intuïtiever én ze hebben meestal nog een pak extra features zoals bvb. het genereren van SQL-code op basis van een grafische voorstelling van je databank-structuur (ERD) en omgekeerd.

Een goed voorbeeld van een grafische tool is “DBeaver” (<https://dbeaver.io/>), die zullen we binnen 4 of 5 weken ook leren gebruiken. Een ander voorbeeld is “MySQL Workbench”. De laatste versie van de Workbench is echter 8.0.38, terwijl de server en de Shell al op 8.4.x zitten. De Workbench wordt dus niet langer ondersteund.

In het 2<sup>de</sup> semester, in het vak “Programmeren Gevorderd”, zal je SQL beginnen gebruiken binnen je programmeercode in C#. In dat geval zal je niet langer een CLI-client of een GUI-

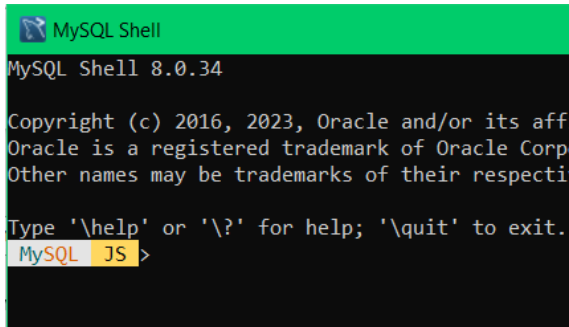
client gebruiken om te connecteren en te communiceren met de database-server, maar je zal bepaalde methods uit de .Net-library gebruiken. We spreken hier dan meestal over “Connectors”.

## Opstarten en verbinden

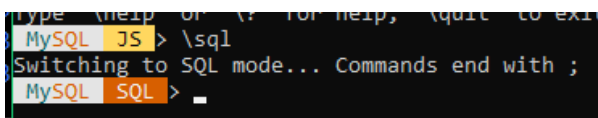
1. Start de “MySQL Shell”.

Druk **⌘+S** (Windows-key+S) of klik op het Vergrootglas) en tik “MySQL Shell”.

Alternatief: Open de “Opdrachtprompt” (**⌘+R** → `cmd`) of “Powershell” (**⌘+R** → `powershell`) en voer volgend commando uit: **"C:\Program Files\MySQL\MySQL Shell 8.4\bin\mysqlsh.exe"**

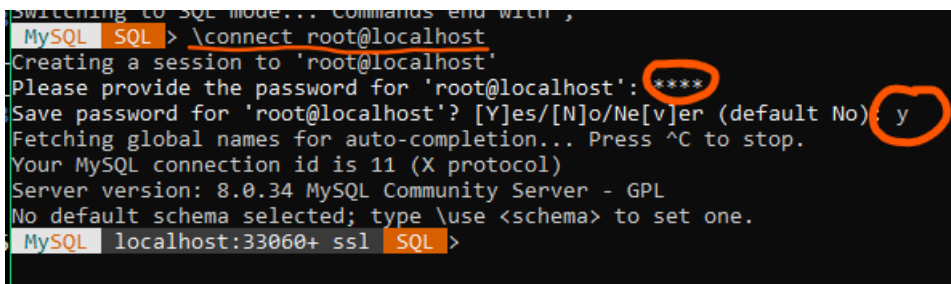


2. Tik **\sql** om van JavaScript-mode naar SQL-mode te gaan.  
(Deze stap is normaal niet meer nodig in MySQL Shell 8.4)



3. Tik **\connect root@localhost** om connectie te maken met je eigen lokale MySQL-server.

Als je niet steeds je paswoord wilt intikken, dan tik je **y** bij de vraag “Save password for 'root@localhost'?”



Met deze database-client kan je connectie maken met gelijk welke MySQL-server die beschikbaar is ergens op het internet (in de cloud). In “root@localhost” vervangen we dan “root” door de juiste username en “localhost” door de hostname of het IP-adres van de MySQL-server waarop we willen inloggen.

Bijvoorbeeld:

**\connect jan@server17.combell.be**  
of **\connect els@123.45.67.89**

“jan@server17.combell.be” en “els@123.45.67.89” zijn *geen* e-mailadressen. We noemen het een “connection string”. Het bevat alle info (behalve het paswoord) om connectie te maken met een bepaalde MySQL-server.

Bij het vak Databanken gaan we altijd met onze eigen MySQL-server werken die we lokaal op onze laptop geïnstalleerd hebben. We zullen dus steeds “root@localhost” als connectie-string gebruiken.

In plaats van `\connect` mag je ook `\c` gebruiken. Dat is wat minder tikwerk.

We kunnen de ***syntax*** van het connect-commando als volgt samenvatten:

```
\c[onnect] <username>@<hostname|ip>
```

De vierkante haakjes ([ en ]) geven aan dat de `onnect` optioneel is. Je mag dus kiezen of je het schrijft of niet.

De hoekhaakjes of vishaakjes (< en >) bij `<username>` geven aan dat je deze tekst (inclusief de haakjes) moet vervangen door een username.

Het verticaal streepje of de “pipe” (|) bij `<hostname|ip>` betekent “of”.

4. We zijn nu geconnecteerd met de database-server. Vanaf nu kunnen we SQL gebruiken om te communiceren met de server. Het gebruik van SQL leer je in de volgende hoofdstukken.

Hier alvast enkele voorbeelden:

Kijk welke databases er bestaan op de server:

```
SHOW DATABASES ; (vergeet de puntkomma (;) niet !!)
```

Binnen MySQL zijn **database** en **schema** synoniemen. Ze betekenen exact hetzelfde. Je kan dus ook SHOW SCHEMAS; tikken. Bij Microsoft SQL Server kan je meerdere schema's binnen één databank hebben en wordt het concept schema vooral gebruikt om verschillende permissies te kunnen toekennen binnen een bepaalde databank. In andere gevallen (bvb. in cursussen of in de spreektaal) wordt er met “het schema” enkel de **structuur** bedoeld, terwijl database dan eerder op het geheel, inclusief de gegevens slaat.

Kies welke database je gaat gebruiken. Dit wordt dan de “actieve” databank:

```
USE <database>;
```

Bijvoorbeeld:

```
USE bib;
```

Kijk welke tabellen er bestaan binnen de “actieve” databank:

```
SHOW TABLES ;
```

Bekijk de inhoud van een bepaalde tabel:

```
SELECT * FROM <table>;
```

Bijvoorbeeld:

```
SELECT * FROM boeken;
```

Bekijk enkel de eerste 10 lijnen van een bepaalde tabel:

```
SELECT * FROM <table> LIMIT 10;
```

Bijvoorbeeld:

```
SELECT * FROM boeken LIMIT 10;
```

## Afsluiten

Je sessie afsluiten doe je met:

`\q` of `\quit`

## Auto-completion (de TAB-toets)

Je kan op ieder moment één of twee keer op de TAB-toets (boven “shift-lock”) duwen om je SQL-instructie automatisch te laten aanvullen of om te zien wat de mogelijkheden zijn.

Tik bvb. “**sel**” en dan op de TAB-toets. De “**sel**” wordt automatisch “**SELECT**”.

Tik nu “ **\* F**” en dan *tweemaal* op TAB. Je ziet een aantal voorstellen (FALSE, FAST, FETCH, ...). “FROM” staat er ook tussen. Tik nu “**R**” en dan één keer op TAB. Er verschijnt automatisch “**FROM**”. Tik de eerste 2 of 3 letters van de tabel die je wilt zien (bvb. “**bo**”), daarna één of twee keer op TAB. Enz, enz, ...

## History van instructies

Met “pijlje omhoog” zie je je vorige instructie. Nogmaals “pijlje omhoog” toont de instructie daarvoor. Enz. Met “pijlje omlaag” loop je in de ander richting door de “history” van je instructies.

Je kan ook `\history` gebruiken. Je krijgt dan een mooi overzicht van alle commando’s die je reeds uitvoerde.

## Help

Tik zeker ook eens `\help` (of `\?` of `\h`). Het is een bron van extra informatie...

## Ctrl-C

Met Ctrl-C kan je het *invoeren* van een instructie onderbreken (wat je ingetikt hebt) ofwel het *uitvoeren* van een instructie onderbreken (als die te lang duurt).

Je hebt bijvoorbeeld al een lange instructie op de command line ingetikt, maar je bedenkt je en je beslist dat je eigenlijk eerst iets totaal anders wilt doen. Druk dan Ctrl-C, hierdoor krijg een terug een lege lijn waarop je iets nieuws kan tikken.

Als je per ongeluk een “**SELECT \* FROM ...**” gedaan hebt op een tabel met 1 miljoen rijen, dan wil je niet wachten tot al die rijen over je scherm gescrolld zijn. Druk dan Ctrl-C om dit te onderbreken.

## Kopiëren en plakken

### A. Tekst kopiëren

Selecteer de tekst *met de muis* en druk op de rechter muistoets  
*ofwel*

selecteer de tekst *met de muis* en druk op ENTER

**ofwel**

selecteer de tekst **met de muis** en druk Ctrl-C

Het gebruik van Ctrl-C om te kopiëren vanuit een CLI-client wordt afgeraden omdat Ctrl-C al een andere betekenis heeft. Niet alle CLI-clients ondersteunen Ctrl-C. De “rechter muistoets” is het meest universeel binnen CLI-clients in het algemeen.

## B. Tekst plakken

Controleer eerst of er geen tekst geselecteerd is. (Eén leeg, wit blokje is óók geselecteerde tekst, namelijk een spatie.)

Druk **eerst** op de ESC-toets als er wél iets geselecteerd was.

Druk op de rechter muistoets. De tekst wordt dan geplakt op de plaats waar de **tekst**-cursor staat. De positie van de **muis**-cursor is hier **niet** van belang.

(Ctrl-V werkt niet binnen MySQL Shell)

Let op: Als je iets wilt **plakken** in een Shell-venster, dan mag je er dus **niet** eerst in klikken. Bij het klikken selecteer je automatisch één karakter. Als je **dan** op de rechter muistoets klikt, dan **kopieer** je dus dat ene karakter. Als je onmiddellijk erna nog een 2<sup>de</sup> keer op de rechter muistoets klikt, dan plak je dus gewoon dat ene karakter (in veel gevallen gewoon een spatie). De oplossing is dus om eerst op de ESC-toets te drukken (zoals hierboven reeds beschreven) maar nog beter is om niet **in** het Shell-venster te klikken, maar op de balk bovenaan het venster, of om gebruik te maken van “Alt-TAB” om te wisselen van venster.

## Oorzaken van verwarring binnen een CLI

### A. Tekst selecteren BEVRIEST het scherm

... druk “ESC” (toets links bovenaan) om het bevroren (freezen) te beëindigen.

### B. Muis gebruiken heeft geen zin

Binnen een shell/CLI-omgeving heeft het totaal geen zin om met de muis te werken. Je kan de muis **enkel** gebruiken om tekst te selecteren of om tekst te plakken (zie hoger).

### C. Grootte van lettertype of venster aanpassen, brengt de CLI in de war

Als je de grootte van je lettertype aangepast hebt of je hebt de grootte van je venster aangepast, dan kan je best eerst eens rechts bovenaan op “Maximaliseren” (of “Verkleinen”) klikken en daarna op “Verkleinen” (of “Maximaliseren”). Druk daarna enkele keren op ENTER.

(Dubbelklikken op de balk bovenaan je venster heeft hetzelfde effect als op “Maximaliseren/Verkleinen” klikken.)

## Queries die uit meerdere lijnen bestaan

Soms is het handig om een SQL-query te verspreiden over meerdere lijnen, dit maakt het overzichtelijker. Als je dit doet in de Shell ziet dit er als volgt uit:

```
bib SQL > SELECT
-> id, titel
-> FROM
-> boeken
-> ORDER BY
-> id LIMIT 5;
+-----+
```

Het vervelende is dat als je daarna “pijltje omhoog” doet, omdat je deze SQL-query opnieuw wilt gebruiken (bvb. om nog iets te corrigeren of toe te voegen), dat alles dan plotseling terug op één lijn staat.

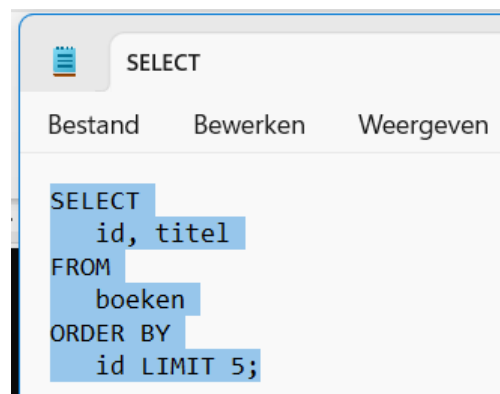
```
bib SQL > SELECT id, titel FROM boeken ORDER BY id LIMIT 5;_
```

Dit is vervelend en maakt het nogal onoverzichtelijk, zeker bij langere queries.

Het is dan aan te raden om de SQL-query in een tekst-editor te tikken (bvb. Kladblok of Notepad). En telkens naar de Shell te copy-and-pasten. Dit is een beetje vervelend, maar kan wel vrij snel gaan met volgende toetsen-combinaties:

- Ctrl-A (selecteer alles)
- Ctrl-C (copy)
- Alt-Tab (switch naar Shell)
- rechter muisklik (plak in de Shell).

Tijdens de evaluatie (voor punten) kan je geen Notepad gebruiken, maar je kan dan het invul-veld van het online antwoordformulier gebruiken om de SQL-query over meerdere lijnen te schrijven, en van daaruit copy-and-pasten.



De beste oplossing voor dit probleem is natuurlijk om een GUI-client te gebruiken (bvb. DBeaver), maar dit is niet toegelaten tijdens de evaluatie.

## Nog enkele shortcuts

### Ctrl-L: Scherm leeg maken

Als je vindt dat er te veel rommel op je scherm staat, druk dan “Ctrl-L”, dan krijg je terug een leeg scherm. (L van cLear)

## Ctrl-A (of HOME): naar het begin van de lijn gaan

Met “Ctrl-A” (of HOME) springt de *tekst*-cursor naar het begin van de lijn. Dat is soms handiger dan 50 keer op “pijlje naar links” te duwen...

In sommige situaties werkt de HOME-toets niet. De HOME-toets staat bij verschillende toetsenborden ook op een andere plaats. Daarom is “Ctrl-A” soms gemakkelijker.

## Ctrl-E (of END): naar het einde van de lijn gaan

Zie Ctrl-A.

## Ctrl-D: terminal afsluiten

Dit heeft hetzelfde effect als `\q` of `\quit`.

# Configuratie aanpassen

## A. Automatisch connectie maken

Als je niet steeds `\c root@localhost` wilt tikken, elke keer dat je MySQL Shell gebruikt, dan kan je `root@localhost` meegeven als argument bij het starten van MySQL Shell. Dit kan door de snelkoppeling (shortcut) aan te passen en het veld “Doel” te wijzigen van:

```
"C:\Program Files\MySQL\MySQL Shell 8.4\bin\mysqlsh.exe"
```

naar:

```
"C:\Program Files\MySQL\MySQL Shell 8.4\bin\mysqlsh.exe" root@localhost
```

## B. Prompt aanpassen

Vind je dat de standaard-prompt een beetje te lang is? Je kan:

veranderen in:

door een bestand `prompt.json` aan te maken in de folder

`C:\Users\xxx\AppData\Roaming\MySQL\mysqlsh\` met als inhoud:

```
{
  "symbols": {
    "separator": "-",
    "separator2": "-",
    "ellipsis": "-"
  }
}
```



```
    },  
    "segments": [  
        { "text": "%schema%" },  
        { "text": "%Mode%" }  
    ]  
}
```

## C. Automatisch naar SQL-mode

(Dit is niet meer nodig bij MySQL Shell 8.4)

Als je niet steeds **\sql** wilt tikken, elke keer dat je MySQL Shell gebruikt, dan kan je de optie **--sql** gebruiken bij het starten van de Shell. Het “Doel”-veld in de snelkoppeling (shortcut) wordt dan:

```
"C:\Progr...\MySQL\MySQL Shell 8.4\bin\mysqlsh.exe" root@localhost --sql
```