Original task was to setup databases in the created earlier domain

Table of Contents

Portfolio 2 2
- **Observations** 2
- **Screenshots** 3
- **Reflections** 9
- **Changes** 9

Portfolio 2

Observations

Topology and VMs descriptions can be seen here (Table 1, Figure 1).

Microsoft SQL server 14.0.1000.169 as well as SQL Server Management Studio 18.1 were installed on the third Windows server machine. After that, simple database was created, two tables were added and filled with data samples (Figure 2).

Database was then backed up manually (Figure 3, Figure 4). Several options can be configured: backup type (full, differential or transaction log), backup path (local path or URL), encryption, schedule, etc. In this case, whole database was backed up to the local hard drive.

Then, tables were deleted (Figure 5). Query returns error, that there is no such a table.

Database was restored from the back up (Figure 6). All the data is in place.

MySQL community server 8.0 was installed on the second Linux server. After that, simple database was created (Figure 8).

Database was backed up manually using mysqldump utility (Figure 9). Mysqldump utility has a lot of different options. One can back up different databases to different backup files, remote server can be specified, back up files can be compressed and/or encrypted, etc.

Then, database was dropped completely (Figure 10).

Database was then restored from the back up (Figure 11). All the data can be seen again (Figure 12).

Table 1. VMs description

VM Name	OS	Memory	Disk Space	Admin account	VM function	IP address
WINSRV1-SB-8569394	Windows server 2016	4 GB	40 GB	Administrator Secret55	Domain Controller, DNS and DHCP server	10.174.68.10/24
WINSRV2-SB-8569394	Windows server 2016	4 GB	40 GB	Administrator Secret55	Domain Controller, DNS and DHCP server	10.174.68.11/24
LINSRV1-SB-8569394	CentOS 7	1 GB	8 GB	Administrator Root Secret55	Web-Server	10.174.68.12/24
LINSRV2-SB9394	CentOS 7	1 GB	8 GB	Administrator Root Secret55	MySQL Server	10.174.68.13/24
WINWS-SB-8569394	Windows 10	2 GB	32 GB	Administrator Secret55	Workstation	10.174.68.40/24
LINWS-SB-8569394	CentOS 7	1 GB	4 GB	Administrator Root Secret55	Workstation	10.174.68.41/24
BACKSRV-SB-8569394	Windows Server 2016	4 GB	40 GB	Administrator Secret55	MS SQL Server	10.174.68.30/24

Screenshots

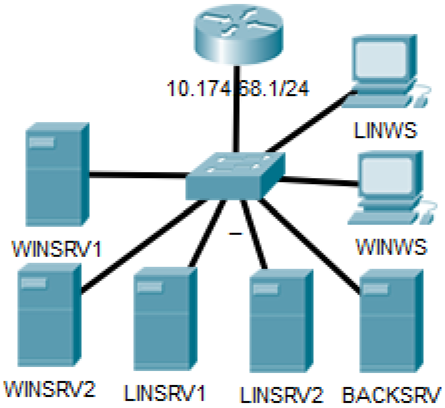
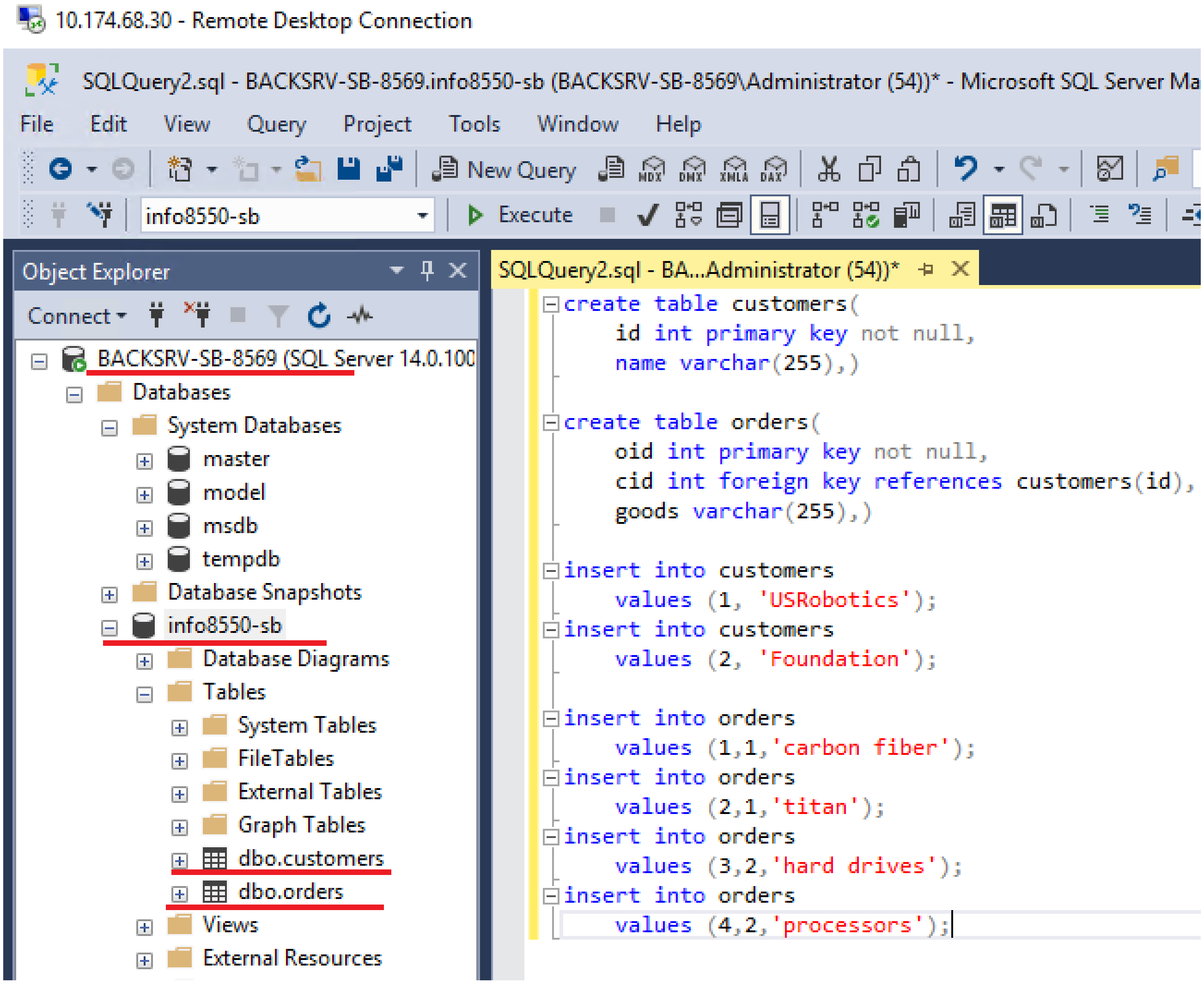
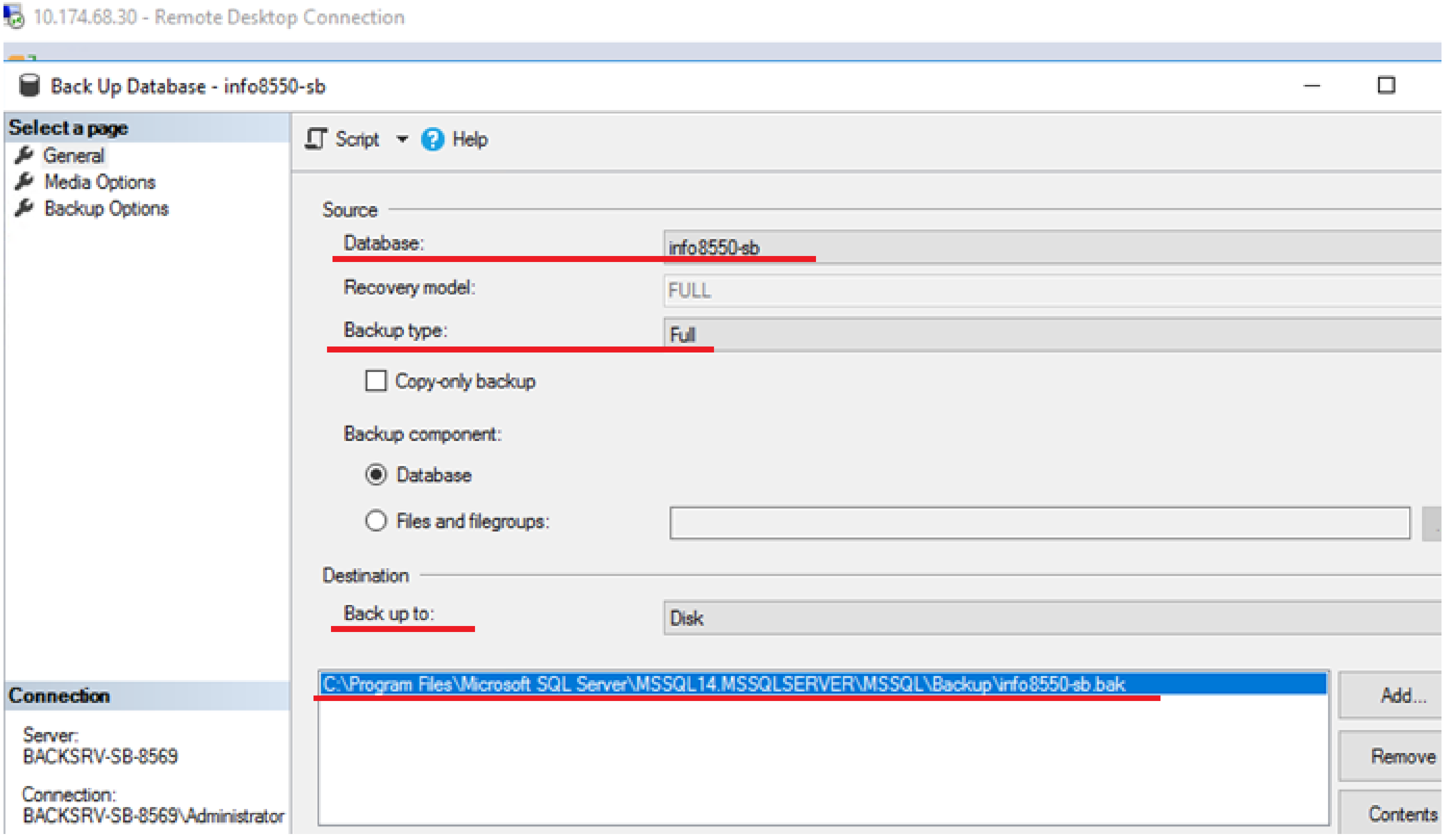


Figure 1. Topology



```
create table customers(
    id int primary key not null,
    name varchar(255),)

create table orders(
    oid int primary key not null,
    cid int foreign key references customers(id),
    goods varchar(255),)

insert into customers
    values (1, 'USRobotics');
insert into customers
    values (2, 'Foundation');

insert into orders
    values (1,1,'carbon fiber');
insert into orders
    values (2,1,'titan');
insert into orders
    values (3,2,'hard drives');
insert into orders
    values (4,2,'processors');
```

Figure 2. MS SQL database was created



Source

Database: info8550-sb

Recovery model: FULL

Backup type: Full

□ Copy-only backup

Backup component:

● Database

○ Files and filegroups:

Destination

Back up to: Disk

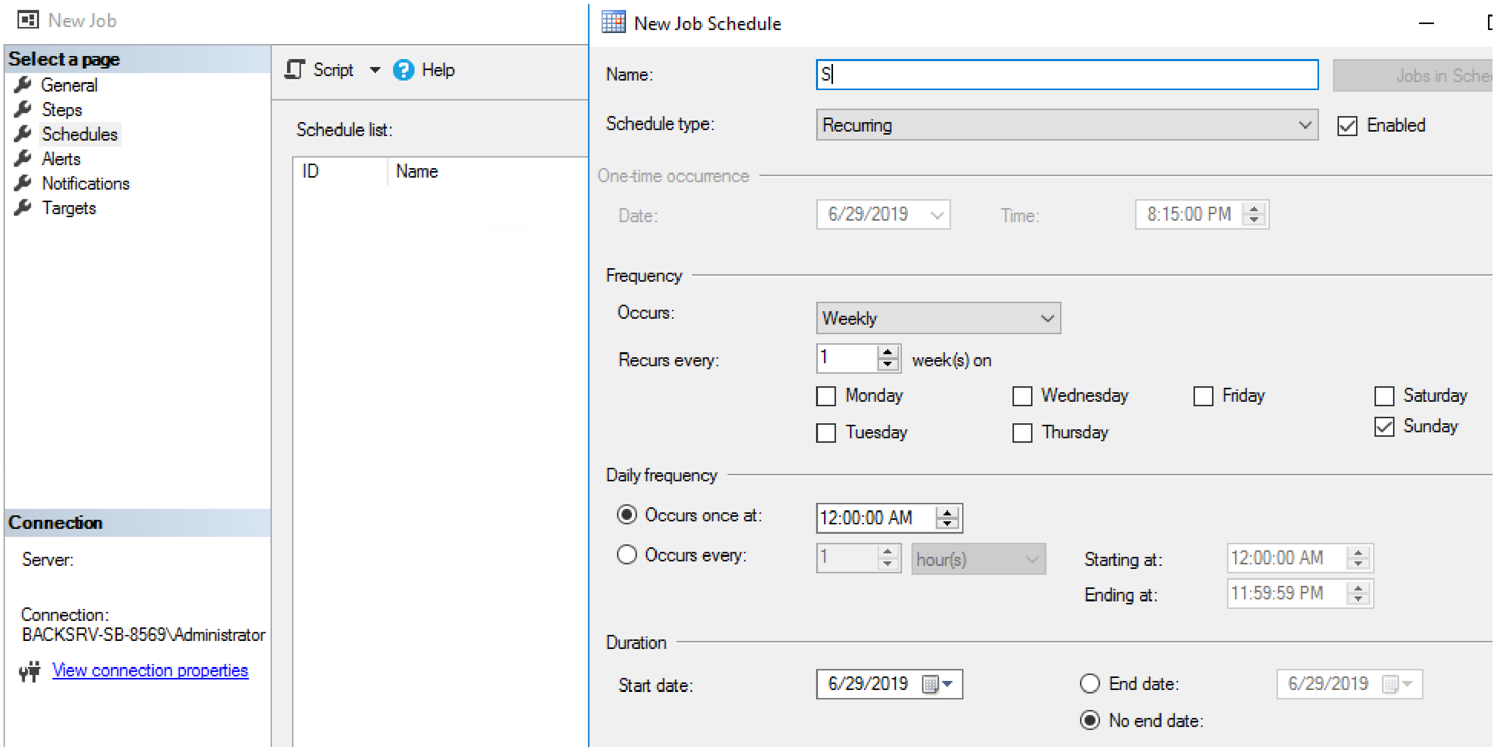
C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\MSSQL\Backup\info8550-sb.bak

Connection

Server:
BACKSRV-SB-8569

Connection:
BACKSRV-SB-8569\Administrator

Figure 3. Backing up MS SQL database



Schedule list:

ID	Name

Name: S

Schedule type: Recurring ☑ Enabled

One-time occurrence

Date: 6/29/2019 Time: 8:15:00 PM

Frequency

Occurs: Weekly

Recurs every: 1 week(s) on

□ Monday □ Wednesday □ Friday □ Saturday

□ Tuesday □ Thursday ☑ Sunday

Daily frequency

● Occurs once at: 12:00:00 AM

○ Occurs every: 1 hour(s) Starting at: 12:00:00 AM Ending at: 11:59:59 PM

Duration

Start date: 6/29/2019 ○ End date: 6/29/2019

● No end date:

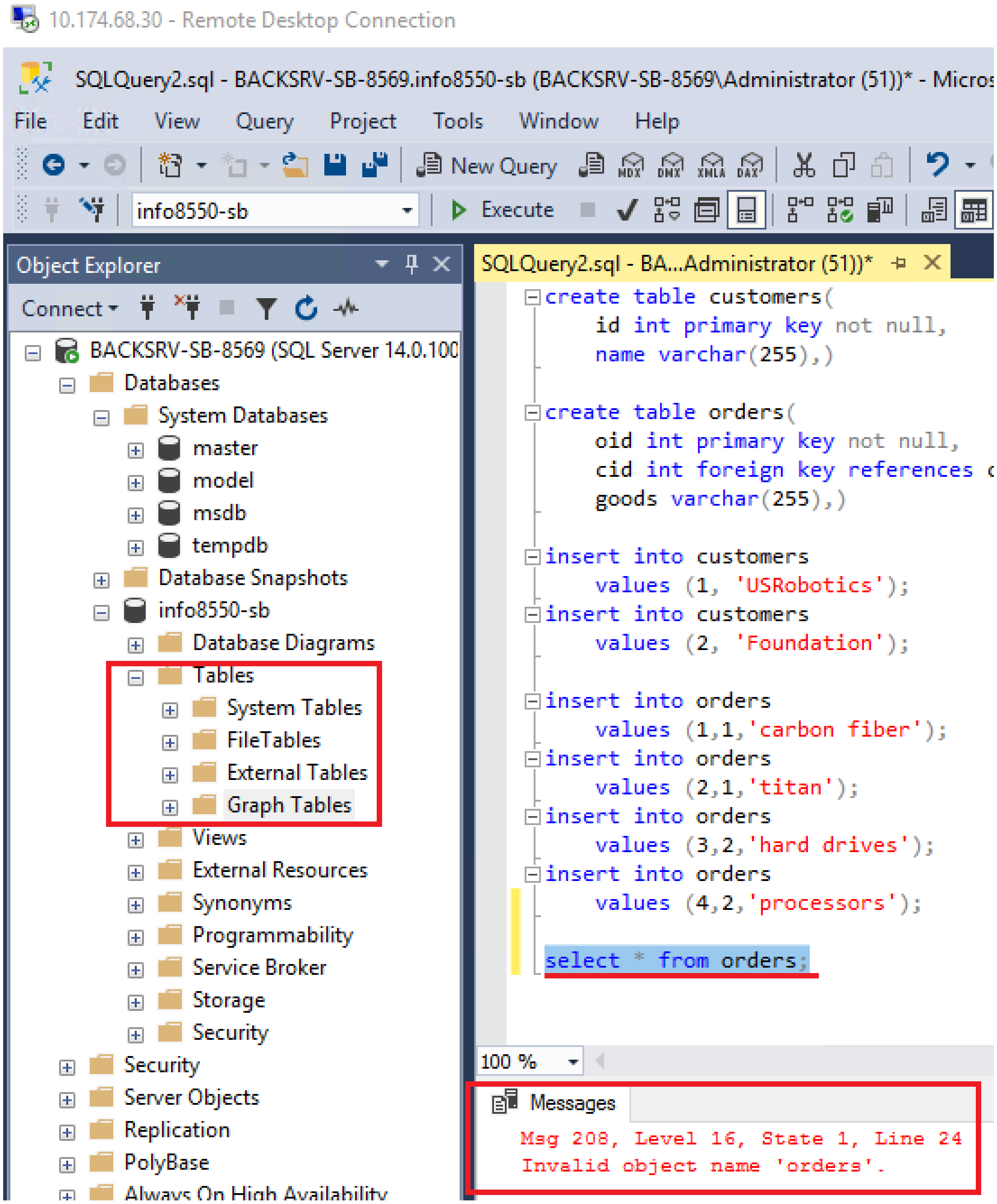
Connection

Server:

Connection:
BACKSRV-SB-8569\Administrator

View connection properties

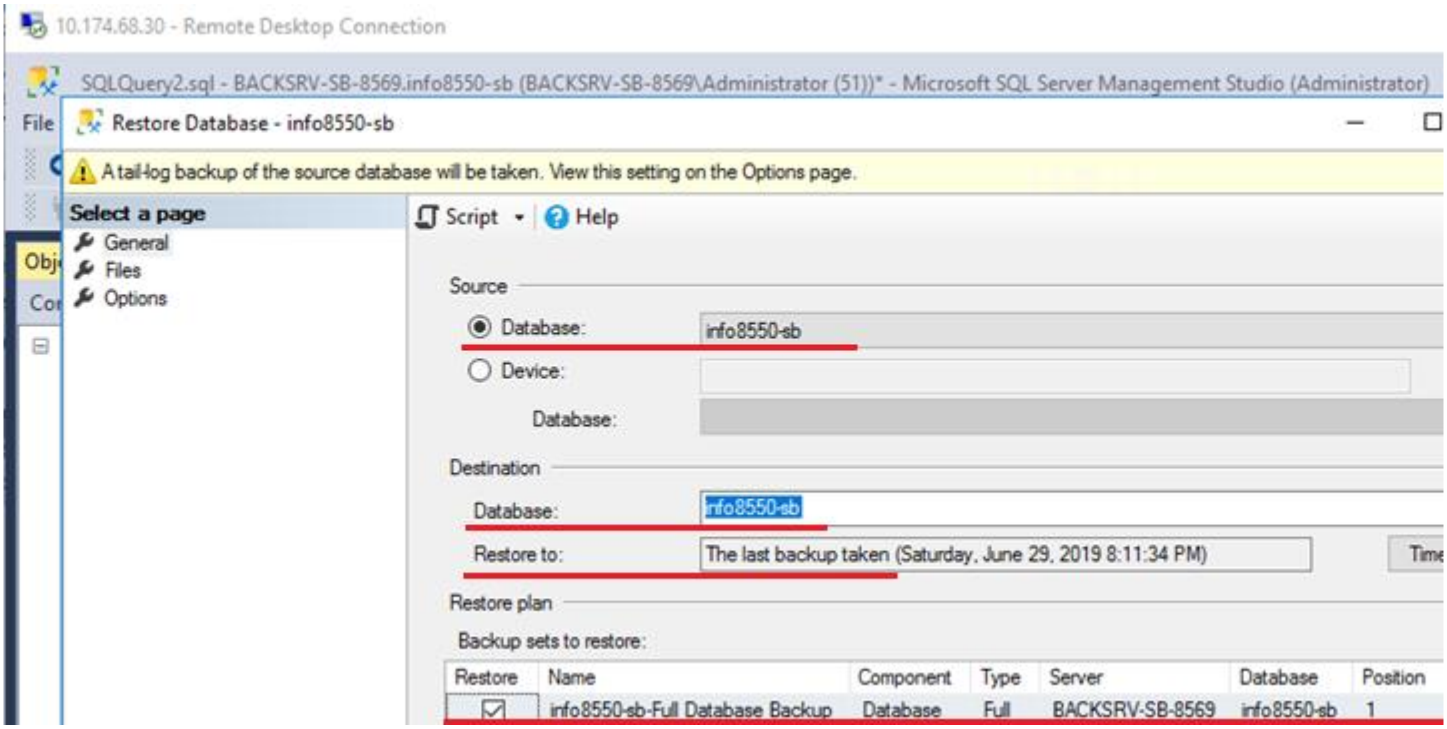
Figure 4. Scheduling backing up process of MS SQL database



```
create table customers(
    id int primary key not null,
    name varchar(255),)

create table orders(
    oid int primary key not null,
    cid int foreign key references c
    goods varchar(255),)

insert into customers
    values (1, 'USRobotics');
insert into customers
    values (2, 'Foundation');

insert into orders
    values (1,1,'carbon fiber');
insert into orders
    values (2,1,'titan');
insert into orders
    values (3,2,'hard drives');
insert into orders
    values (4,2,'processors');

select * from orders;
```

```
Msg 208, Level 16, State 1, Line 24
Invalid object name 'orders'.
```

Figure 5. Tables were deleted



10.174.68.30 - Remote Desktop Connection

SQLQuery2.sql - BACKSRV-SB-8569.info8550-sb (BACKSRV-SB-8569\Administrator (51))* - Microsoft SQL Server Management Studio (Administrator)

Restore Database - info8550-sb

A tail-log backup of the source database will be taken. View this setting on the Options page.

Source

● Database: info8550-sb

○ Device:

Database:

Destination

Database: info8550-sb

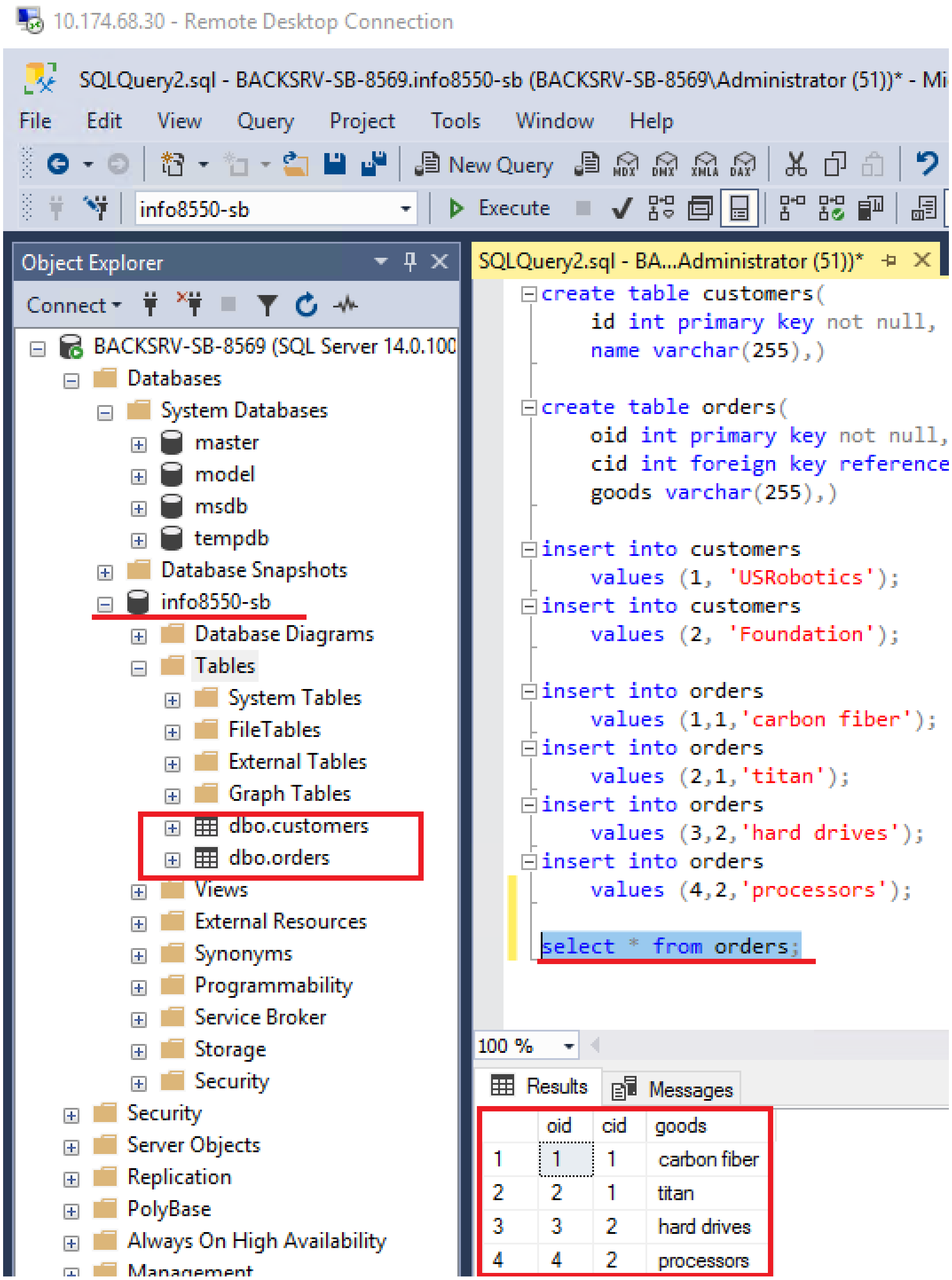
Restore to: The last backup taken (Saturday, June 29, 2019 8:11:34 PM)

Restore plan

Backup sets to restore:

Restore	Name	Component	Type	Server	Database	Position
☑	info8550-sb-Full Database Backup	Database	Full	BACKSRV-SB-8569	info8550-sb	1

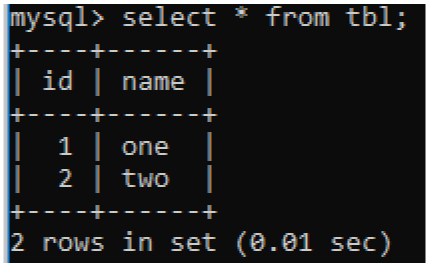
Figure 6. Restoring database



```
create table customers(
    id int primary key not null,
    name varchar(255),)

create table orders(
    oid int primary key not null,
    cid int foreign key reference
    goods varchar(255),)

insert into customers
    values (1, 'USRobotics');
insert into customers
    values (2, 'Foundation');

insert into orders
    values (1,1,'carbon fiber');
insert into orders
    values (2,1,'titan');
insert into orders
    values (3,2,'hard drives');
insert into orders
    values (4,2,'processors');

select * from orders;
```

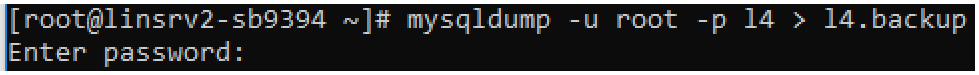
	oid	cid	goods
1	1	1	carbon fiber
2	2	1	titan
3	3	2	hard drives
4	4	2	processors

Figure 7. Database was restored



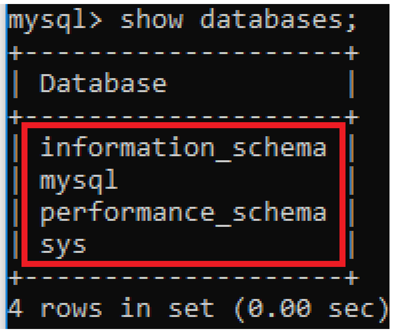
```
mysql> select * from tbl;
+----+------+
| id | name |
+----+------+
|  1 | one  |
|  2 | two  |
+----+------+
2 rows in set (0.01 sec)
```

Figure 8. MySQL database was created



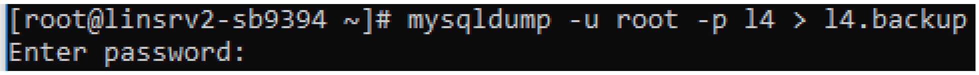
```
[root@linsrv2-sb9394 ~]# mysqldump -u root -p l4 > l4.backup
Enter password:
```

Figure 9. MySQL database was backed up



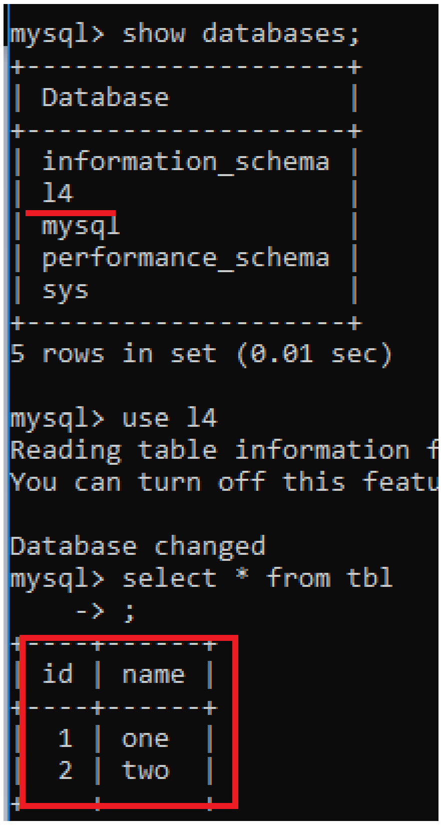
```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.00 sec)
```

Figure 10. Database was deleted



```
[root@linsrv2-sb9394 ~]# mysqldump -u root -p l4 > l4.backup
Enter password:
```

Figure 11. Restoring database



```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| l4                 |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.01 sec)

mysql> use l4
Reading table information f
You can turn off this featu

Database changed
mysql> select * from tbl
    -> ;
+----+------+
| id | name |
+----+------+
|  1 | one  |
|  2 | two  |
+
```

Figure 12. Restored MySQL database

Reflections

Although, backup was created, there is no big sense in them in the real environment. They can only prevent accidental data changes. For instance, if hard drive failure occurs, database will not be recovered. Database should be backed up to several different places: to another hard drive, to another host, maybe to a cloud. This will increase redundancy significantly. Then, this process must be scheduled. It can be done in the MS SQL Management Studio (Figure 4) on Windows machines and using cron utility for MySQL database on Linux machine. E.g. every day at night, when databases are hardly loaded. Furthermore, several processes can be configured, i.e. one for full backup once a week and another one for quicker differential everyday backup.

One issue with MySQL server was faced during completing this portfolio. MySQL server did not want to install on Linux VM with thin provisioned disk, although maximum capacity of the disk was big enough. 'Ignorespacecheck' flag did not help. So, it was decided to reimage VM with thick provisioned disk.

Changes

There were minor changes in the Observations section, the table with descriptions was added.

Topology screenshot was added to the Screenshots section. On several other screenshots a few details were highlighted.

Reflections sections now contains more details. A problem description and its resolution were also added.