# Developing apps quickly with Azure Database for MySQL

## Overview

Azure Database for MySQL is a fully managed service in the cloud. Azure Database for MySQL offers scalable, highly available functionality for your apps. Just like any MySQL Database on a Windows/Linux/MAC OS you can use MySQL Workbench, PHPMyadmin or any third-party tools for connection.

We have built a sakila database on Azure Database for MySQL as well as on a Local Windows VM Instance for the user. We also created all necessary tables with Primary, Foreign Keys, Constraints, Indexes and imported data into the respective databases.

MySQL is installed on a Windows VM including Workbench (mysql-installer-web-community-5.7.18.1).

### Scenario Overview

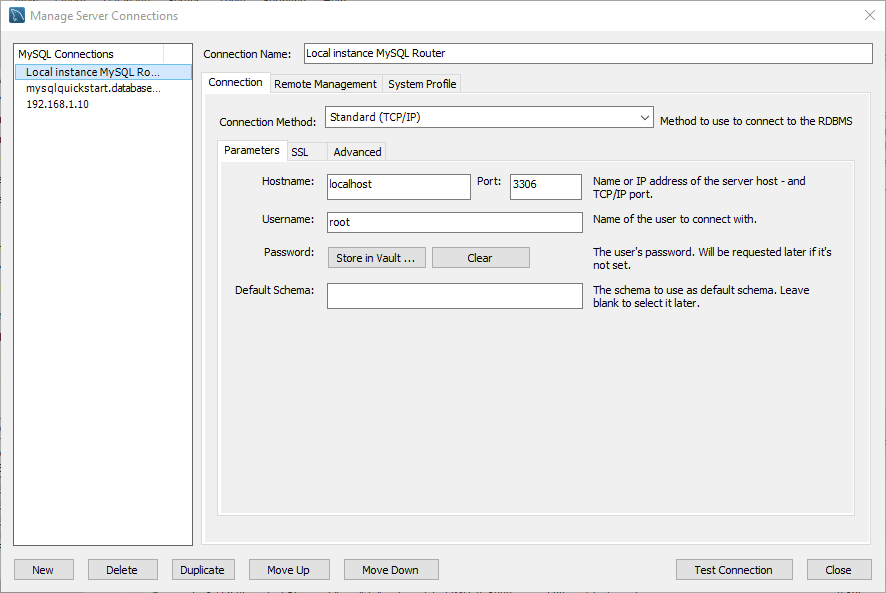
This hands-on lab will step you through the following features:

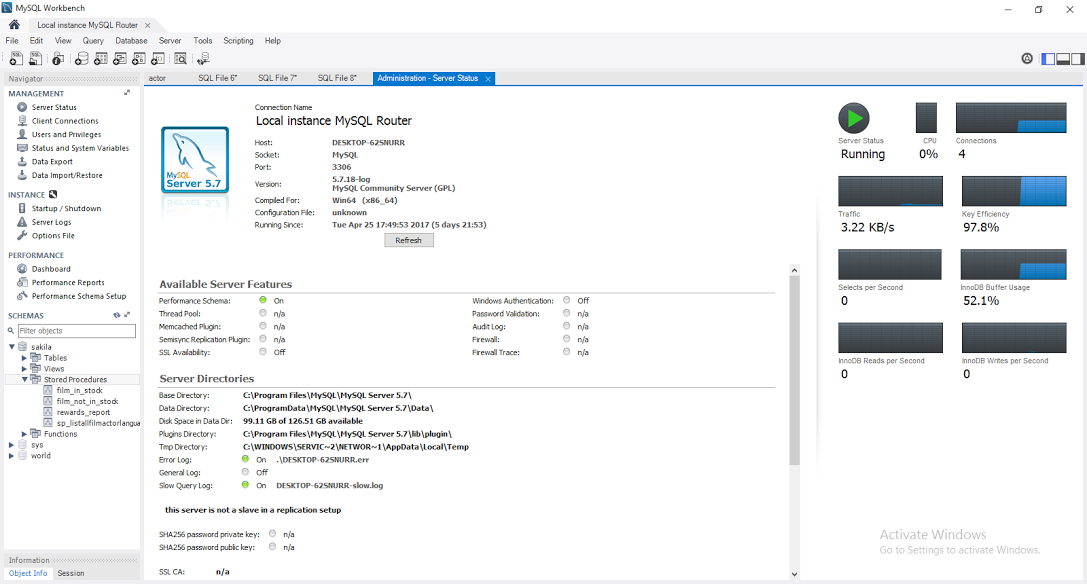
1. Running a Stored Procedure on both (Local MySQL Instance and Azure Database for MySQL) of above environments.
2. Retrieving the data from a View on both (Local MySQL Instance and Azure Database for MySQL) of above environments.

### About the code challenge

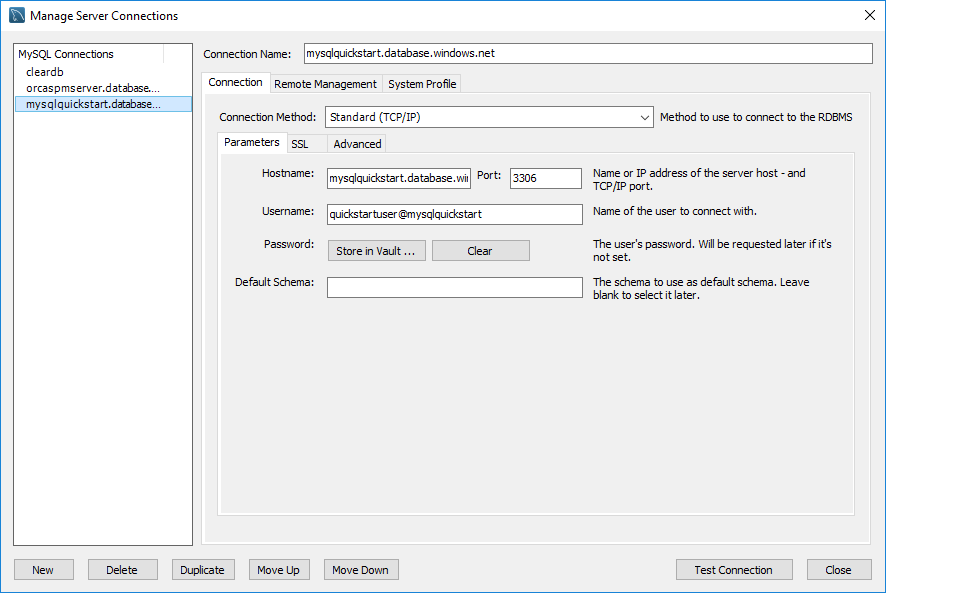
This lab uses a simple MySQL Application as a test application. This simple application can be modified to test the various features of **Azure Database for MySQL.**

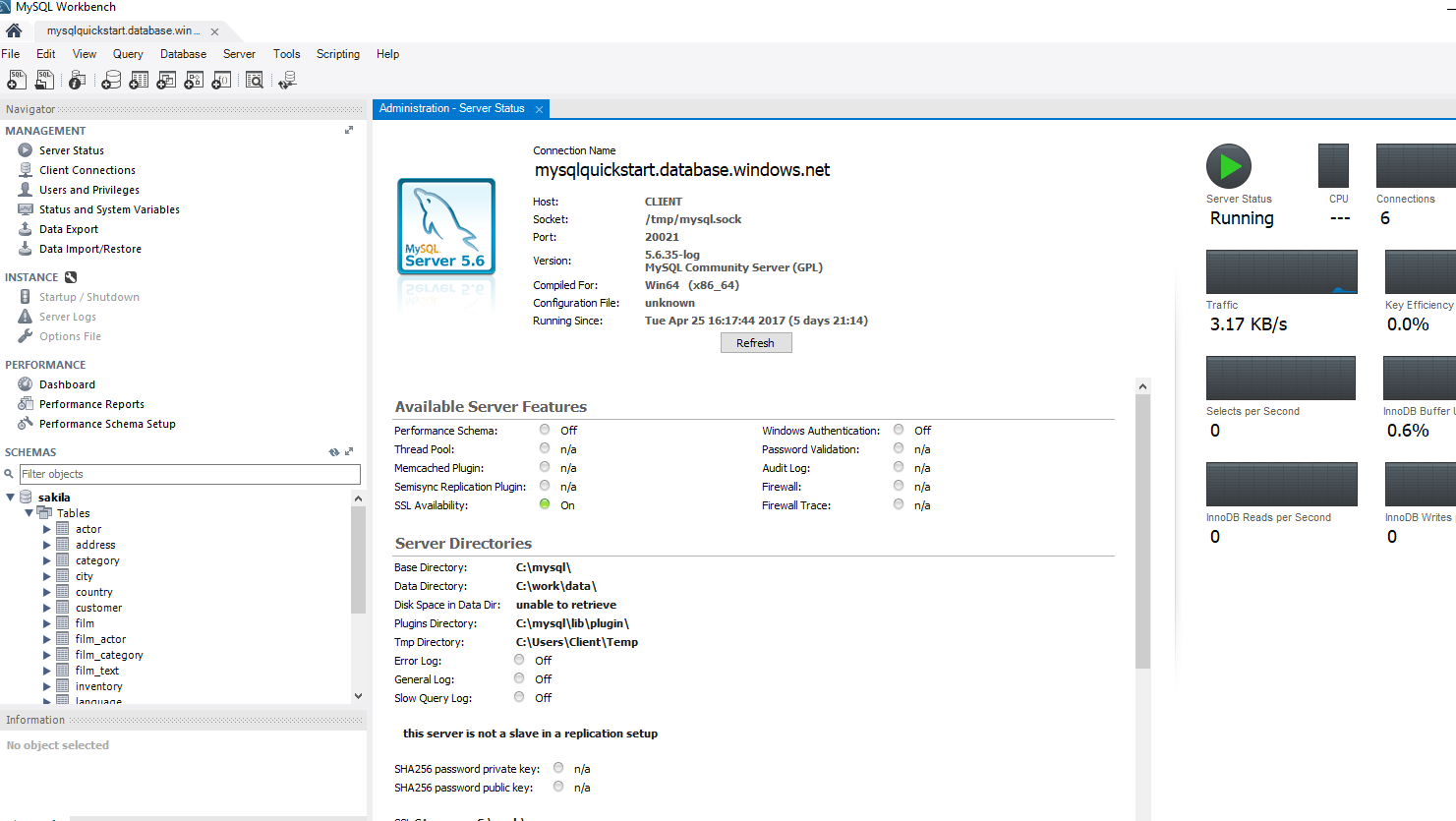
**Below is the screenshot of a Windows VM with a Local MySQL Database using MySQL Workbench:**





**Below is the screenshot of an Azure Database for MySQL using MySQL Workbench:**





**Connection string details**

**Server:** **mysqlquickstart.database.windows.net**

**User:** quickstartuser@mysqlquickstart

**Password**: Quick$tart123

**Resource Group:** Build17MySQLQuickStart

**Location:** West US

**Port:**3306

**Note:** The Azure Database for MySQL that we will be querying during this lab was created via the Azure Portal. For more information on the Azure Portal refer to the **Appendix** at the bottom of the page.

Scenario 1: Running a Stored Procedure on Local MySQL Instance and Azure Database for MySQL:

In this scenario, we will explain how to call or execute **a stored procedure**.

* Open MySQL Workbench with a connection to Local MySQL Instance as well as Azure Database for MySQL as per screenshots above.

Call the stored procedure sakila.sp\_listallfilmactorlanguage by entering below in MySQL Workbench -> New Query -> call sakila.sp\_listallfilmactorlanguage();

and then execute on both above environments.

Below is the create statement of the stored procedure:

DELIMITER //

CREATE PROCEDURE sakila.`sp\_listallfilmactorlanguage` ()

LANGUAGE SQL

DETERMINISTIC

SQL SECURITY DEFINER

COMMENT 'A procedure'

BEGIN

select FA.film\_id, FA.actor\_id, A.first\_name,A.last\_name, F.title, F.description,

F.release\_year,F.language\_id, L.name, F.rating

from

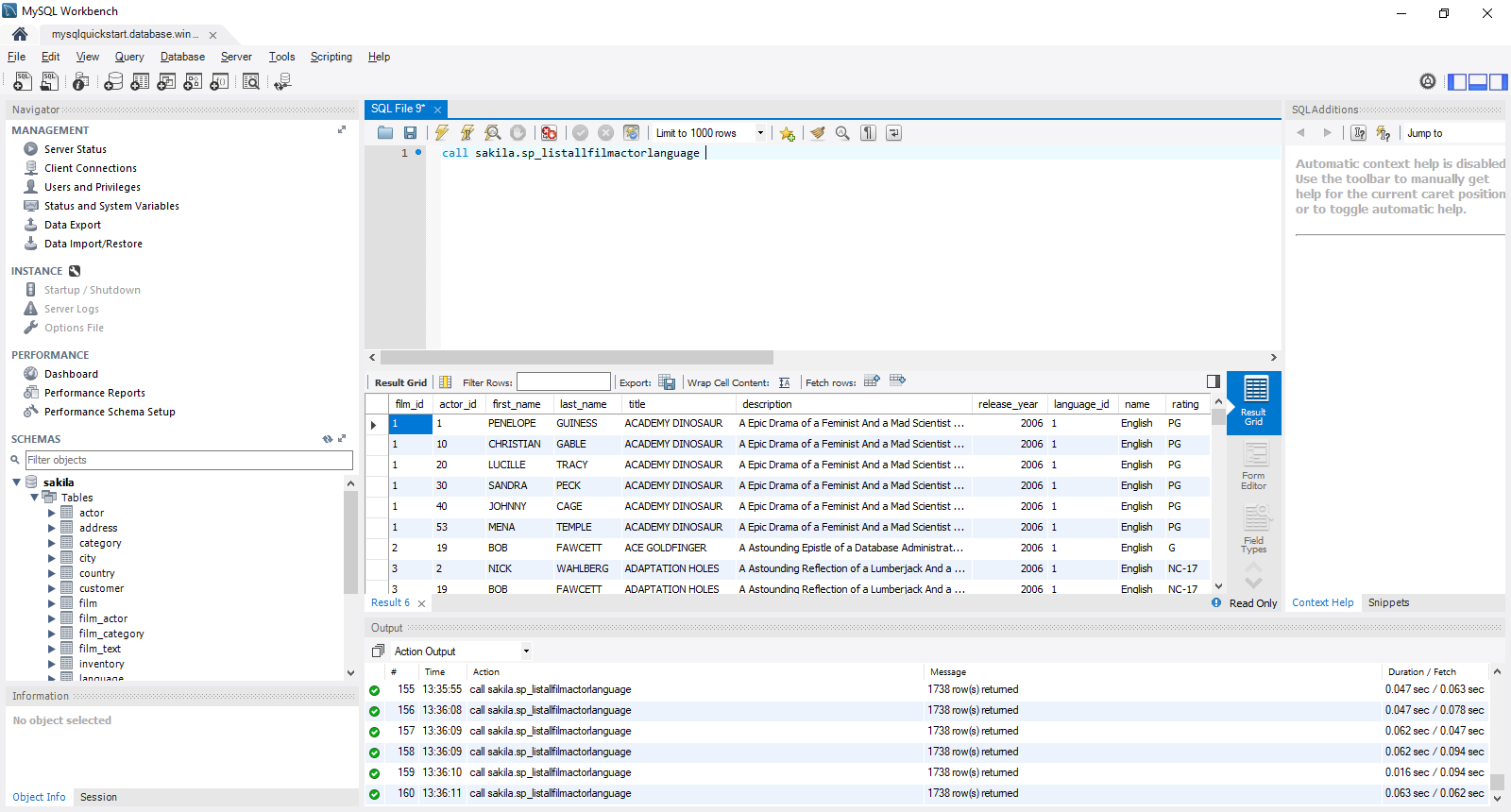
sakila.film\_actor FA

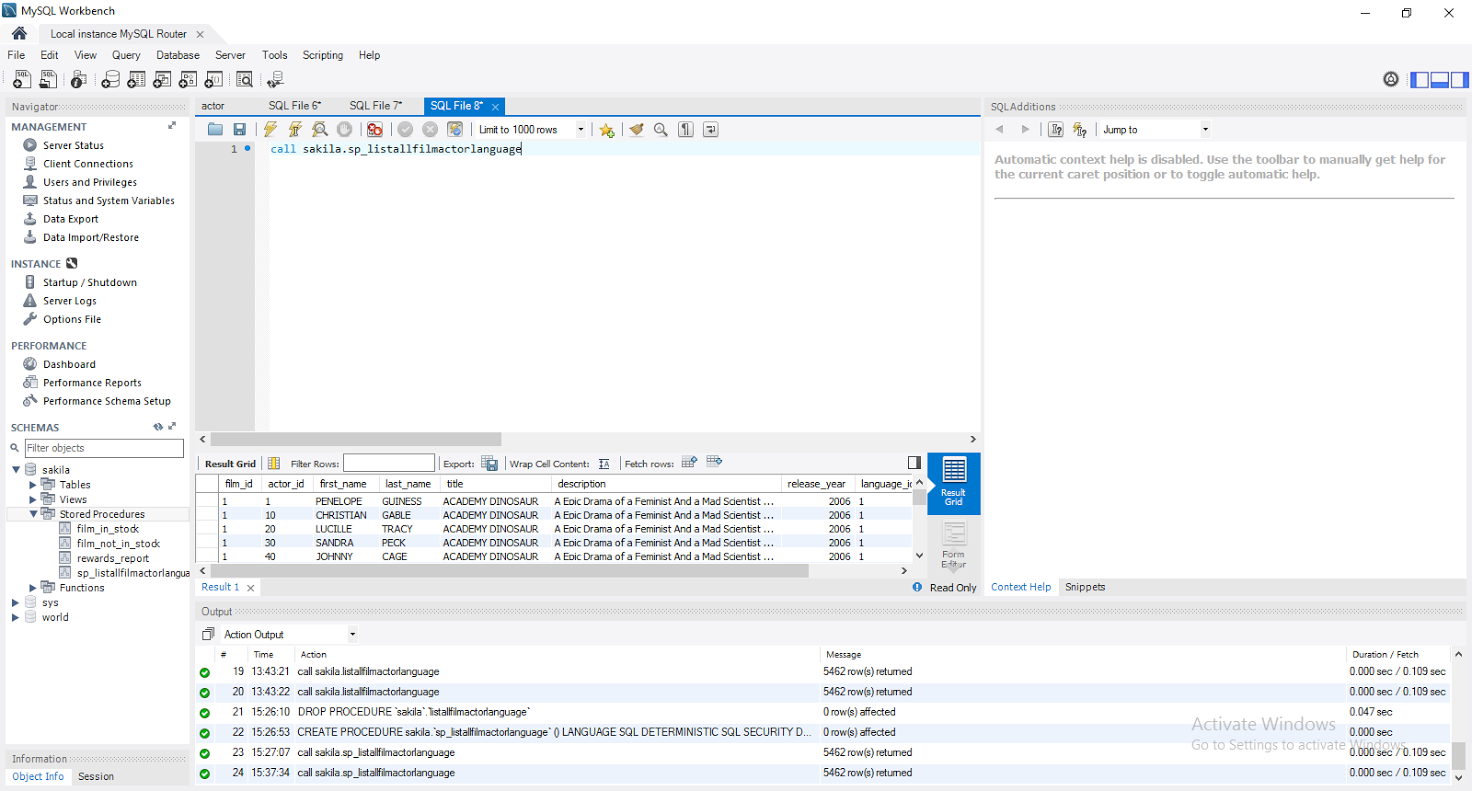
left outer join sakila.actor A on FA.actor\_id=A.actor\_id

left outer join sakila.film F on FA.film\_id=F.film\_id

left outer join sakila.language L on L.language\_id=F.language\_id;

END//





### 

## Scenario 2: Retrieving the data from a View on Local MySQL Instance and Azure Database for MySQL:

In this scenario, we will explain how to retrieve the data from **a View**.

Use above two connections from MySQL Workbench to Local MySQL Instance and Azure Database for MySQL.

* Select the data from the view **sakila.v\_listallcustomerinfo** by entering below in MySQL Workbench -> **New Query -> SELECT \* FROM sakila.v\_listallcustomerinfo;** and then execute on both above environments.

Below is the create statement of the View:

USE sakila;

CREATE view v\_listallcustomerinfo as

select C.customer\_id, C.first\_name, C.last\_name, C.store\_id, C.email, C.address\_id,

A.address,A.address2,A.city\_id, CT.city,

A.postal\_code, CT.country\_id,CN.country, A.phone

from

sakila.customer C

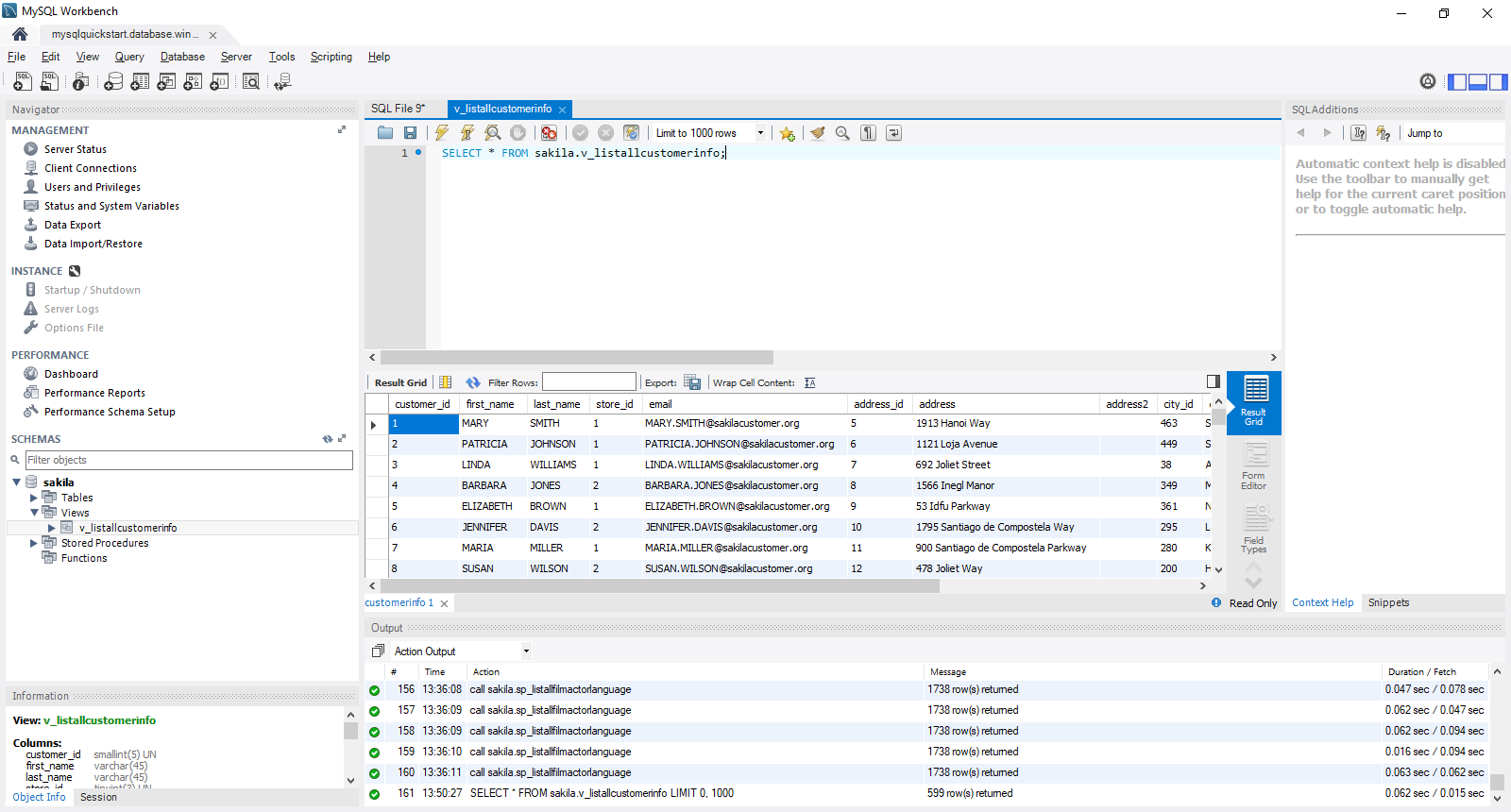
left outer join sakila.store S on C.store\_id=S.store\_id

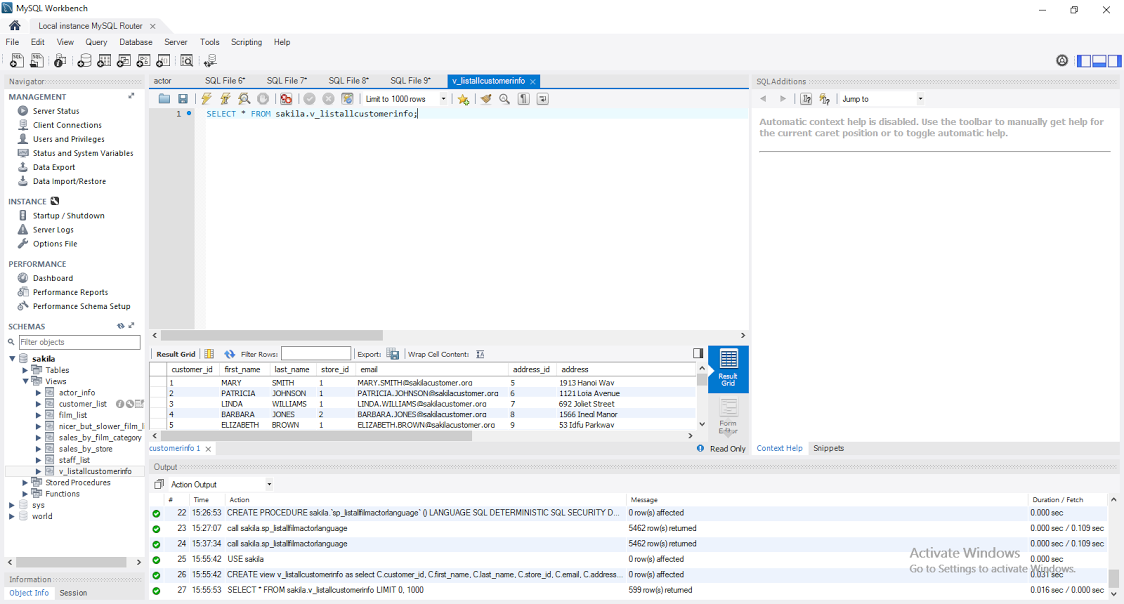
left outer join sakila.address A on C.address\_id=A.address\_id

left outer join sakila.city CT on A.city\_id=CT.city\_id

left outer join sakila.country CN on CT.country\_id=CN.country\_id;

**It will display the result as below:**

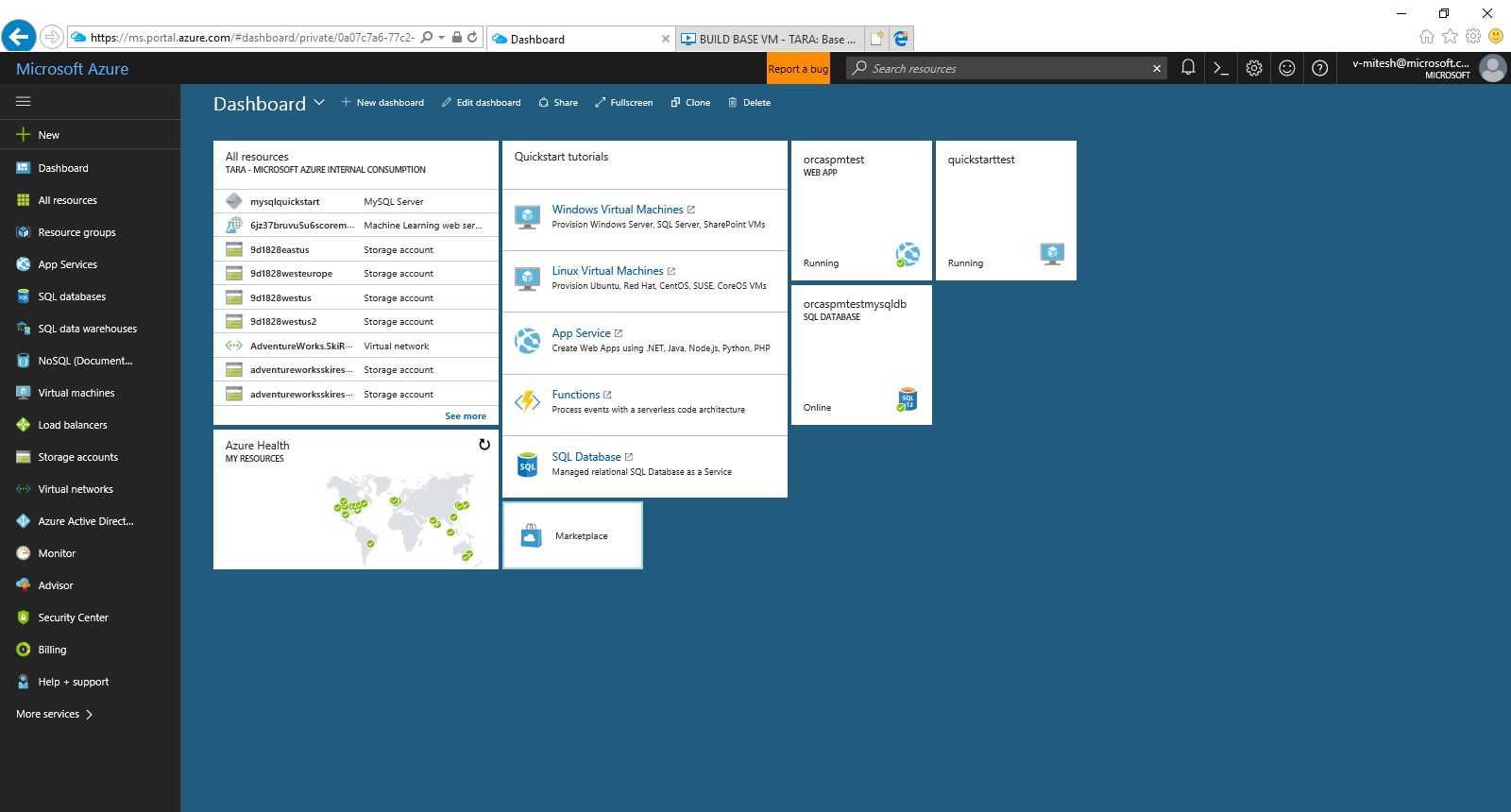




## Appendix

The Azure Portal was used to create the Azure Search server. The Azure Portal can be found at <https://portal.azure.com/>.

**Dashboard:**



**Connection String:**

