



MySQL Shell for Visual Studio Code

MySQL HeatWave's best companion for developers

Frédéric Descamps

Community Manager

Oracle MySQL

MySQL Belgian Days - February 2024



MySQL™ Belgian Days 2024





Who am I ?
about.me/lefred

Frédéric Descamps

- @lefred
- **MySQL** Evangelist
- using **MySQL** since version 3.20
- devops believer
- likes 🏀
- living in 🇧🇪
- <https://lefred.be>





MySQL HeatWave

Terminology

Terminology

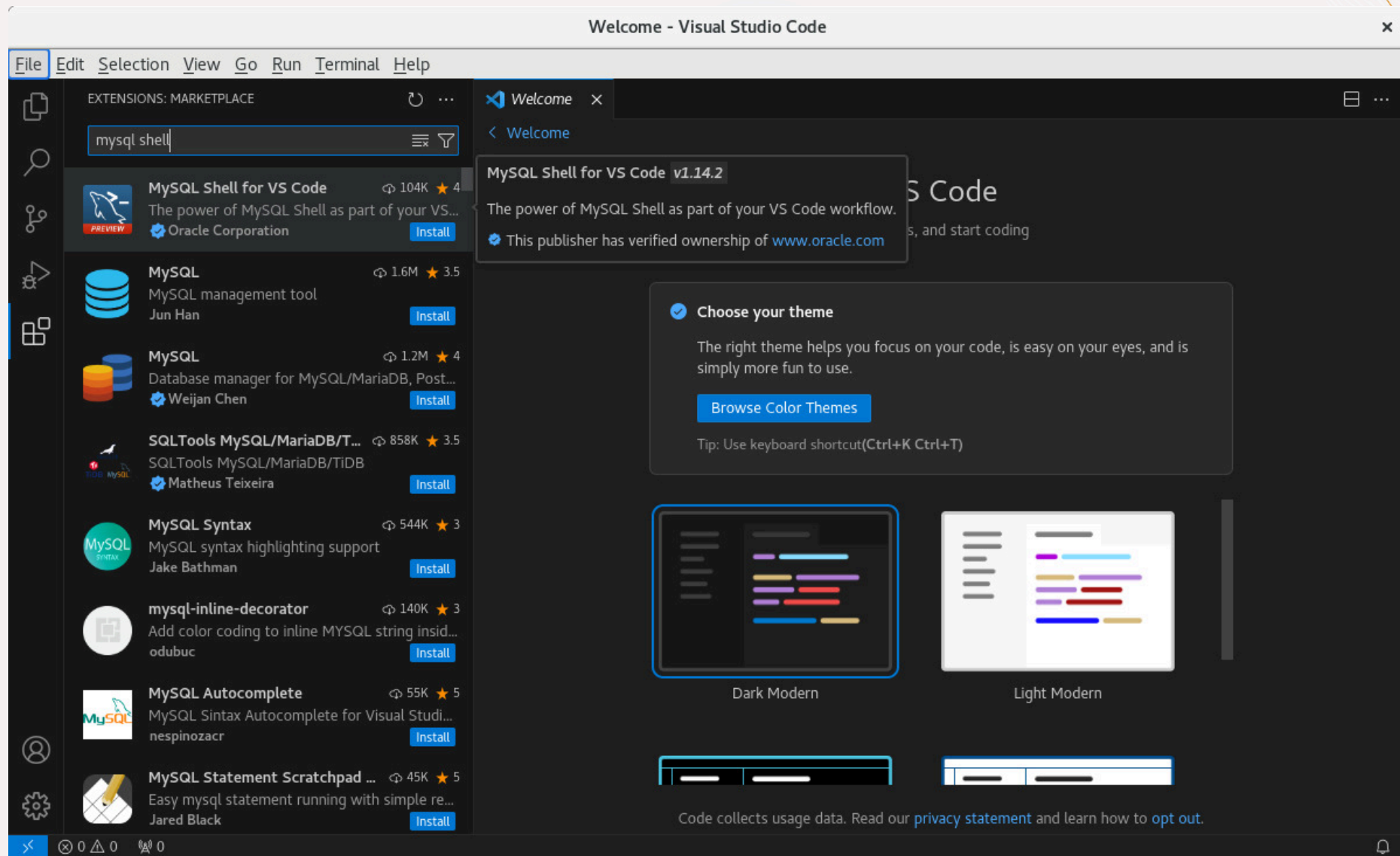
- **OCI**: Oracle Cloud Infrastructure
- **MySQL HeatWave**:
 - **MySQL HeatWave Database Service (MHS)**
 - **MySQL Database Service (MDS)**
- **MySQL HeatWave Cluster**:
 - OLTP and OLAP query accelerator
 - RAPID secondary engine



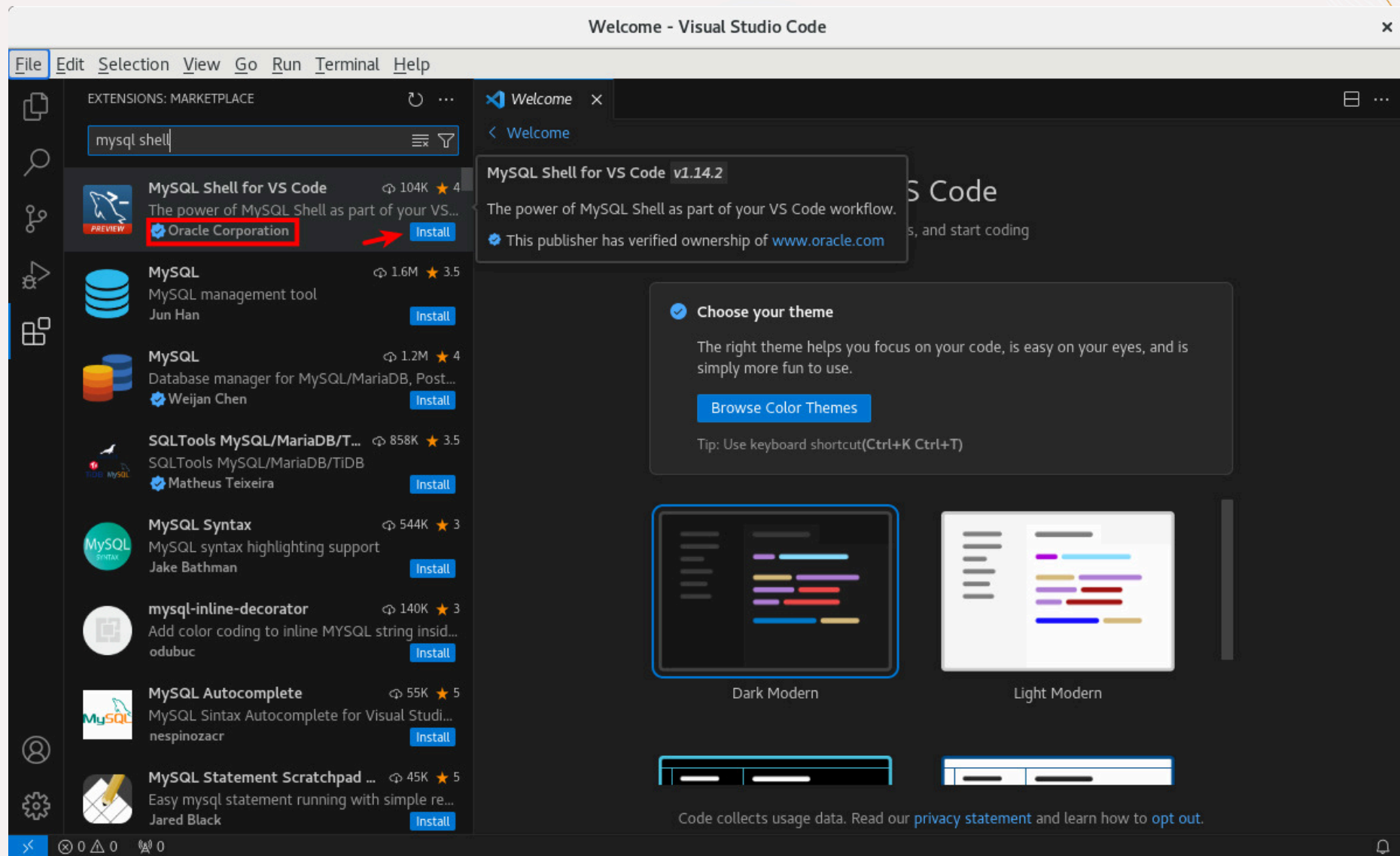
MySQL Shell for Visual Studio Code

Integration with MySQL HeatWave on OCI

Installation



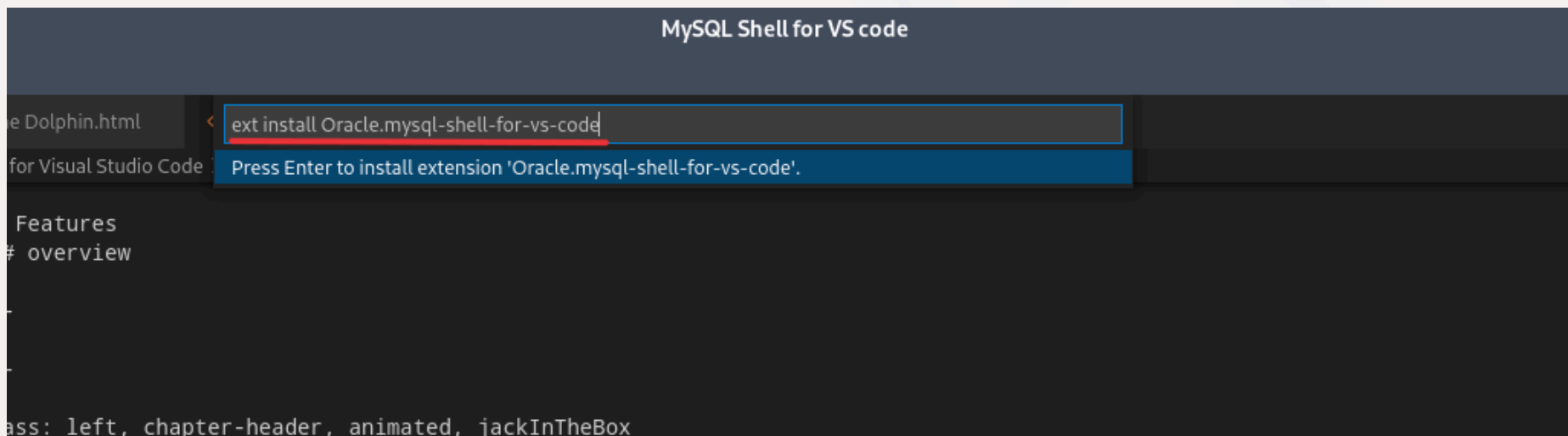
Installation



Installation

Or just use CTRL-P in Visual Studio code and paste the following command:

```
ext install Oracle.mysql-shell-for-vs-code
```



Installation

Extension: MySQL Shell for VS Code - Visual Studio Code

File Edit Selection View Go Run Terminal Help

EXTENSIONS: MARKETPLACE

mysql shell

MySQL Shell for VS Code v1.14.2
Oracle Corporation oracledb.com | 104,932 | ★★★★★ (19)
The power of MySQL Shell as part of your VS Code workflow.
[Install](#)

MySQL 1.6M ★ 3.5
MySQL management tool
Jun Han [Install](#)

MySQL 1.2M ★ 4
Database manager for MySQL/MariaDB, Post...
Weijian Chen [Install](#)

SQLTools MySQL/MariaDB/T... 858K ★ 3.5
SQLTools MySQL/MariaDB/TiDB
Matheus Teixeira [Install](#)

MySQL Syntax 544K ★ 3
MySQL syntax highlighting support
Jake Bathman [Install](#)

mysql-inline-decorator 140K ★ 3
Add color coding to inline MYSQL string insid...
odubuc [Install](#)

MySQL Autocomplete 55K ★ 5
MySQL Syntax Autocomplete for Visual Studi...
nespinozcr [Install](#)

MySQL Statement Scratchpad ... 45K ★ 5
Easy mysql statement running with simple re...
Jared Black [Install](#)

MySQL Shell for VS Code v1.14.2
Oracle Corporation oracledb.com | 104,932 | ★★★★★ (19)
The power of MySQL Shell as part of your VS Code workflow.
[Install](#)

DETAILS FEATURE CONTRIBUTIONS CHANGELOG

MySQL Shell for VS Code 1.14.2+8.1.1 Preview

This extension enables interactive editing and execution of SQL for MySQL Databases and the MySQL HeatWave Service. It integrates the MySQL Shell directly into VS Code development workflows.

IMPORTANT: Please note that this is a PREVIEW release which is not meant to be used in production.

Categories

- Programming Languages
- Data Science
- Visualization

Extension Resources

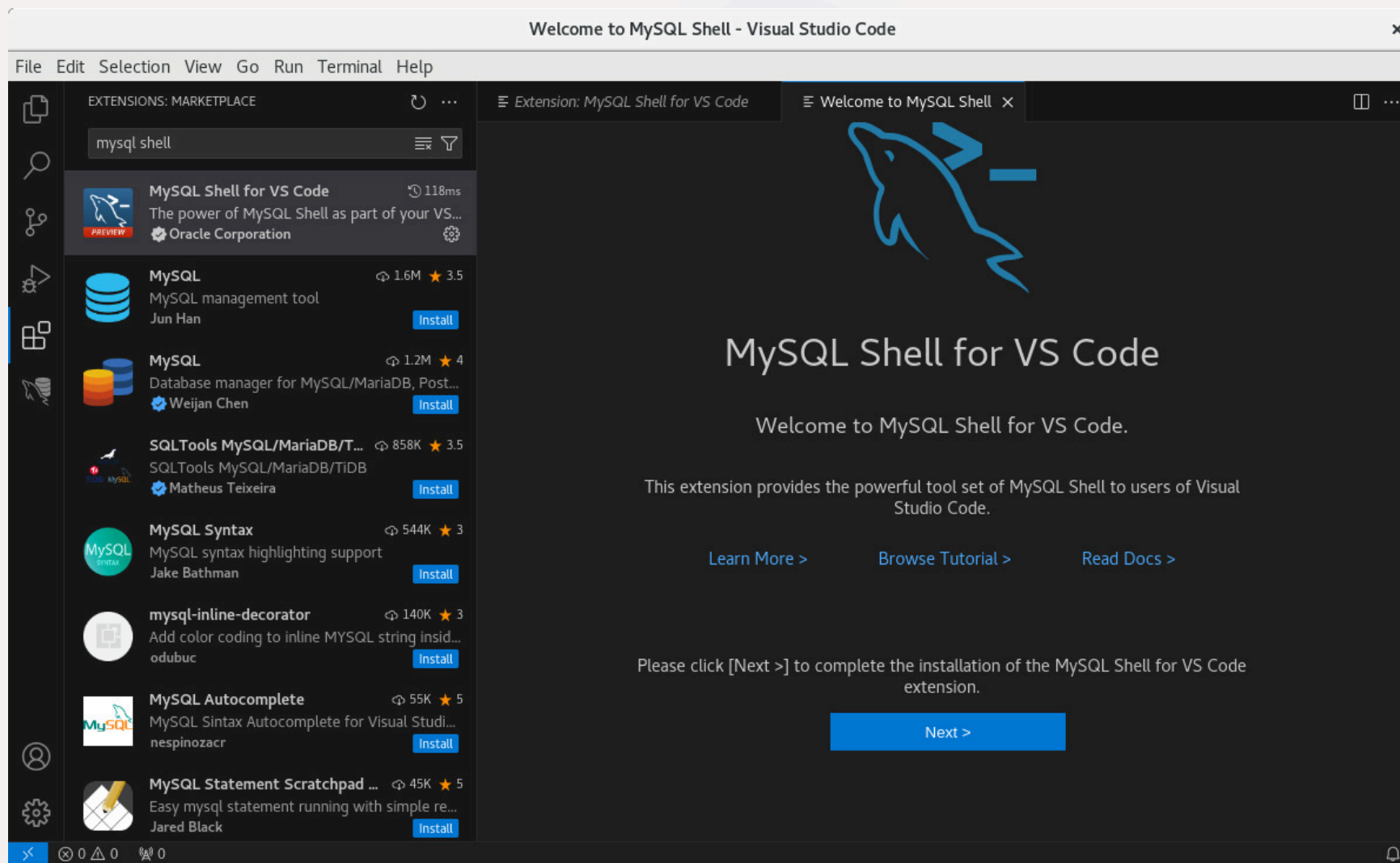
- [Marketplace](#)
- [Repository](#)
- [License](#)
- [Oracle Corporation](#)

More Info

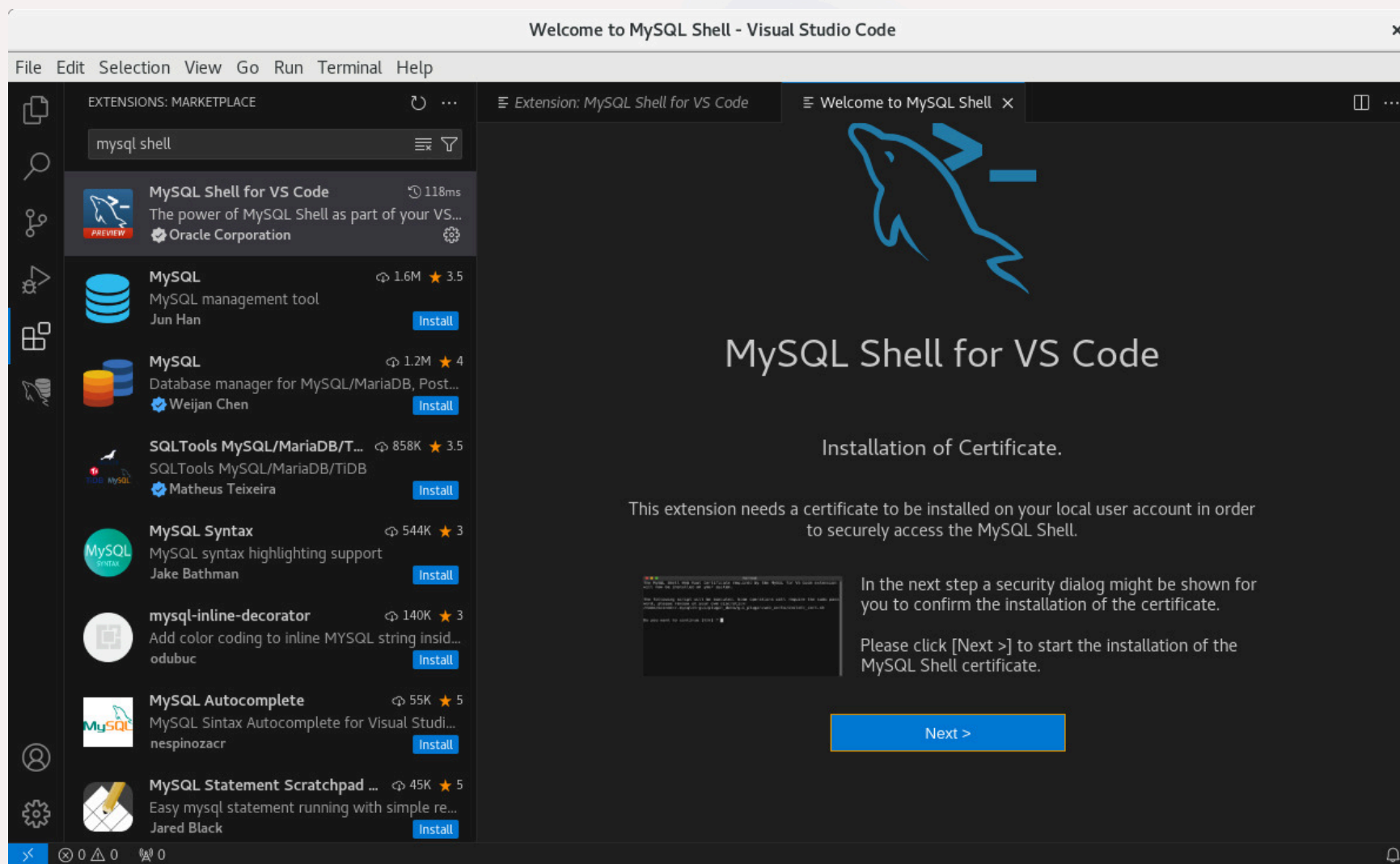
Published 2022-03-23, 22:30:23

id	first_name	last_name	email	phone_number	hire_date	job_id	salary	commission_pct
1	DEHAENE	DEHAENE	DEHAENE.D	0112345678	2000-03-15 00:00:00	SA_REP	12000	0.1
2	BAERENTZEN	BAERENTZEN	BAERENTZEN.B	0112345678	2000-03-15 00:00:00	SA_REP	12000	0.1
3	CHEN	CHEN	CHEN.C	0112345678	2000-03-15 00:00:00	SA_REP	12000	0.1
4	DEHAENE	DEHAENE	DEHAENE.D	0112345678	2000-03-15 00:00:00	SA_REP	12000	0.1
5	BAERENTZEN	BAERENTZEN	BAERENTZEN.B	0112345678	2000-03-15 00:00:00	SA_REP	12000	0.1
6	CHEN	CHEN	CHEN.C	0112345678	2000-03-15 00:00:00	SA_REP	12000	0.1
7	DEHAENE	DEHAENE	DEHAENE.D	0112345678	2000-03-15 00:00:00	SA_REP	12000	0.1

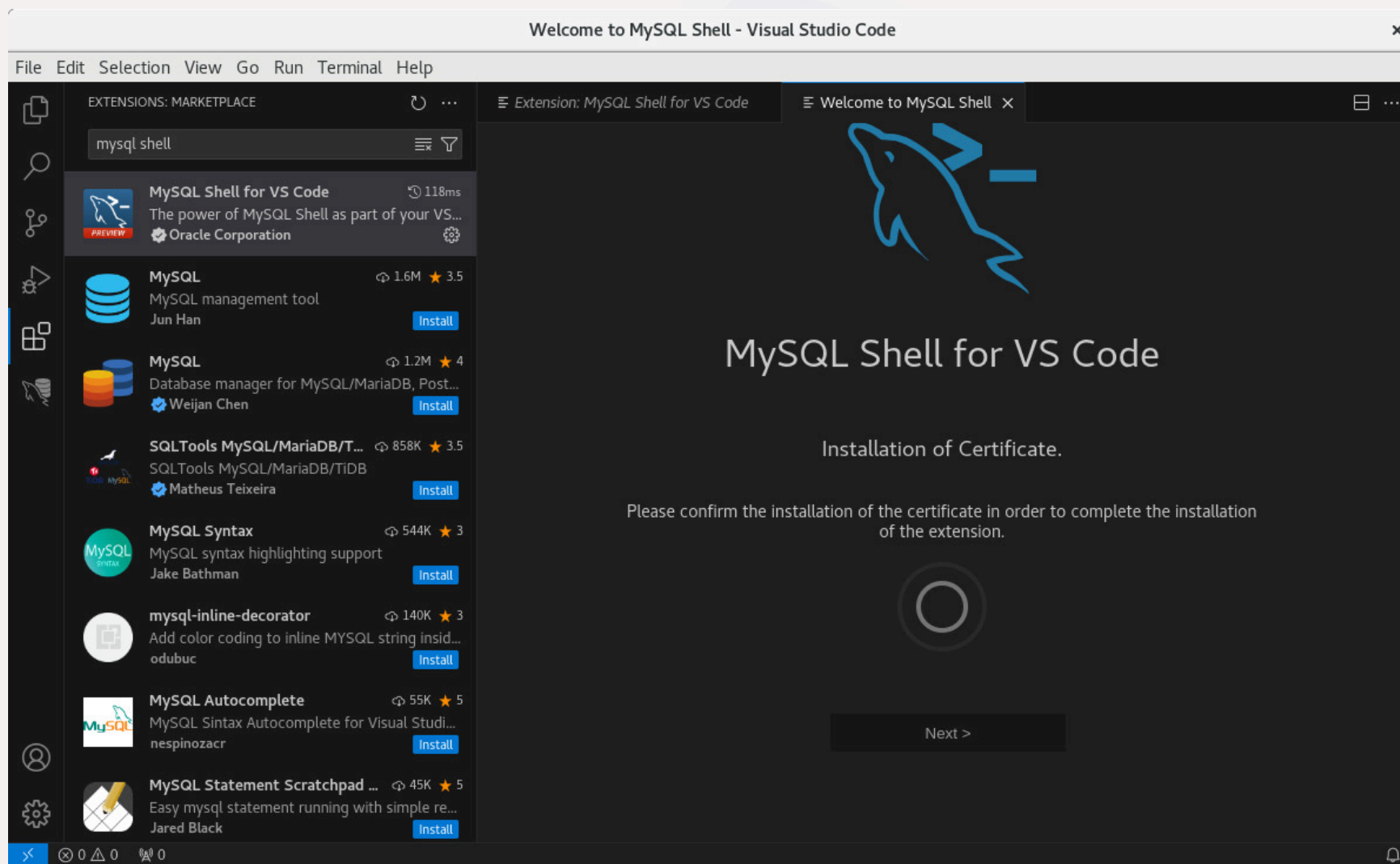
Installation



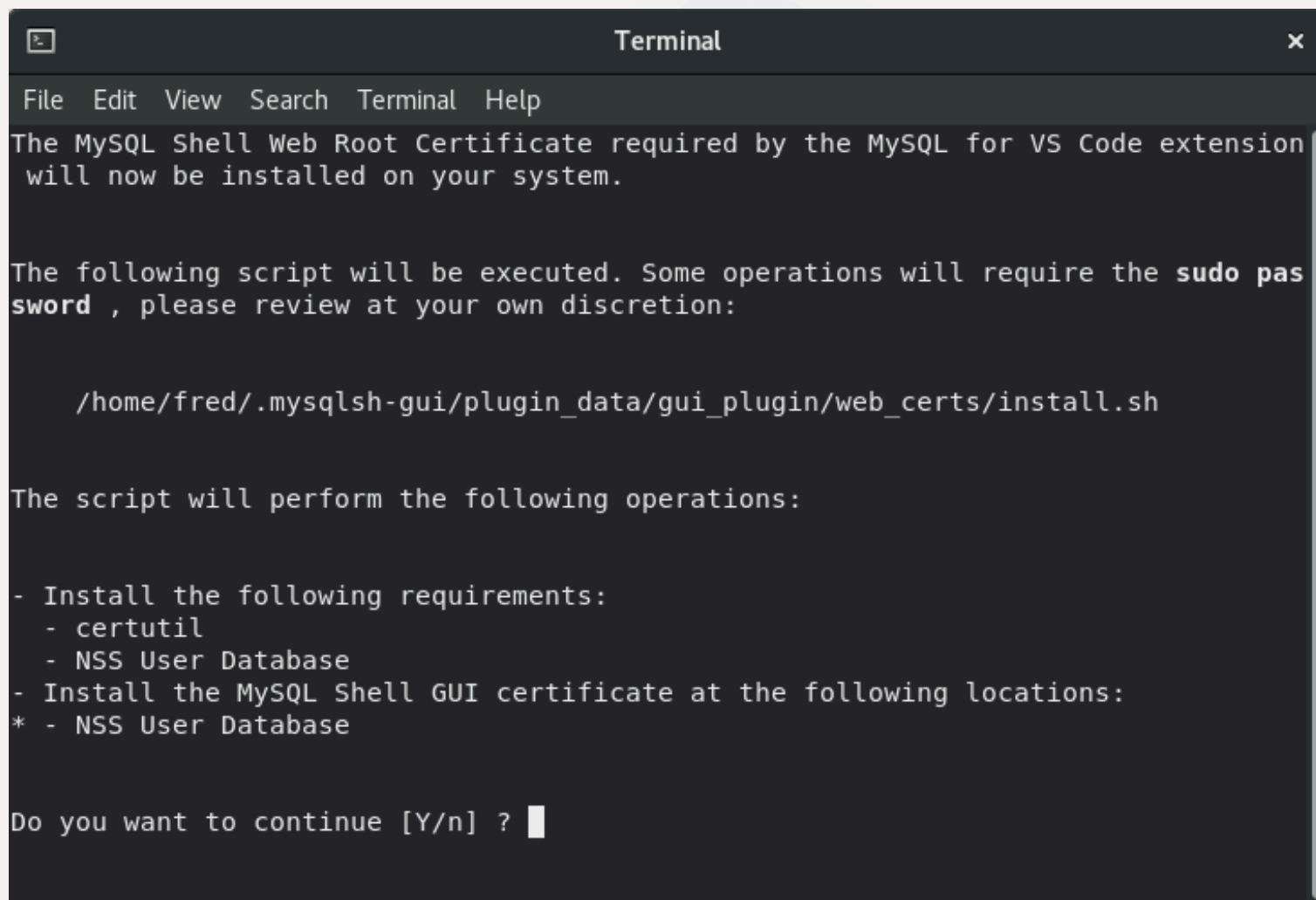
Installation



Installation



Installation



```
Terminal
File Edit View Search Terminal Help
The MySQL Shell Web Root Certificate required by the MySQL for VS Code extension
will now be installed on your system.

The following script will be executed. Some operations will require the sudo pas
sword , please review at your own discretion:

    /home/fred/.mysqlsh-gui/plugin_data/gui_plugin/web_certs/install.sh

The script will perform the following operations:

- Install the following requirements:
  - certutil
  - NSS User Database
- Install the MySQL Shell GUI certificate at the following locations:
* - NSS User Database

Do you want to continue [Y/n] ? █
```

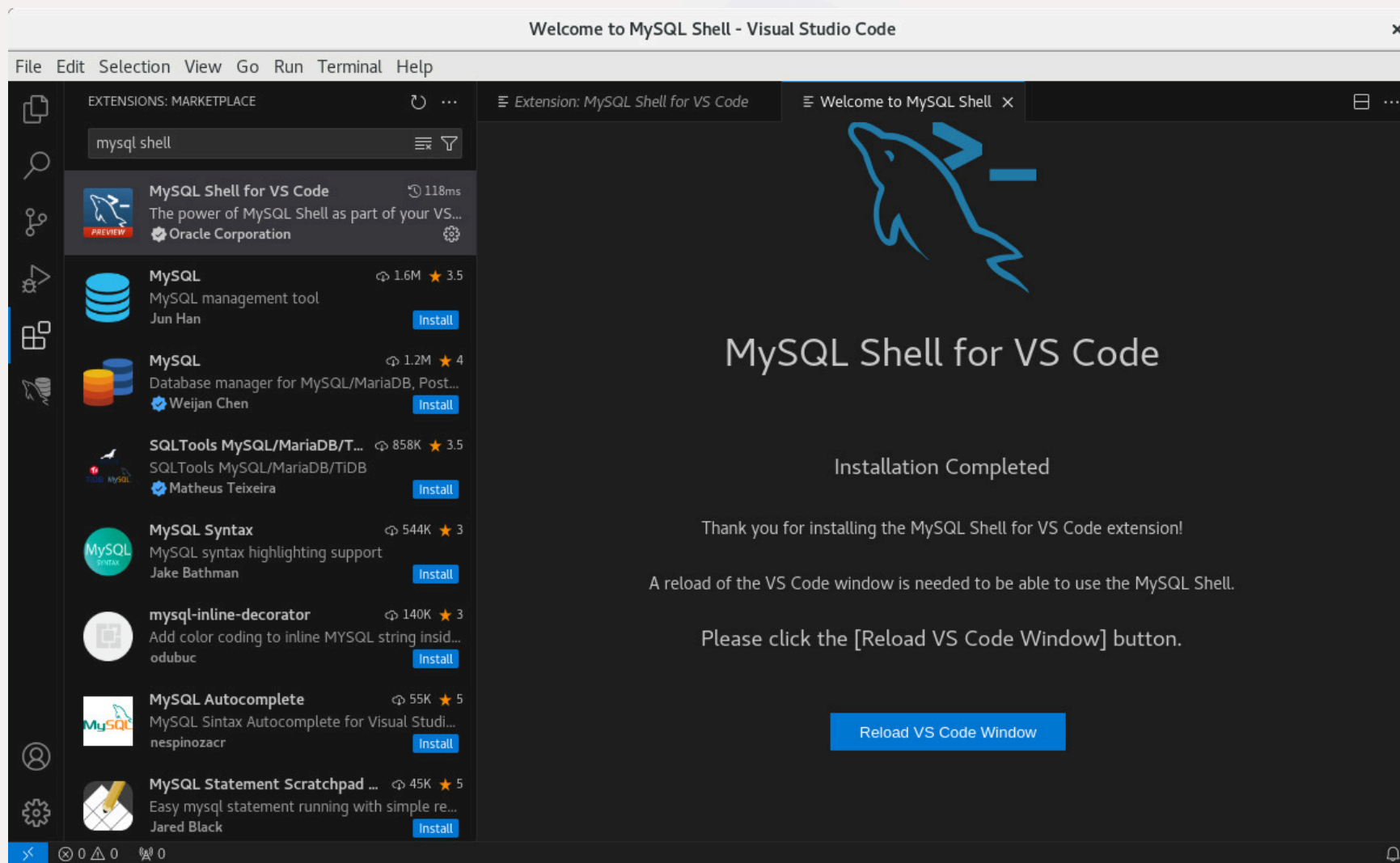
Installation

```
Terminal
File Edit View Search Terminal Help
Running transaction
  Preparing      :                               1/1
  Upgrading      : nss-sysinit-3.90.0-4.el8_9.x86_64 1/5
  Upgrading      : nss-3.90.0-4.el8_9.x86_64        2/5
  Installing     : nss-tools-3.90.0-4.el8_9.x86_64   3/5
  Cleanup        : nss-3.90.0-3.el8_8.x86_64        4/5
  Cleanup        : nss-sysinit-3.90.0-3.el8_8.x86_64 5/5
  Running scriptlet: nss-3.90.0-4.el8_9.x86_64       5/5
  Running scriptlet: nss-sysinit-3.90.0-3.el8_8.x86_64 5/5
  Verifying      : nss-tools-3.90.0-4.el8_9.x86_64   1/5
  Verifying      : nss-3.90.0-4.el8_9.x86_64        2/5
  Verifying      : nss-3.90.0-3.el8_8.x86_64        3/5
  Verifying      : nss-sysinit-3.90.0-4.el8_9.x86_64 4/5
  Verifying      : nss-sysinit-3.90.0-3.el8_8.x86_64 5/5

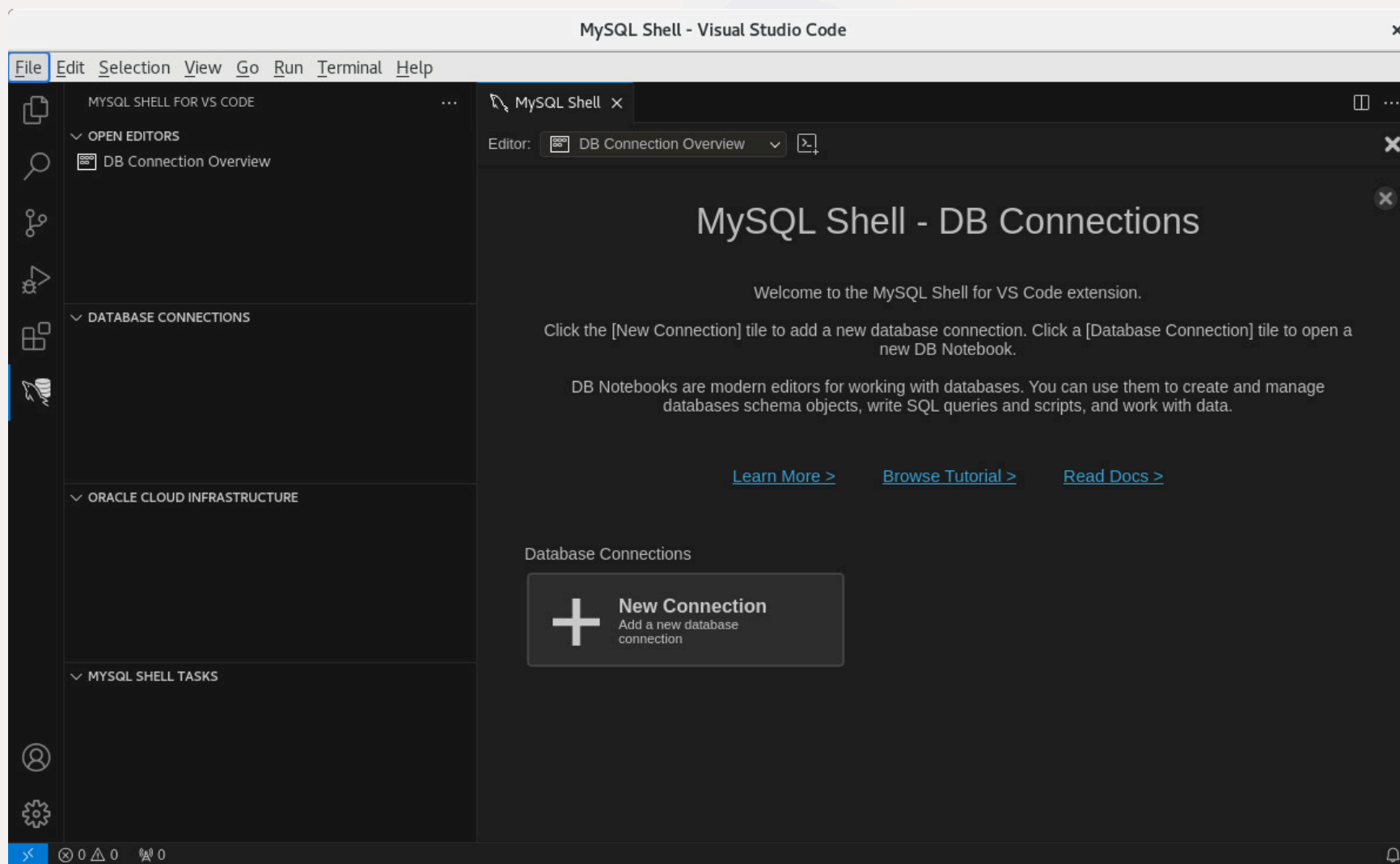
Upgraded:
  nss-3.90.0-4.el8_9.x86_64      nss-sysinit-3.90.0-4.el8_9.x86_64
Installed:
  nss-tools-3.90.0-4.el8_9.x86_64

Complete!
Verifying existence of the NSS database...
Certificate has been installed successfully!
Press any key to continue...
```

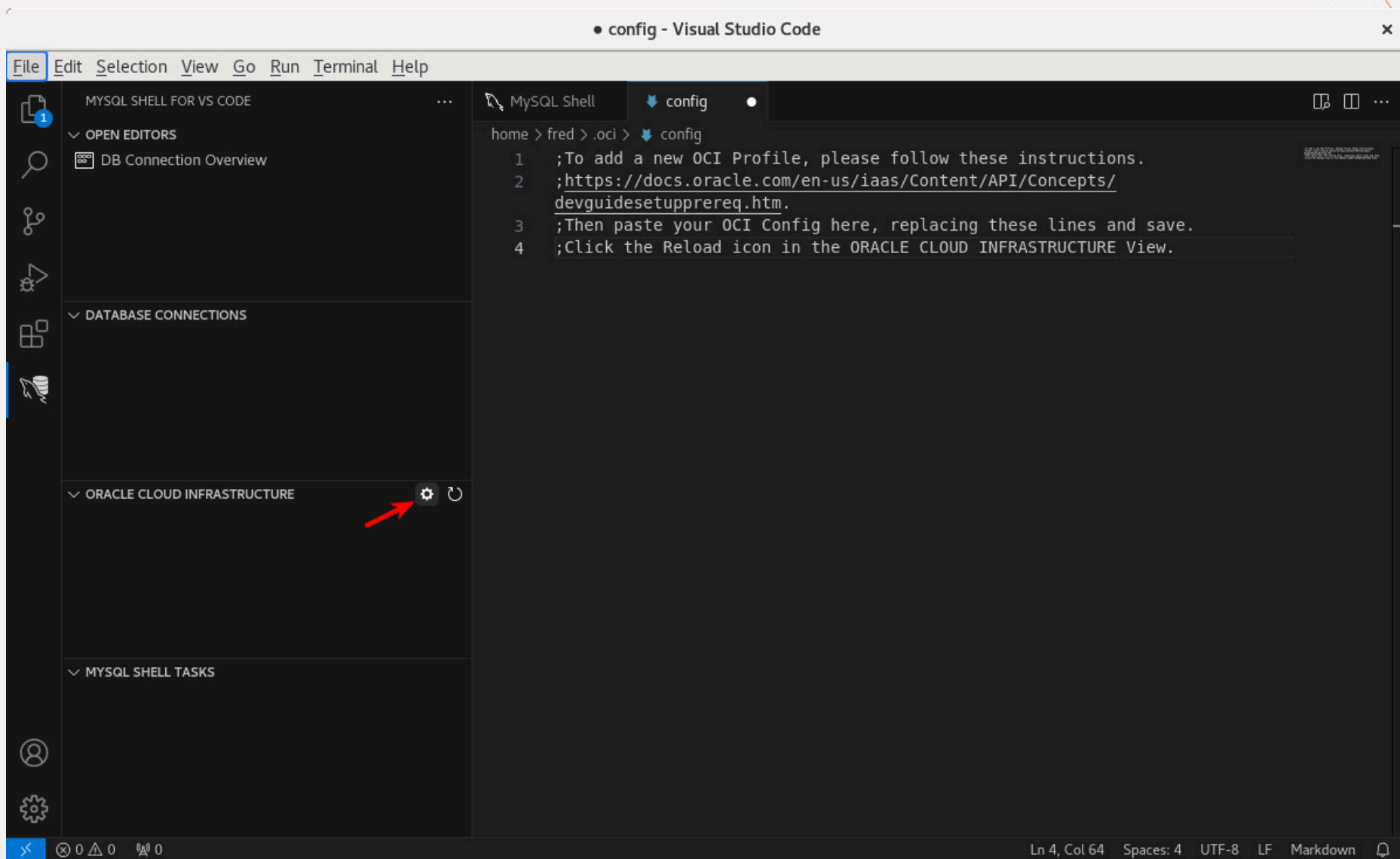
Installation



Installation



OCI Setup



OCI Setup

Oracle Cloud Cloud Classic Search resources, services, documentation, and Marketplace US East (Ashburn) [Icons]

MySQL Community Manager h [Redacted]

U ACTIVE

Edit my profile Change password Link support account Security More actions

User information Tags Capabilities

OCID: [Redacted] [Show](#) [Copy](#) Email: [Redacted]@oracle.com

Username: fdescamp Instant messaging address: -

Prefix: - Home phone number: -

First name: - Mobile phone number: -

Middle name: - Federated: No

Last name: [Redacted] My Oracle Support Account: -

Suffix: - Authenticated by: [Redacted]

> Work information

> Other information

Resources

My groups

Integrated applications

API keys

Request access to a new group

Terms of Use and Privacy Cookie Preferences

Copyright © 2024, Oracle and/or its affiliates. All rights reserved.

OCI Setup

ORACLE Cloud

Cloud Classic >

Search resources, services, documentation, and Marketplace

US East (Ashburn) v

First name: -

Middle name: -

Last name: [REDACTED]

Suffix: -

Mobile phone number: -

Federated: No

My Oracle Support Account: -

Authenticated by [REDACTED]

> Work information

> Other information

Resources

My groups

Integrated applications

API keys

Auth tokens

Customer secret keys

My access tokens

My requests

My consents

My oauth 2.0 client credentials

SMTP credentials

Add API key

Delete

<input type="checkbox"/>	Fingerprint	Created	
<input type="checkbox"/>	[REDACTED]	Fri, Aug 26, 2022, 14:10:28 UTC	⋮
<input type="checkbox"/>	[REDACTED]	Tue, Sep 27, 2022, 15:23:25 UTC	⋮
<input type="checkbox"/>	67: [REDACTED] :5c	Tue, Jan 9, 2024, 17:20:40 UTC	<div><div><u>View configuration file</u></div><div>Delete</div></div>

0 selected

Show all

Terms of Use and Privacy

Cookie Preferences

Copyright © 2024, Oracle and/or its affiliates. All rights reserved.

OCI Setup

The screenshot shows the Oracle Cloud console interface. A modal dialog titled "Configuration file preview" is open, displaying the configuration snippet for an API key. The background shows the "API keys" section of the console, with a list of API keys and a "Showing 3 API keys" message.

Configuration file preview [Help](#)

Note: This configuration file snippet includes the basic authentication information you'll need to use the SDK, CLI, or other OCI developer tool. Paste the contents of the text box into your `~/.oci/config` file and update the `key_file` parameter with the file path to your private key. [Learn more](#)

Select API key fingerprint

67 [redacted] :5c

Configuration file preview *Read-only*

```
[DEFAULT]
user=ocid1.user.[redacted]
fingerprint=67[redacted]:5c
tenancy=ocid1.tenancy[redacted]
region=us-ashburn-1
key_file=<path to your private keyfile> # TODO
```

[Copy](#)

Paste the contents of the text box into your `~/.oci/config` file.

Close

OCI Setup

The screenshot shows the Visual Studio Code interface with the MySQL Shell for VS Code extension. The left sidebar contains the following sections:

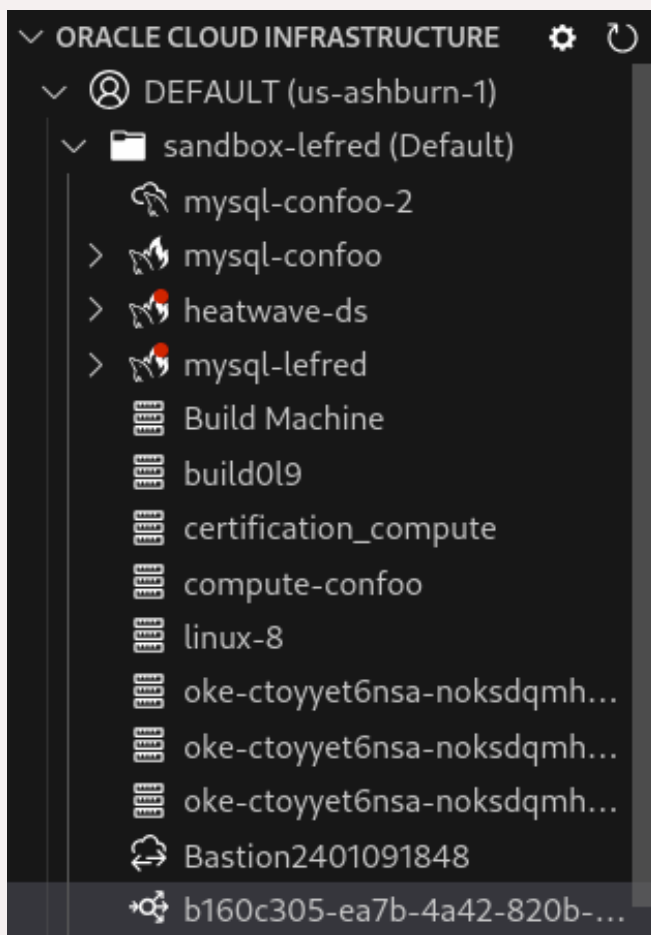
- OPEN EDITORS**: DB Connection Overview
- DATABASE CONNECTIONS**:
 - performance_schema
 - piday
 - Tables
 - temperature_history
 - Columns
 - Indexes
 - Foreign Keys
- ORACLE CLOUD INFRASTRUCTURE**:
 - mysql-confoo-2
 - mysql-confoo
 - heatwave-ds
 - HeatWave Cluster (1 node)
 - mysql-lefred
 - Build Machine
 - build019
- MYSQL SHELL TASKS**:
 - Start DB System (done)
 - Dump Schema carsten to Disk (done)
 - Loading Dump test from Disk (error)
 - Dump Schema sbtest to Disk (done)
 - Loading Dump sbtest from Disk (error)
 - Dump Schema sbtest to Disk (error)



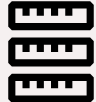



The main editor shows the configuration file for the MySQL Shell. The content is as follows:

```
home > fred > .oci > config
1 ;To add a new OCI Profile, please follow these instructions.
2 ;https://docs.oracle.com/en-us/iaas/Content/API/Concepts/devguidesetupprereq.htm.
3 ;Then paste your OCI Config here, replacing these lines and save.
4 ;Click the Reload icon in the ORACLE CLOUD INFRASTRUCTURE View.
5 [DEFAULT]
6 user=ocid1.user.
7 fingerprint=67:
8 tenancy=ocid1.tenancy.
9 region=us-ashburn-1
10 key_file=/home/fred/OCI/
11
```

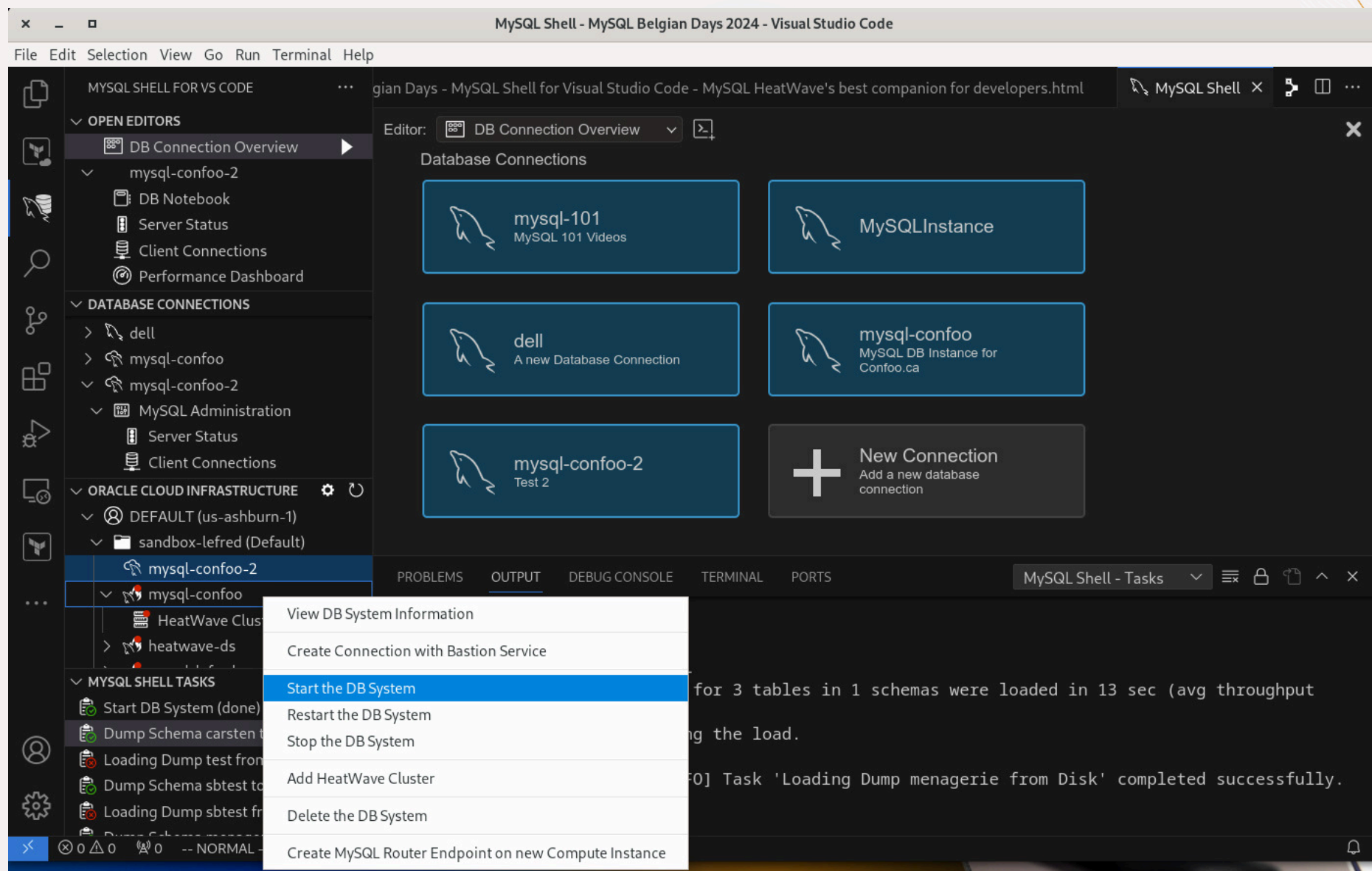
The bottom status bar shows the current file is "MySQL Belgian Days - MySQL Shell for Visual Studio Code - MySQL HeatWave's bes" and the cursor is at line 11, column 1.

List of instances



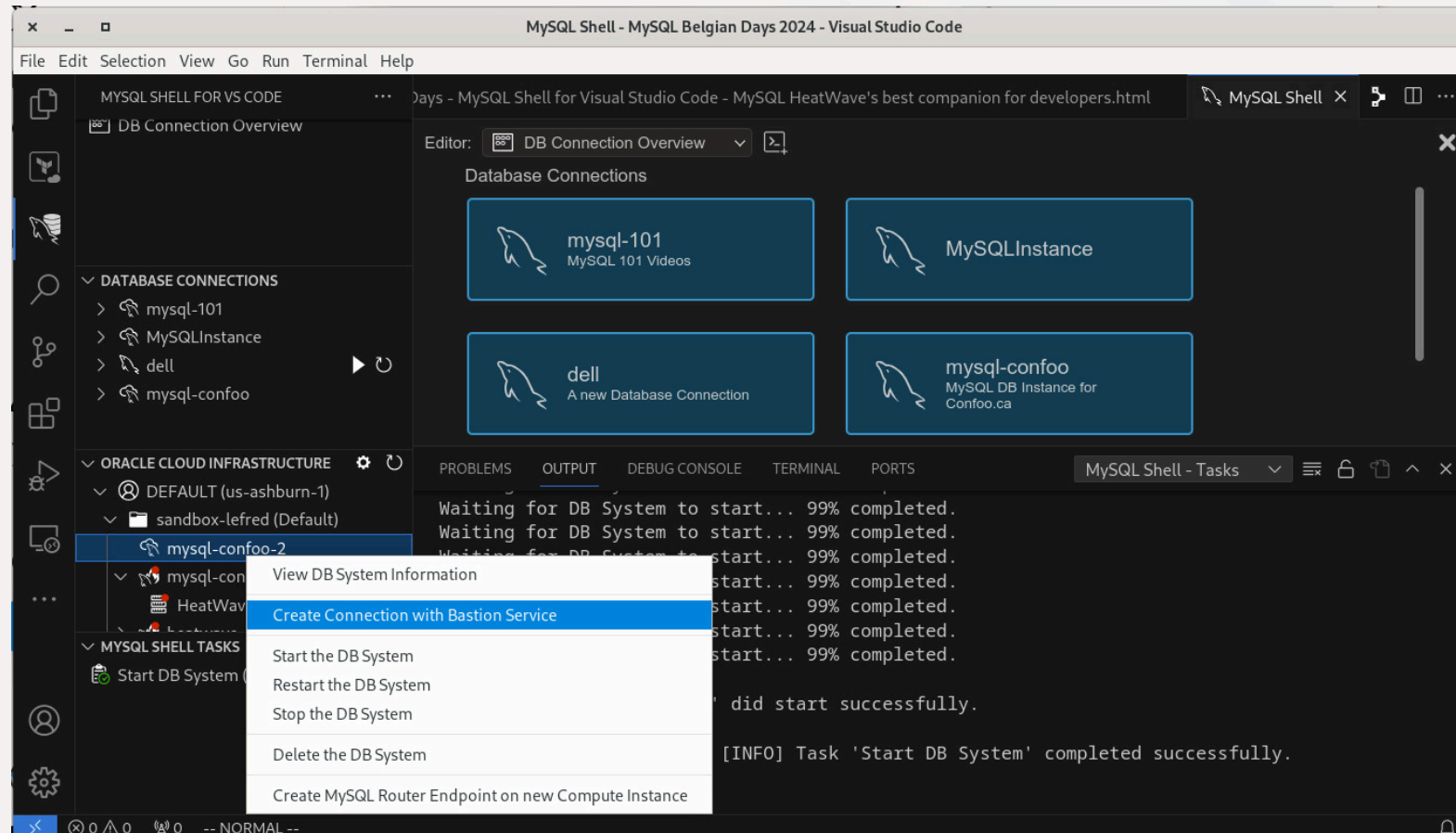
-  *MySQL DB System*
-  *MySQL DB System with HeatWave Cluster*
-  *Compute Instance / HeatWave Cluster Node*
-  *Bastion Host*
-  *OCI Load Balancer*
-  *MySQL Router*

MySQL HeatWave Instance Operations

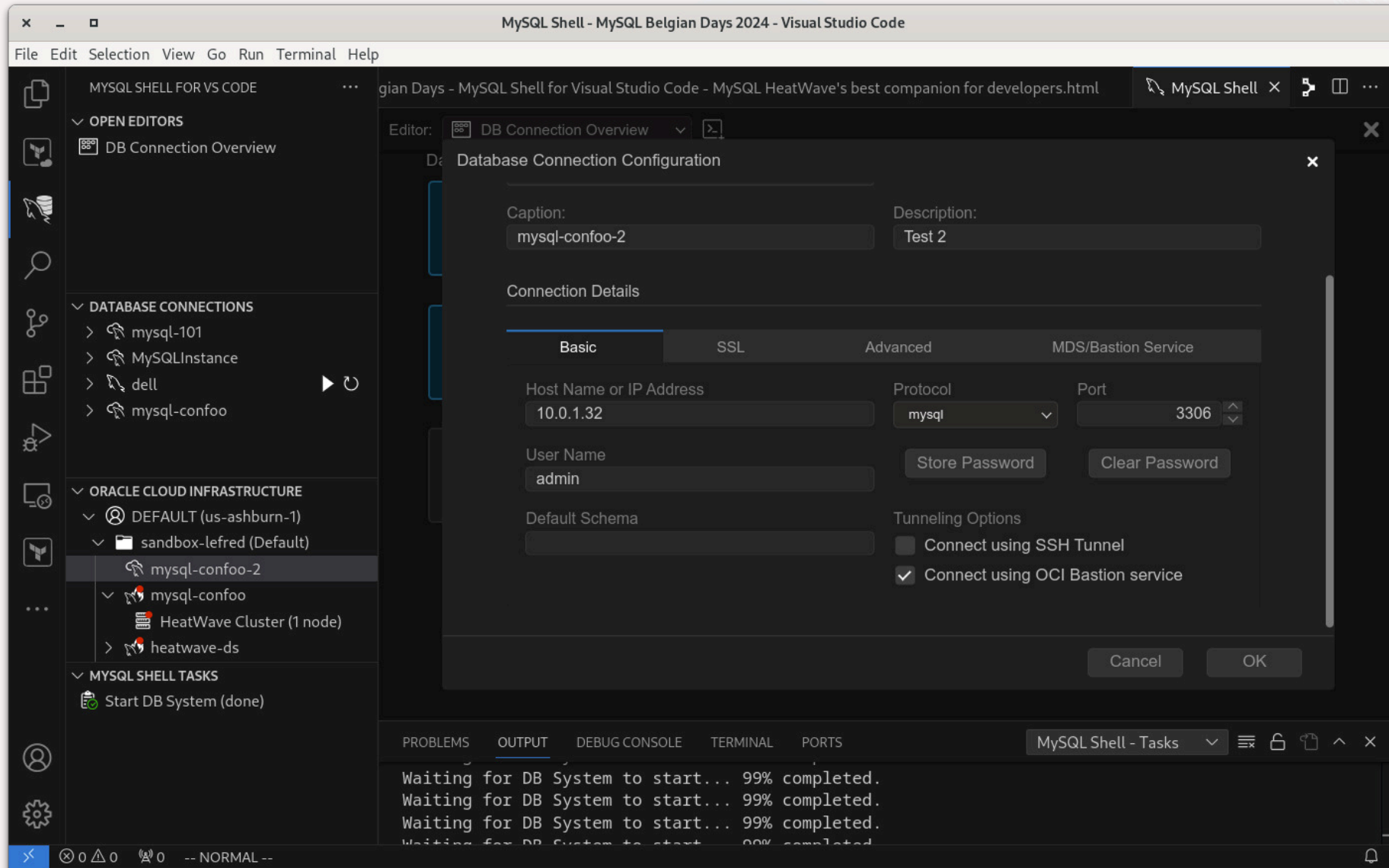


Connection

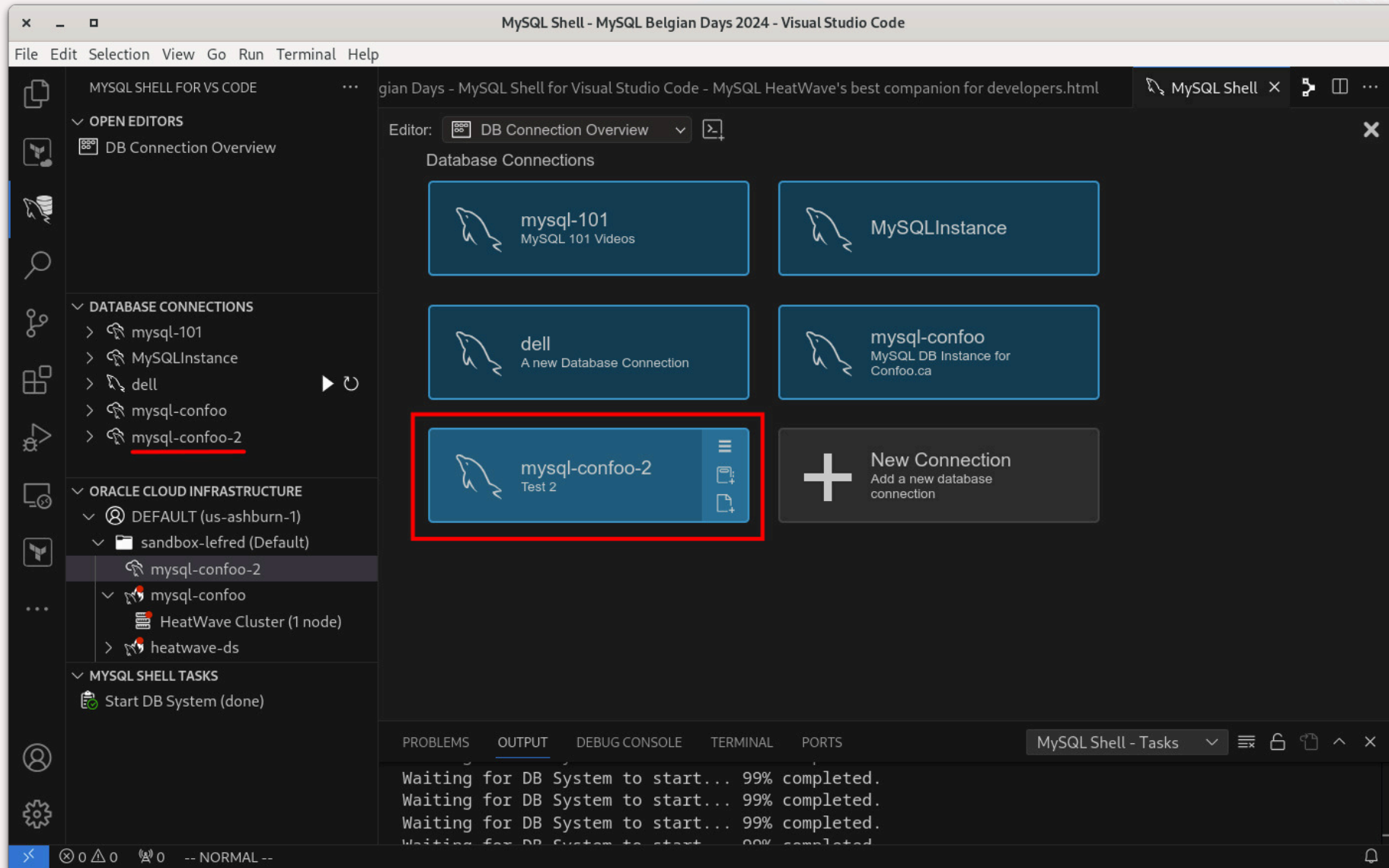
*As Developer, the easiest way to connect to your **MySQL** DB System if you don't have any VPN to your **OCI** tenancy, it to use the **Bastion Service**:*



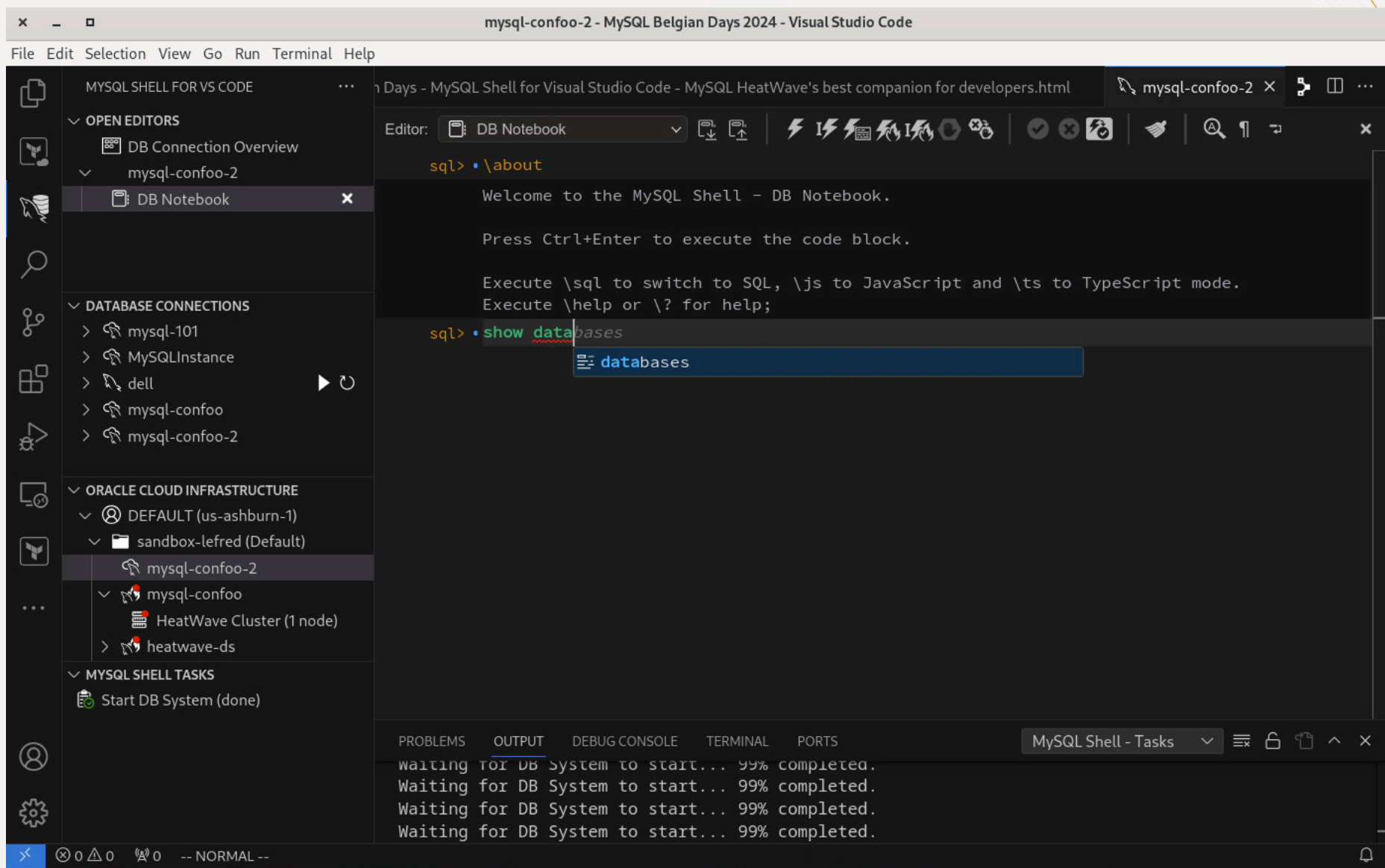
Connection - Bastion Service



Connection - Bastion Service



Connection - Bastion Service



Connection - Bastion Service

The screenshot displays the Visual Studio Code interface for the 'mysql-confoo-2' connection. The left sidebar shows the 'DATABASE CONNECTIONS' section with 'mysql-confoo-2' selected. The main editor shows the 'DB Notebook' with a welcome message and a list of databases. The bottom panel shows the 'MySQL Shell - Tasks' output.

Database Connections:

- mysql-confoo-2
 - MySQL Administration
 - information_schema
 - mysql_audit
 - performance_schema
 - sys

MySQL Shell Tasks:

- Start DB System (done)

DB Notebook Output:

```
sql> \about

Welcome to the MySQL Shell - DB Notebook.

Press Ctrl+Enter to execute the code block.

Execute \sql to switch to SQL, \js to JavaScript and \ts to TypeScript mode.
Execute \help or \? for help;

sql> show databases

Database
information_schema
mysql
mysql_audit
performance_schema
sys

OK, 5 records retrieved in 114.378ms
```

MySQL Shell - Tasks Output:

```
waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.
```



MySQL Shell for Visual Studio Code

MySQL Administration

Administration - Server Status

The screenshot shows the MySQL Server Status window in Visual Studio Code. The window is titled "mysql-conf00-2 - MySQL Belgian Days 2024 - Visual Studio Code". The left sidebar contains a tree view with the following sections:

- MYSQL SHELL FOR VS CODE
 - OPEN EDITORS
 - DB Connection Overview
 - mysql-conf00-2
 - DB Notebook
 - Server Status (selected)
- DATABASE CONNECTIONS
 - mysql-conf00-2
 - MySQL Administration
 - Server Status (selected)
 - Client Connections
 - Performance Dashboard
 - information_schema
- ORACLE CLOUD INFRASTRUCTURE
 - DEFAULT (us-ashburn-1)
 - sandbox-lefred (Default)
 - mysql-conf00-2 (selected)
- mysql-conf00
 - HeatWave Cluster (1 node)
 - heatwave-ds

- MYSQL SHELL TASKS
- Start DB System (done)

The main content area displays the "Server status" for "MySQL Enterprise - Cloud". It is divided into four sections:

- Main Settings**
 - Host: ycq4lp05uvskfwdk
 - Socket: /var/run/mysqld/mysqld.sock
 - Port: 3306
 - Version: MySQL Enterprise - Cloud(8.0.35-u2-cloud)
 - Compiled For: Linux
 - Configuration File: none
 - Running Since: 0 day, 0 hours, 16 minutes
- Server Directories**
 - Base Directory: /usr/
 - Data Directory: /db/data/
 - Plugins directory: /usr/lib64/mysql/plugin/
 - Tmp Directory: /db/tmp
 - Error Log: ☒ on /db/log/error.log
 - General Log: ☒ off [Stored in database]
 - Slow Query Log: ☒ off [Stored in database]
- Server Features**
 - Performance Schema: ☒ on
 - Thread Pool: ☒ on
 - Memcached Plugin: none
 - Semisync Replication Plugin: none
 - PAM Authentication: ☐ none
 - Password Validation: ☐ none
- Server SSL**
 - SSL CA: /db/metadata/pki/customer_endpoint/ca.pem
 - SSL CA Path: /db/metadata/pki/customer_endpoint/ssl-ca-path
 - SSL Cert: /db/metadata/pki/customer_endpoint/server-cert.pem
 - SSL Cipher: DHE-RSA-AES256-GCM-SHA384
 - SSL CRL: /db/metadata/pki/customer_endpoint/crl.pem

The bottom status bar shows "PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS" and "MySQL Shell - Tasks". The terminal output displays:

```
waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.
```


Administration - Client Connections

The screenshot displays the MySQL Shell for VS Code interface. The left sidebar shows the 'Client Connections' view selected under the 'mysql-confoo-2' instance. The main editor area shows the 'Client Connection List' with a table of active connections. The bottom status bar indicates the MySQL Shell is running.

mysql-confoo-2 - MySQL Belgian Days 2024 - Visual Studio Code

File Edit Selection View Go Run Terminal Help

Editor: Client Connections Refresh Rate: 5 seconds

Threads Connected: 6 Threads Running: 2 Threads Created: 20 Threads Cached: 0
Rejected (over limit): Total Connections: 35 Connection Limit: 0
Aborted Connections: 0 Errors: 0 Aborted Clients: 0

Client Connection List

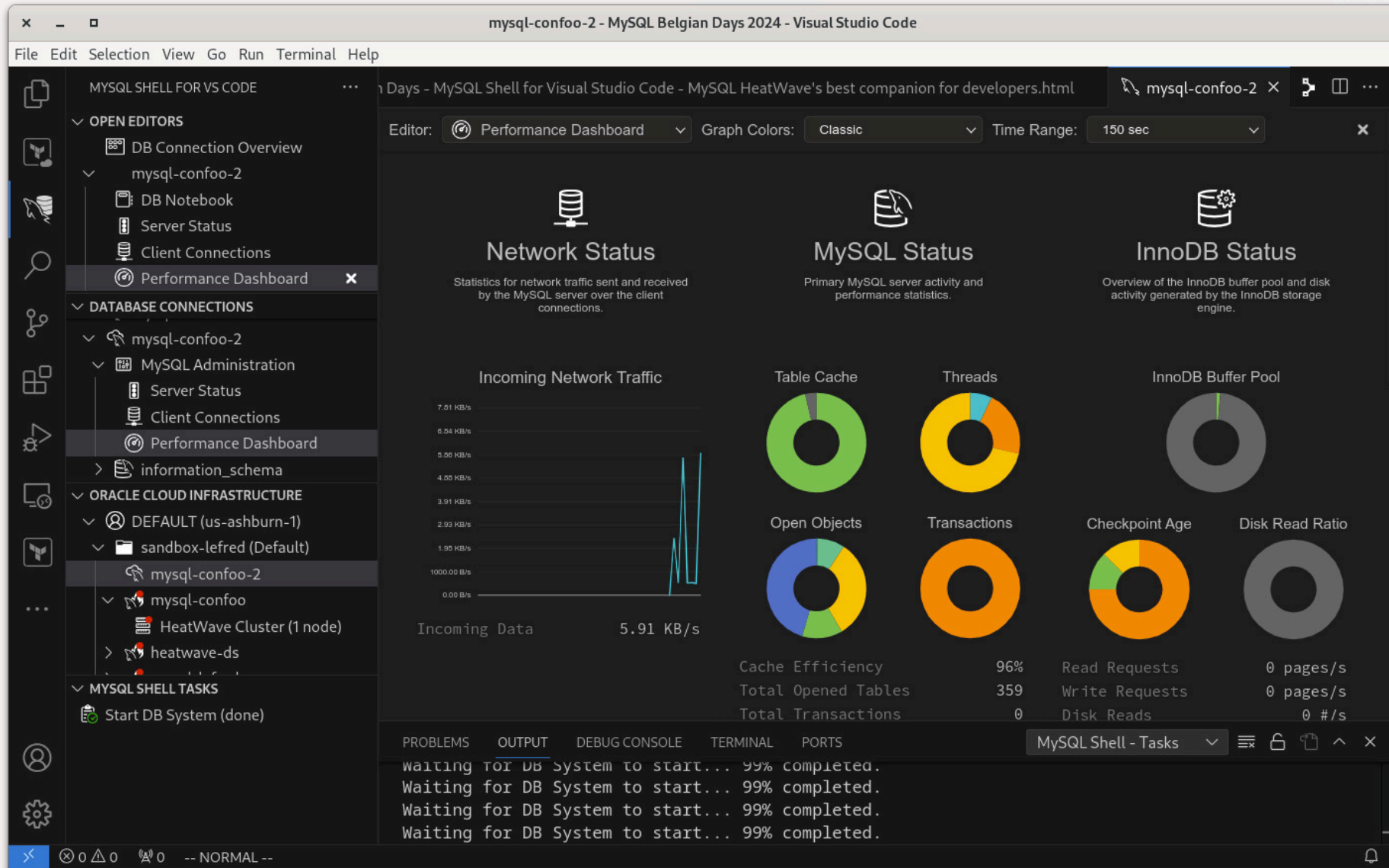
I...	User	Host	Database	Command	T...	State
9	event_scheduler	localhost	NULL	Connect	296	Waiting for next activ..
10	NULL	NULL	NULL	Daemon	988	Suspending
31	admin	10.0.1.170	NULL	Query	0	executing

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS MySQL Shell - Tasks

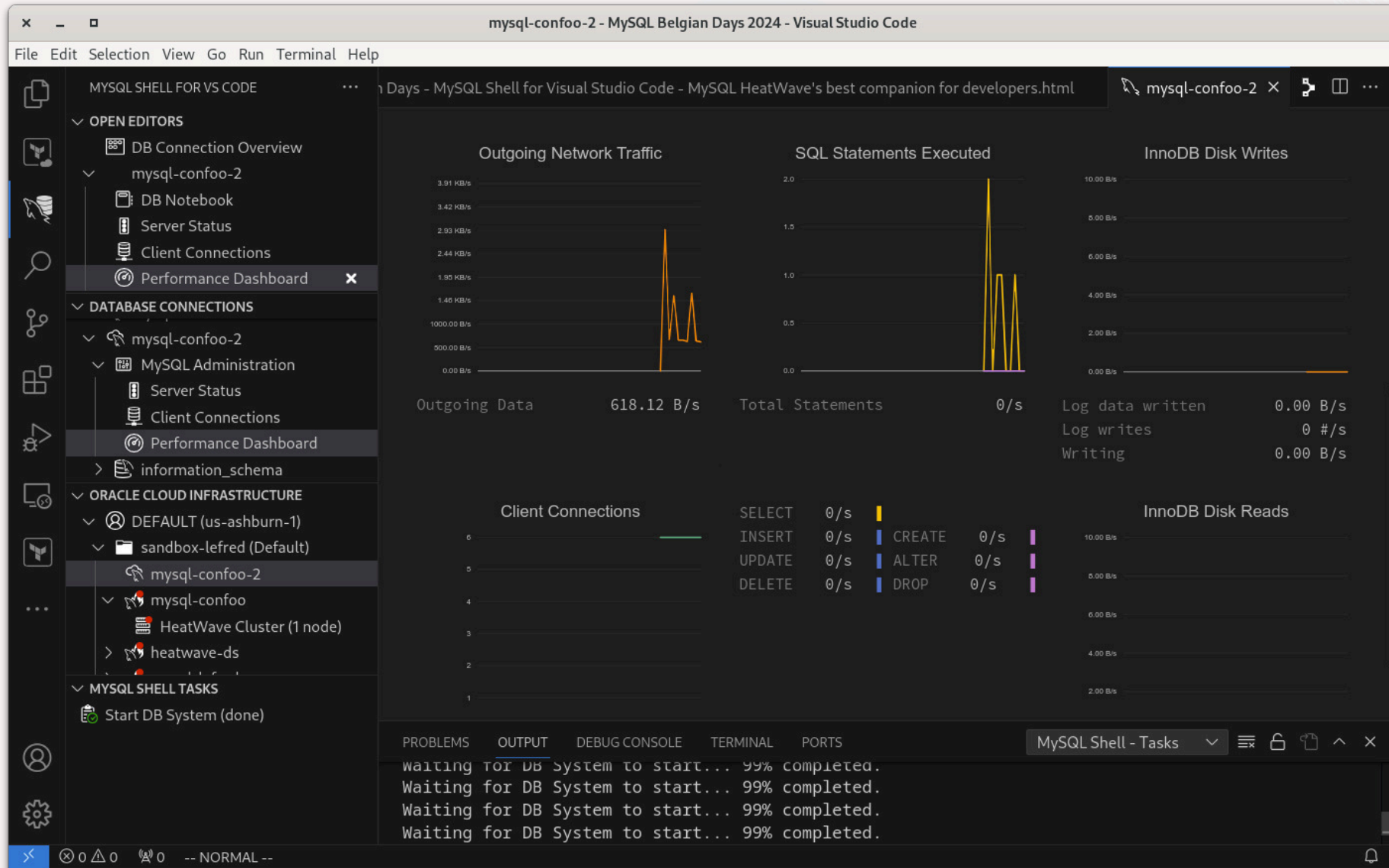
waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.
Waiting for DB System to start... 99% completed.

-- NORMAL --

Administration - Performance Dashboard



Administration - Performance Dashboard



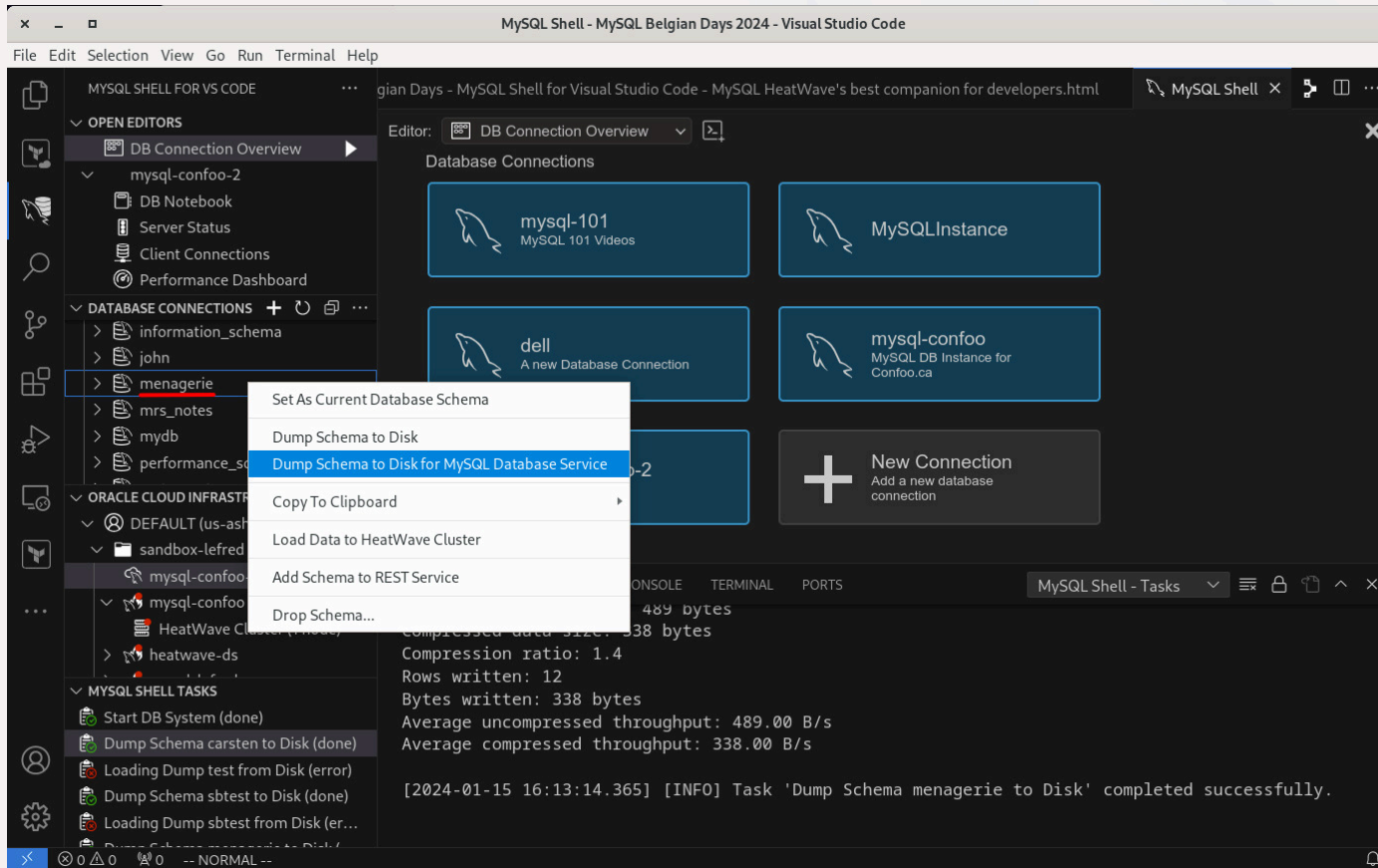


MySQL Shell for Visual Studio Code

Dump & Load Data

Dump a Schema locally to be imported in OCI

*It's possible to dump a schema from an instance locally to be imported to a **MySQL HeatWave** DB System in **OCI**:*

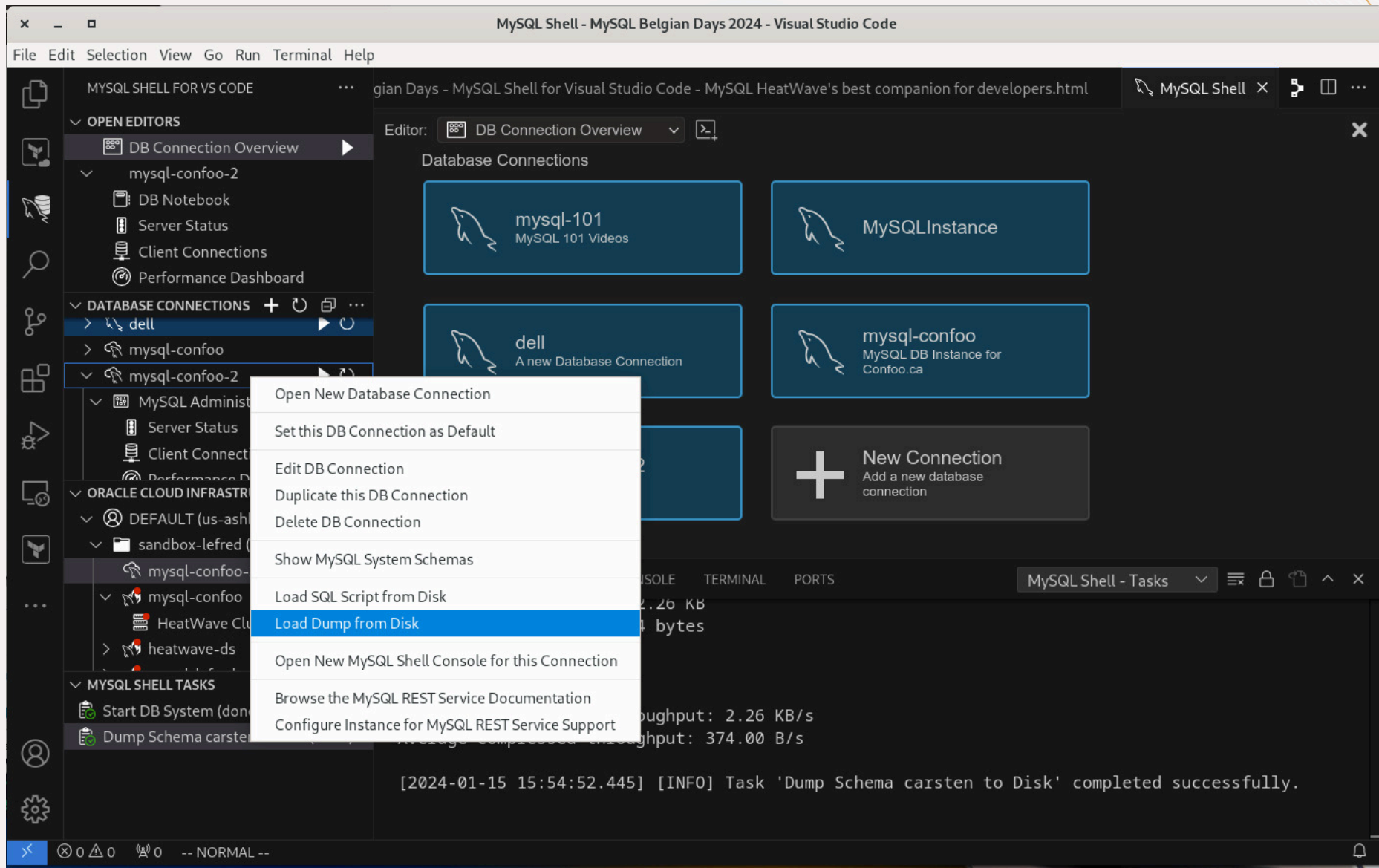


Dump a Schema locally to be imported in OCI

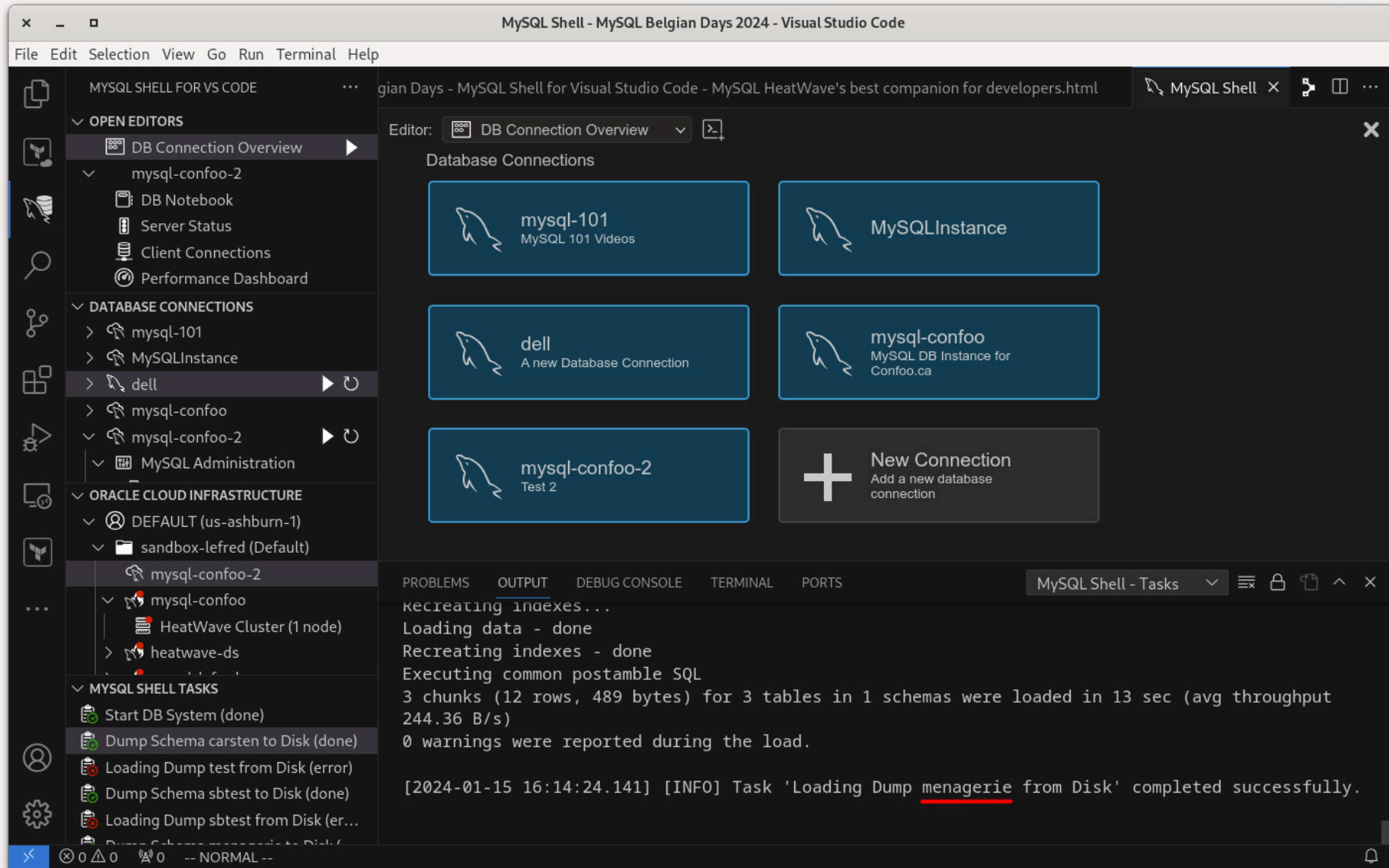
It creates a *MySQL Shell* parallel dump using the following options:

```
"options": {  
  "compatibility": [  
    "create_invisible_pks",  
    "force_innodb",  
    "skip_invalid_accounts",  
    "strip_definers",  
    "strip_restricted_grants",  
    "strip_tablespaces"  
  ],  
  "ocimds": true  
}
```


Load the Dump to a DB System



Load the Dump to a DB System



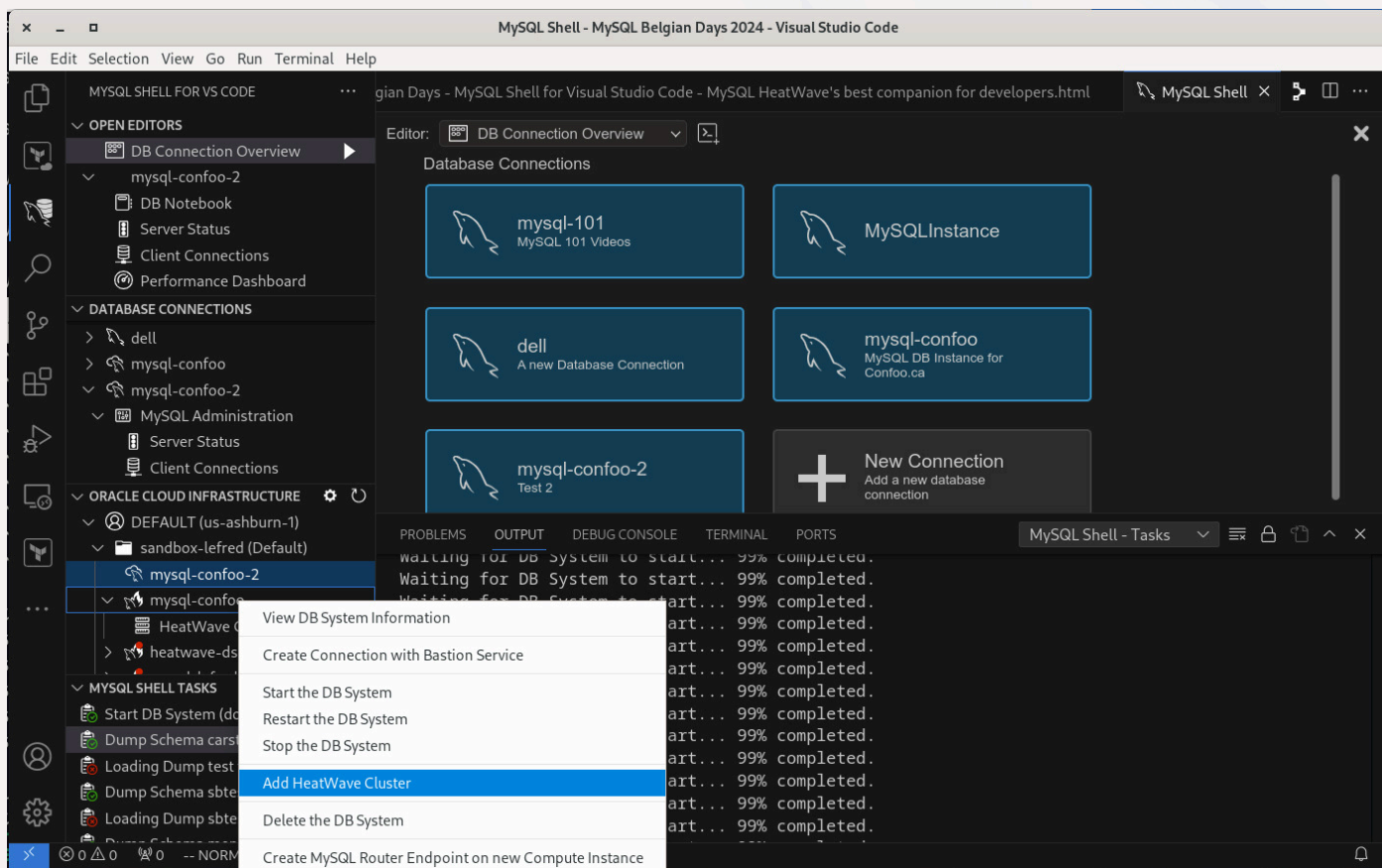


MySQL Shell for Visual Studio Code

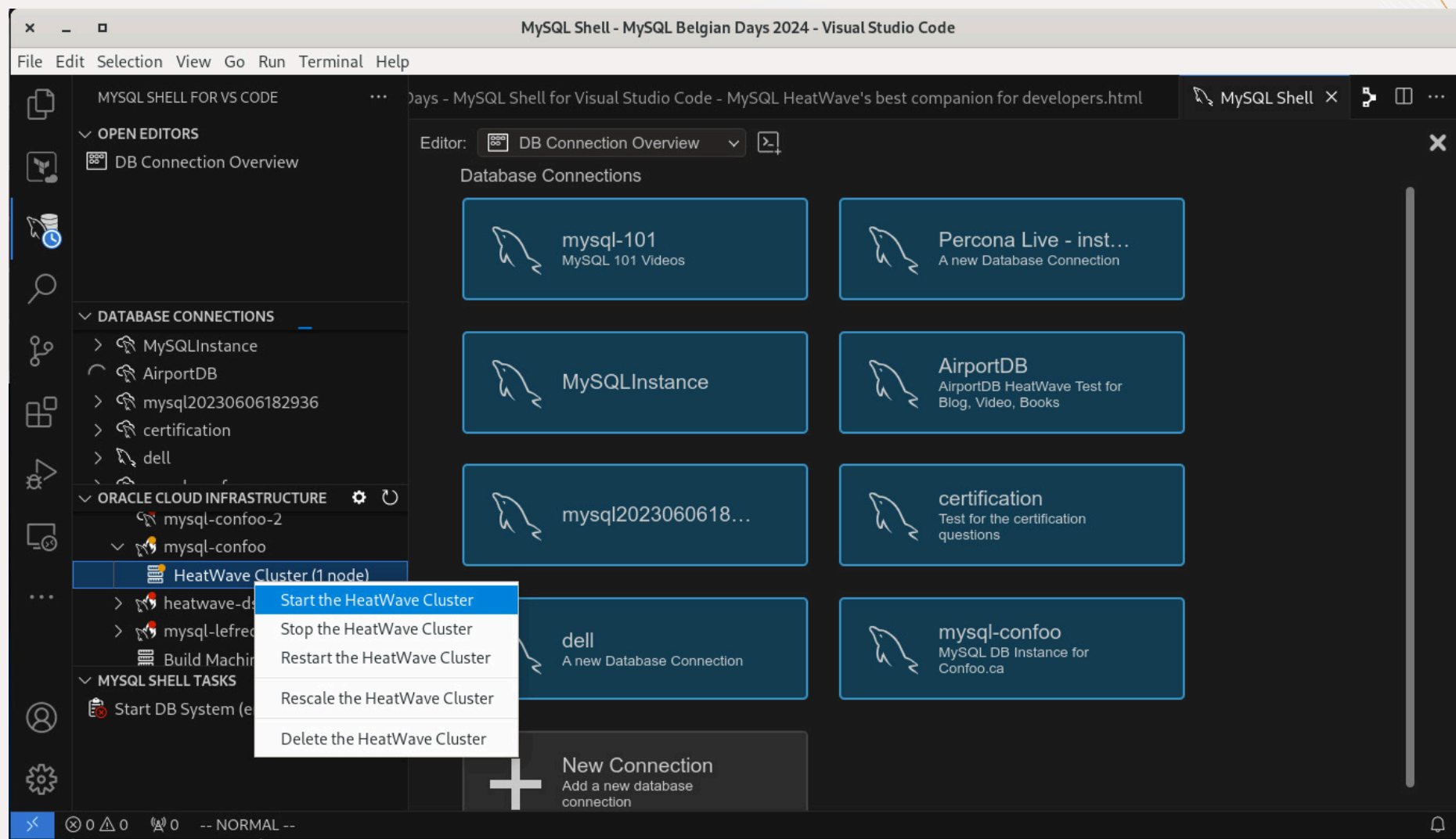
MySQL HeatWave Cluster

MySQL HeatWave Cluster

If the instance's shape supports *MySQL HeatWave Cluster*, the cluster can be added from the *MySQL Shell* for Visual Studio Code's interface:



MySQL HeatWave Cluster - Operations



MySQL HeatWave Cluster - Rescale

The screenshot displays the MySQL Shell interface within Visual Studio Code. The title bar reads "MySQL Shell - MySQL Belgian Days 2024 - Visual Studio Code". The interface includes a menu bar (File, Edit, Selection, View, Go, Run, Terminal, Help) and a sidebar with various toolbars and panels.

The "OPEN EDITORS" panel on the left shows the "DB Connection Overview" tab. The "DATABASE CONNECTIONS" panel on the right displays a grid of connections: "mysql-101", "MySQLInstance", "dell", "mysql-confoo", and "mysql-confoo-2". A "New Connection" button is also visible.

The "PROBLEMS" panel at the bottom shows a list of messages, including "waiting for DB System to start... 99% completed." and "Waiting for DB System to start... 99% completed."

A context menu is open over the "HeatWave" icon in the sidebar, showing the following options:

- Start the HeatWave Cluster
- Stop the HeatWave Cluster
- Restart the HeatWave Cluster
- Rescale the HeatWave Cluster**
- Delete the HeatWave Cluster





Loading Data to HeatWave Cluster

*If you want to load data to **HeatWave Cluster**, the manual operation is the following to use the Auto Parallel Load command:*

```
call sys.heatwave_load(JSON_ARRAY('menagerie'), NULL);
```


Loading Data to HeatWave Cluster

*If you want to load data to **HeatWave Cluster**, the manual operation is the following to use the Auto Parallel Load command:*

```
call sys.heatwave_load(JSON_ARRAY('menagerie'), NULL);
```

and I forgot about the options :-)

Loading Data to HeatWave Cluster

*If you want to load data to **HeatWave Cluster**, the manual operation is the following to use the Auto Parallel Load command:*

```
call sys.heatwave_load(JSON_ARRAY('menagerie'), NULL);
```

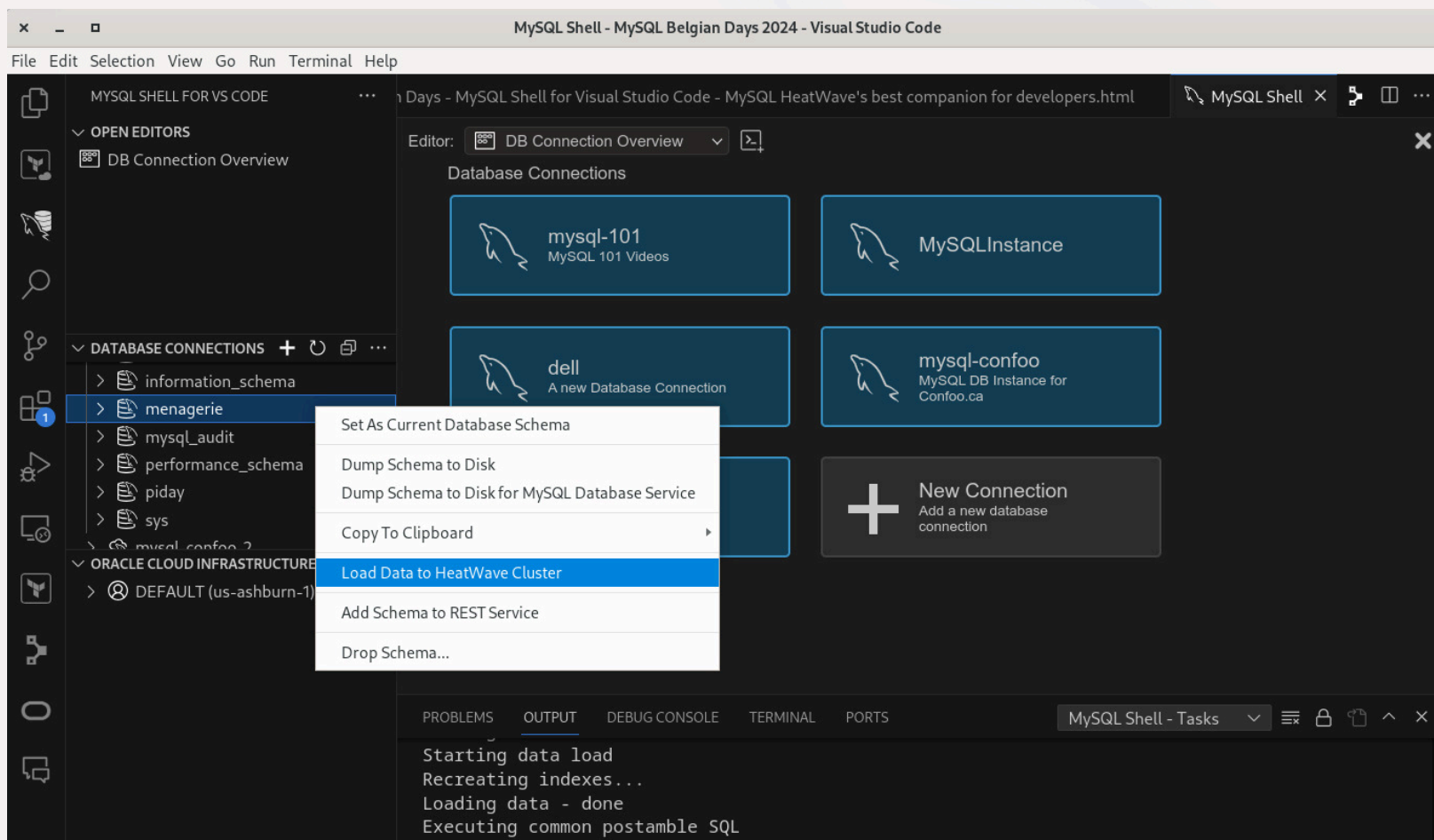
and I forgot about the options :-)

Use this command to get the full help:

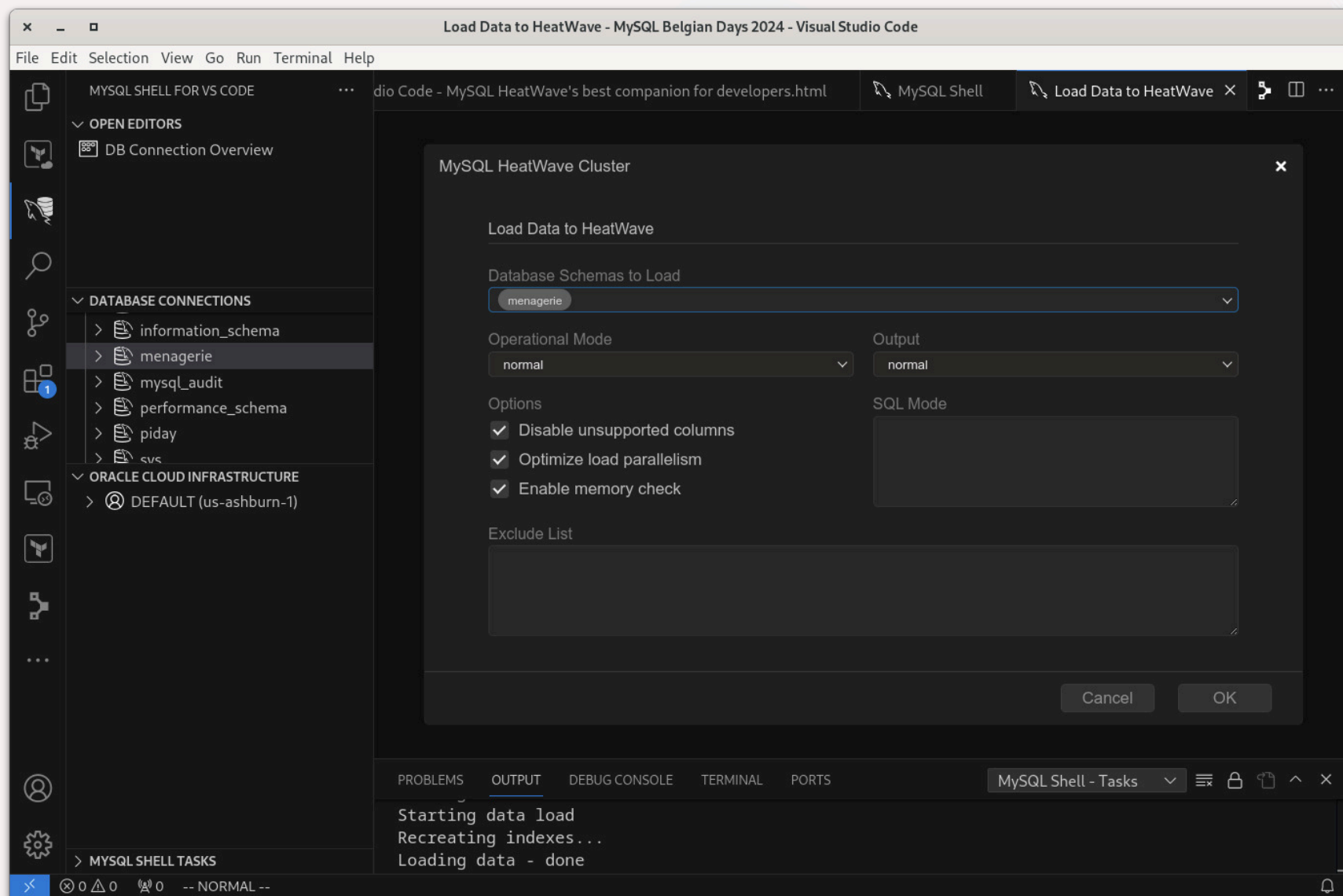
```
CALL sys.heatwave_load(JSON_ARRAY("sys"), JSON_OBJECT("output", "help"));
```

Loading Data to HeatWave Cluster

Or use *MySQL Shell* for VS Code:



Loading Data to HeatWave Cluster



Loading Data to HeatWave Cluster

The screenshot shows the MySQL Shell interface within Visual Studio Code. The left sidebar displays the 'DATABASE CONNECTIONS' tree with the 'menagerie' database selected. The main editor area shows the 'OUTPUT' tab with the following text:

```
+-----+
| LOADING TABLE
+-----+
| TABLE (2 of 2): `menagerie`.`pet`
| Commands executed successfully: 3 of 3
| Warnings encountered: 0
| Table loaded successfully!
|   Total columns loaded: 5
|   Table loaded using 1 thread(s)
|   Elapsed time: 256.00 ms
+-----+
```

Below this, a 'LOAD SUMMARY' table is displayed:

SCHEMA NAME	TABLES LOADED	TABLES FAILED	COLUMNS LOADED	LOAD DURATION
`menagerie`	2	0	8	601.49 ms

At the bottom, a status bar indicates the task completion: [2024-01-16 11:54:06.312] [INFO] Task 'Load' completed. A notification bubble at the bottom right states: 'The data load to the HeatWave cluster operation has finished.'



MySQL Shell for Visual Studio Code

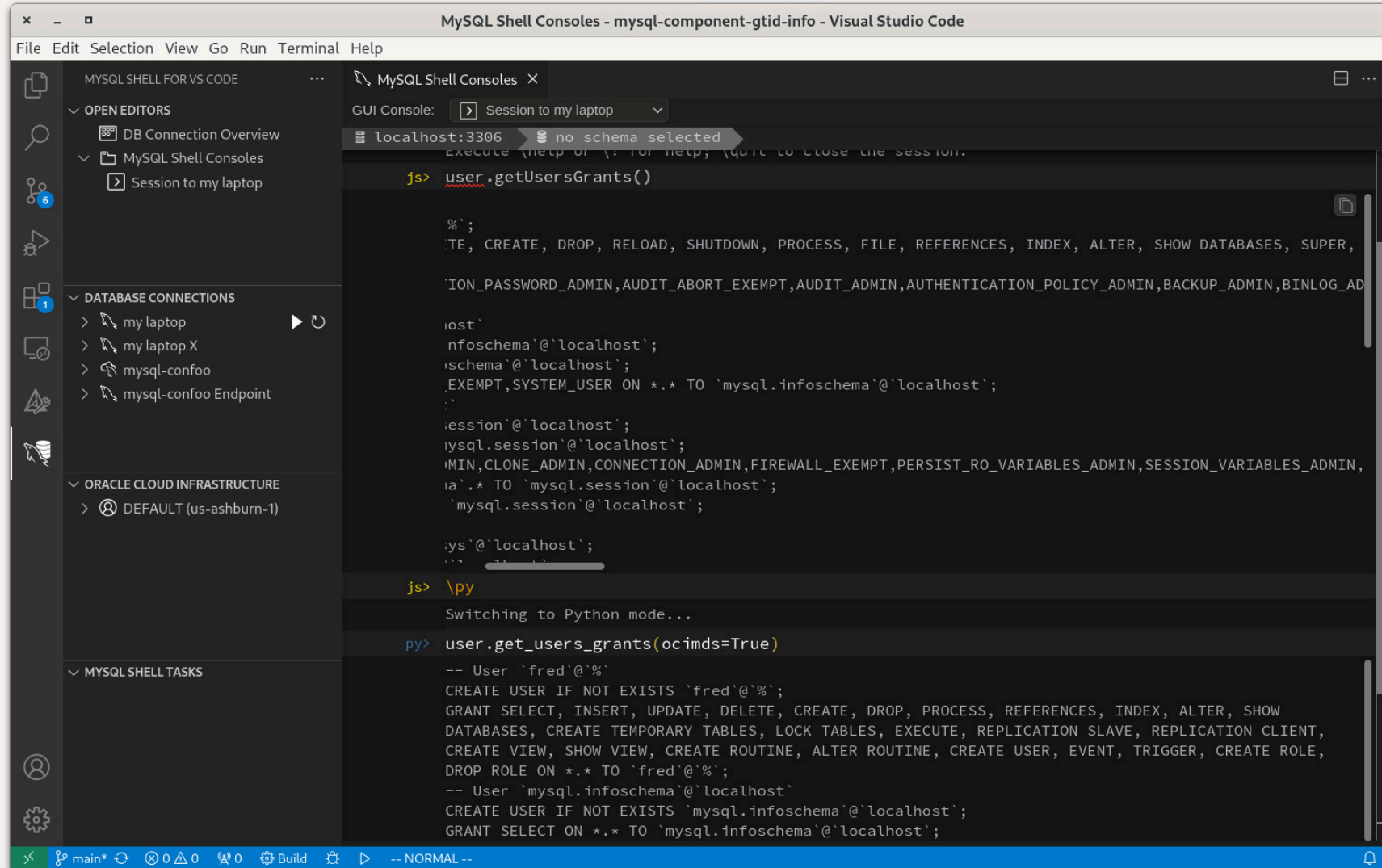
Extending the friendship ?

You can get more...

*You can still use and extend **MySQL Shell** for VS Code with your own plugins !*



MySQL Shell Extensions - example



```
MySQL Shell Consoles - mysql-component-gtid-info - Visual Studio Code
File Edit Selection View Go Run Terminal Help

MYSQL SHELL FOR VS CODE
OPEN EDITORS
  DB Connection Overview
  MySQL Shell Consoles
    Session to my laptop
DATABASE CONNECTIONS
  my laptop
  my laptop X
  mysql-confoo
  mysql-confoo Endpoint
ORACLE CLOUD INFRASTRUCTURE
  DEFAULT (us-ashburn-1)
MYSQL SHELL TASKS

GUI Console: Session to my laptop
localhost:3306 no schema selected
Execute (help or \? for help, \quit to close the session.)

js> user.getUsersGrants()

%';
TE, CREATE, DROP, RELOAD, SHUTDOWN, PROCESS, FILE, REFERENCES, INDEX, ALTER, SHOW DATABASES, SUPER,
TION_PASSWORD_ADMIN,AUDIT_ABORT_EXEMPT,AUDIT_ADMIN,AUTHENTICATION_POLICY_ADMIN,BACKUP_ADMIN,BINLOG_AD

host'
nfoschema`@`localhost`;
schema`@`localhost`;
EXEMPT,SYSTEM_USER ON *.* TO `mysql.infoschema`@`localhost`;
session`@`localhost`;
mysql.session`@`localhost`;
MIN,CLONE_ADMIN,CONNECTION_ADMIN,FIREWALL_EXEMPT,PERSIST_RO_VARIABLES_ADMIN,SESSION_VARIABLES_ADMIN,
ia`.* TO `mysql.session`@`localhost`;
`mysql.session`@`localhost`;

ys`@`localhost`;

js> \py
Switching to Python mode...

py> user.get_users_grants(ocimds=True)
-- User `fred`@`%`
CREATE USER IF NOT EXISTS `fred`@`%`;
GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, PROCESS, REFERENCES, INDEX, ALTER, SHOW
DATABASES, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, REPLICATION SLAVE, REPLICATION CLIENT,
CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, CREATE USER, EVENT, TRIGGER, CREATE ROLE,
DROP ROLE ON *.* TO `fred`@`%`;
-- User `mysql.infoschema`@`localhost`
CREATE USER IF NOT EXISTS `mysql.infoschema`@`localhost`;
GRANT SELECT ON *.* TO `mysql.infoschema`@`localhost`;
```



Share your ❤️ to **MySQL**

#mysql #MySQLCommunity



Join our slack channel!

bit.ly/mysql-slack

Questions ?