



Infraestrutura I

Objetivos

Revise os termos vistos nos módulos do PowerShell e coloque-os em prática, baixando um módulo e executando comandos.

Pré-requisitos

MySQL é um banco de dados comum entre muitas organizações. Como tal, ao criar um script de automação que precisa consultar dados de algum lugar, você pode precisar consultar um banco de dados MySQL.

Para consultar um banco de dados MySQL, presumimos que você já tenha um servidor MySQL do Windows e um banco de dados com pelo menos uma tabela que você gostaria de recuperar. Também vamos supor que você não está gerenciando este servidor a partir do próprio servidor, mas de um software cliente, como o MySQL Workbench, que permite que você se conecte remotamente à sua instância MySQL.

Infelizmente, o PowerShell não vem equipado com suporte ao MySQL. Temos que baixar e instalar o software chamado Connector/NET. Este é um driver ADO ao qual o PowerShell pode se conectar para gerenciar bancos de dados MySQL. Se estiver trabalhando em uma máquina que já possui o software MySQL Workbench completo, não precisará instalá-lo.

Depois que o driver ADO for instalado, você precisará baixar algum código do PowerShell que já foi criado para fazer isso. Será usado um módulo do PowerShell chamado MySQL. Este módulo consiste em vários comandos que podem ser usados para gerenciar bancos de dados MySQL.

Construindo o ambiente

Opção 1:

Você pode fazer o exercício em seu computador, como requisito necessário para isso, é ter o MySQL instalado.

Opção 2:

Caso não tenha MySQL ou MariaDB instalados, você pode executar o database engine dentro de um container, para isso é necessário ter o Docker instalado.

Exercício

No exercício a seguir, trabalharemos com os módulos do PowerShell.

Se você não tiver MySQL ou MariaDB instalado, você pode executar o mecanismo de banco de dados dentro de um container:

```
docker run -d --rm -e MARIADB_ROOT_PASSWORD=Password123 --name testdb -p 3306:3306 mariadb:latest
```

E então execute o seguinte comando para entrar no contexto do container:

```
docker exec -it testdb bash
```

Uma vez dentro do container, execute o seguinte comando para entrar no MySQL:

```
mysql -p
```

O comando anterior nos pedirá para entrar com a senha do usuário root. Caso você tenha decidido executar o banco de dados com um container Docker, a senha é aquela que colocamos na variável de ambiente MARIADB_ROOT_PASSWORD. Ou seja, "Password123" sem aspas.

Então você deve criar um banco de dados em MySQL (ou MariaDB) executando estes comandos:

```
CREATE DATABASE test;
```

```
Use test;
```

```
CREATE TABLE test (field1 INT);
```

```
ALTER TABLE
```

```
test
```

```
CONVERT TO CHARACTER SET utf8mb4
```

```
COLLATE utf8mb4_unicode_ci;
```

```
INSERT INTO test (field1) VALUES (1);
```

No PowerShell vamos a buscar um módulo que se chama "invokequery":

```
find-module -name invokequery
```

Se não encontrá, terá que instalar:

```
install-module -name invokequery
```

Pode ser que PowerShell te pergunte se quer confiar no repositório remoto, responda que sim apertando o "Y".

Em seguida, usaremos o módulo recém-instalado para executar uma consulta no banco de dados que criamos em uma instância anterior.

Para isso, primeiro armazenaremos as credenciais do MySQL em nosso objeto PSCredential:

```
$creds = Get-Credential
```

Nesse caso, o PowerShell solicitará que você insira as credenciais em um prompt. Se você usou o Docker para iniciar o BD, o usuário é "root" (sem aspas) e a senha é aquela que colocamos na variável de ambiente MARIADB_ROOT_PASSWORD ("Password123" sem aspas).

Em seguida, vamos escrever a consulta SQL que vamos executar no banco de dados e salvá-la em uma variável:

```
$query = New-SqlQuery -sql "select * from test"
```

Por fim, vamos executar o comando no DB:

```
Invoke-MySqlQuery -SqlQuery $query -Credential $creds -Server localhost -Database test
```