Screenshots should show the key points in steps of lab. It will help you to response, and us – to ask correct questions. Do all possible steps manually too, not only TSQL, it is not critical to write SQL, it is critical you could recognize the actions and steps on interview, and prove your answer.

Part 1. Preparing Lab infrastructure

1. Attach “AdventureWorks” to one of the SQL instances (if was not done)
2. Make Full backup of DB with simple recovery model
3. Restore Database to new database, named it as “your student name”\_db (vshulhach\_db for example)

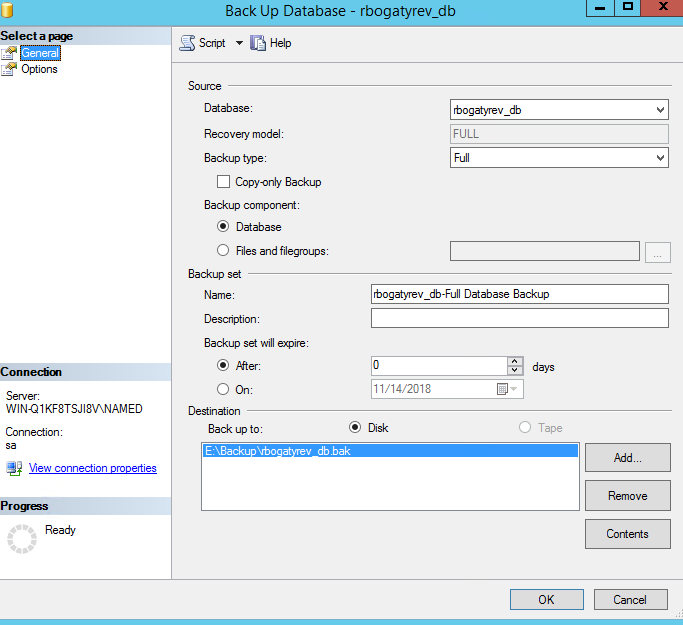
Can be done manually, no script.

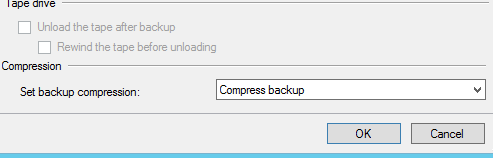
Part 2. Working with DB backups and recovering.

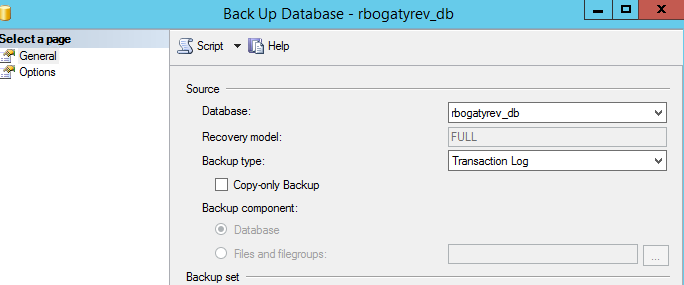
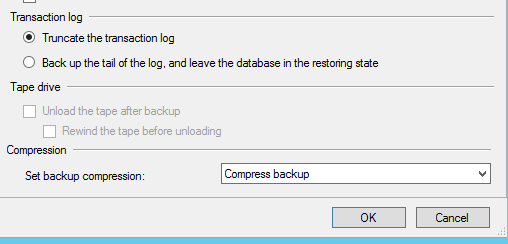
As you have full backup of DB, you can break it in anyway, and restore to the custom point without any problem.

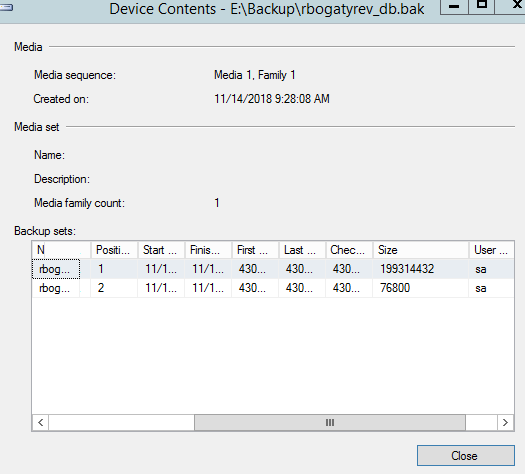
All next operations should be performed on “Student\_name” db.

1. Set recovery model to FULL. Make backup of DB. Make backup of transaction log. Perform everything using your custom dataset. Backups should be compressed. Compare sizes of backups. Do not forget about screenshots.

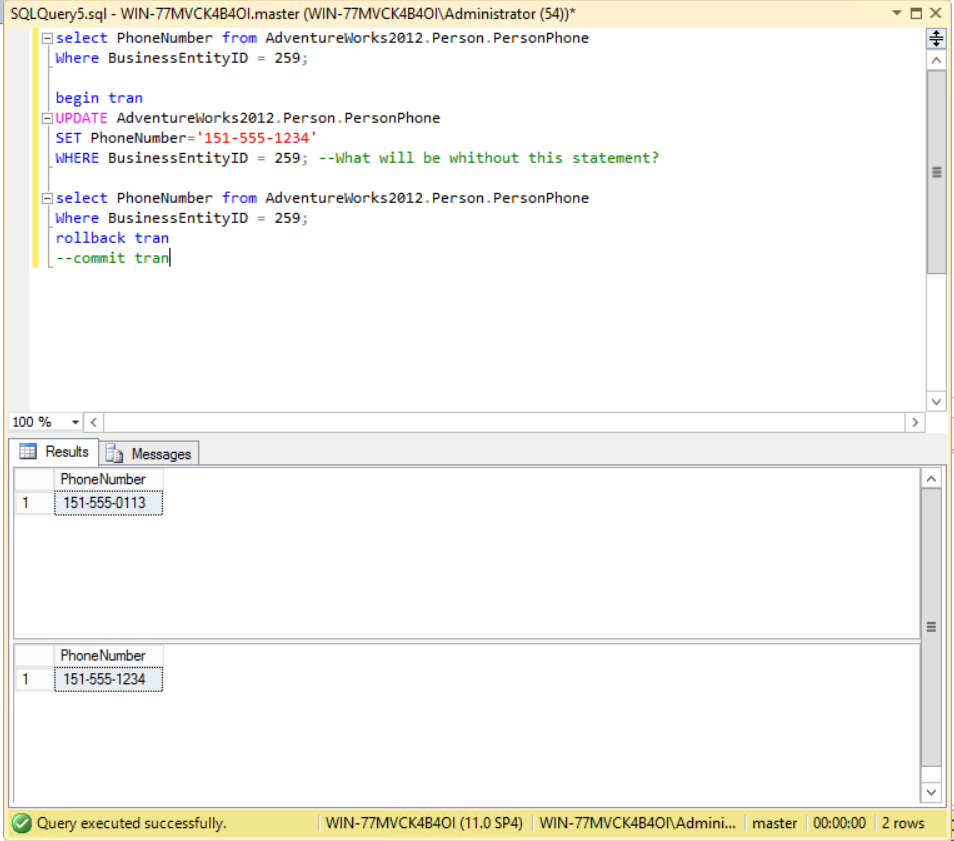
(делаем бэкап БД

)

(делаем бэкап лога ,ужимаем)

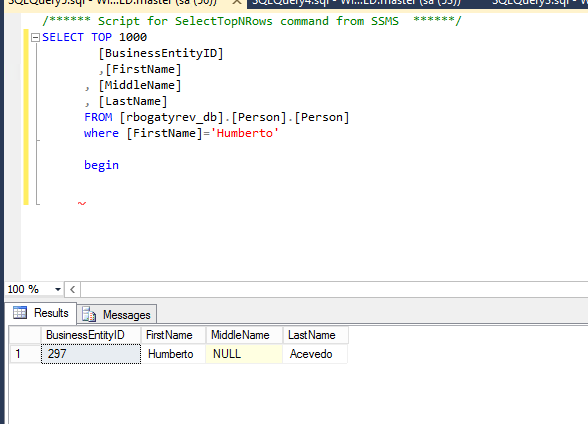
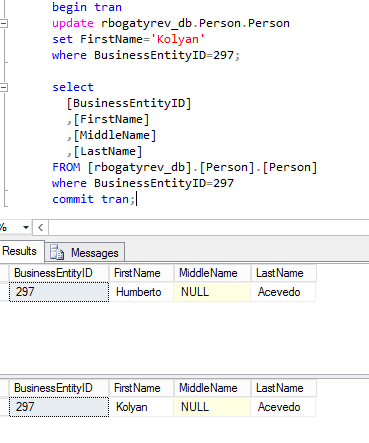
(размеры(верзний-бд,нижний-лог)

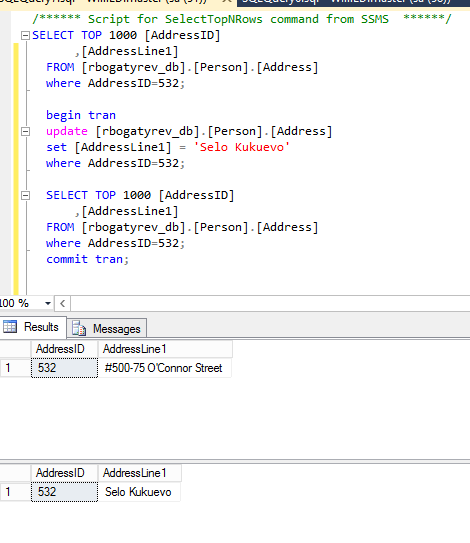
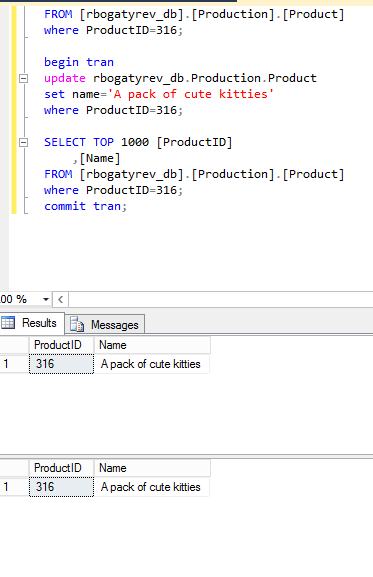
1. Perform your custom data changes in DB. See example on the picture:



The purpose of these changes – to make DB different than its backup, so update and change whatever you want. You can use transactions with SELECT statement to show changes, as done on the picture. Use COMMIT TRAN to apply changes to DB.

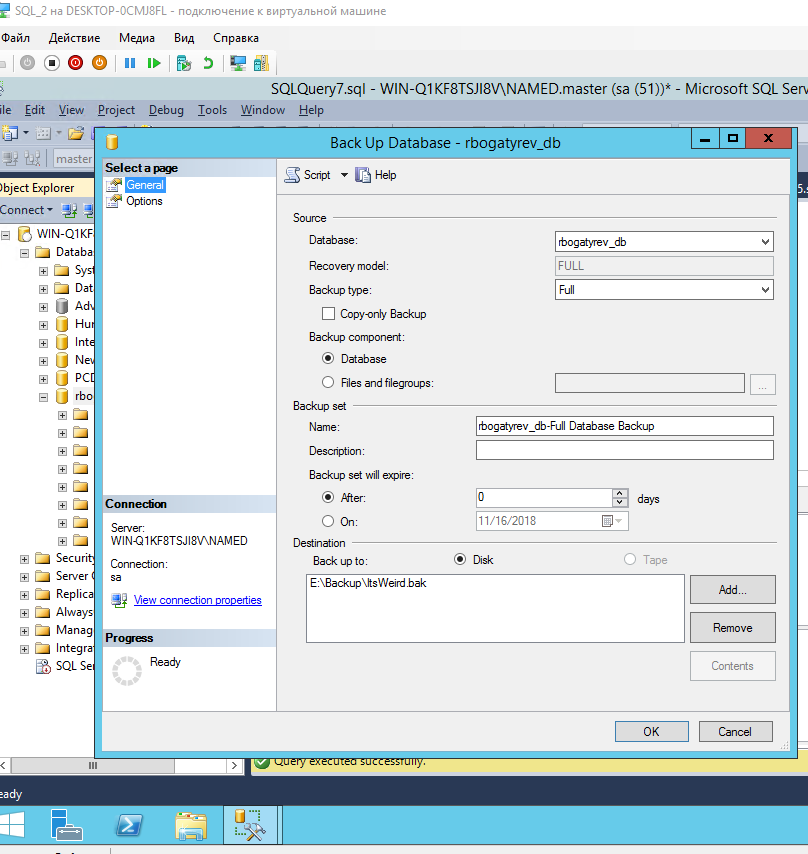
Use right click on the table – select top 1000 to see table structure and data, what you can change. Make changes at least at 3 different tables. (with screenshots of SELECT before – script to change something – SELECT after). Make differential backups for each table you change (before 1 table, before 2nd, before 3rd )

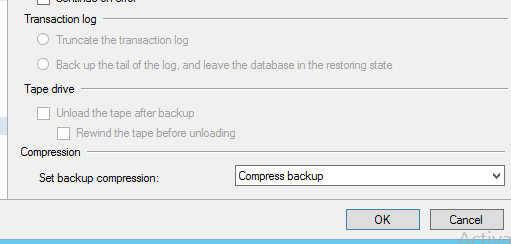
  (table 1)

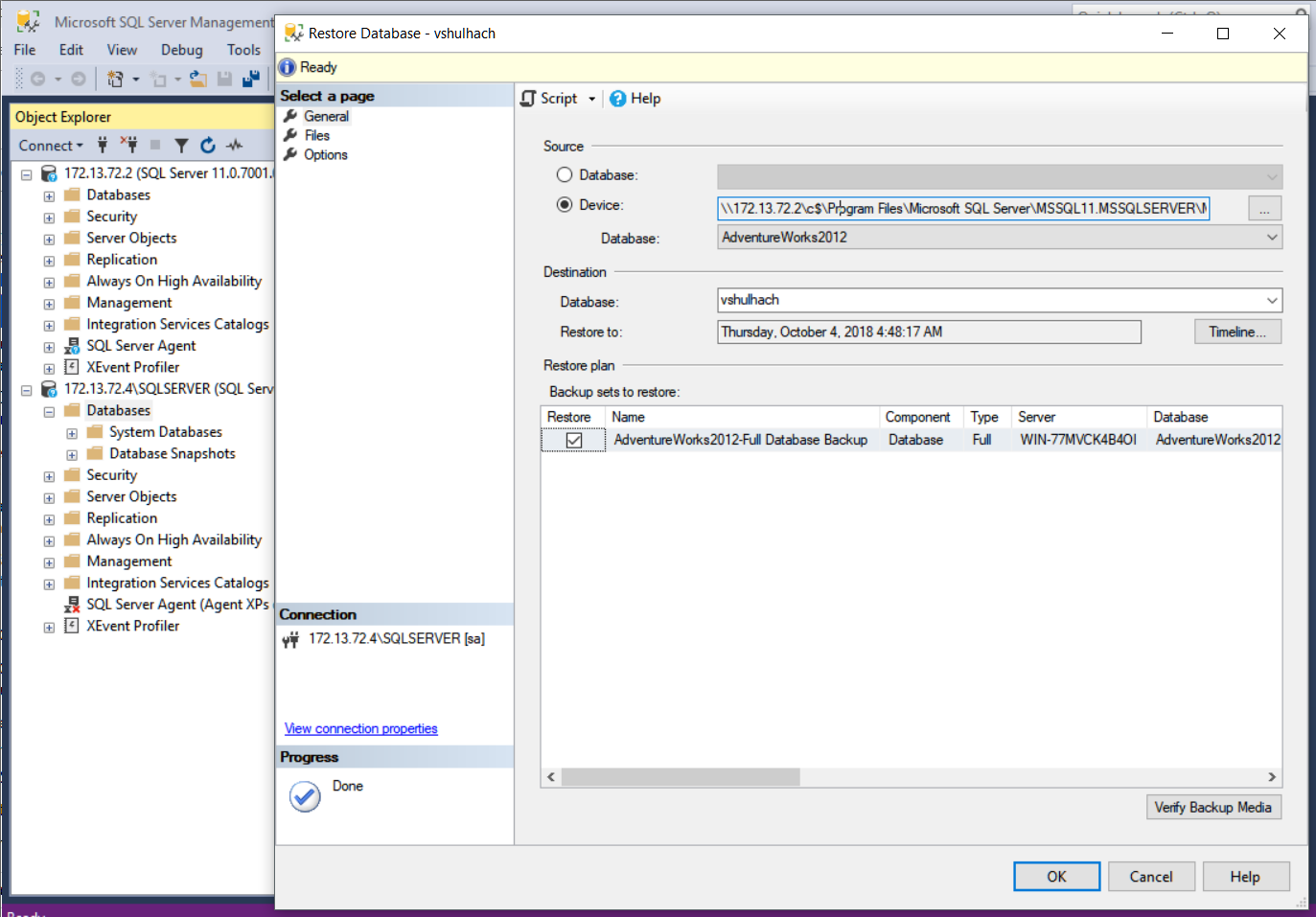


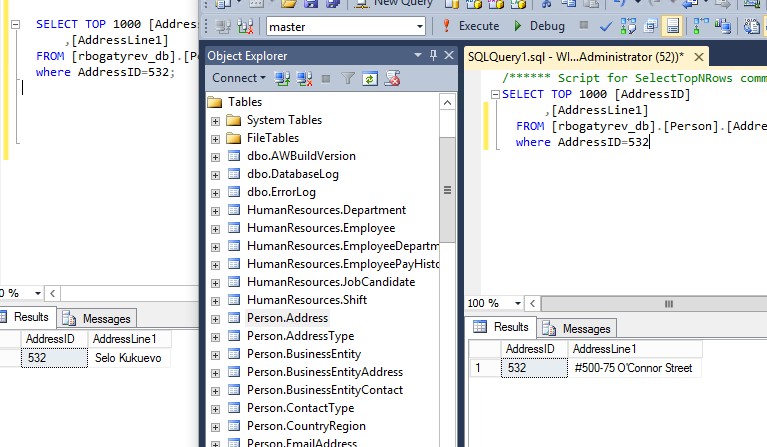
(table 2) (table 3)

1. Make Full backup of DB, create new media set. Use compression, no not overwrite previous media set with backups.



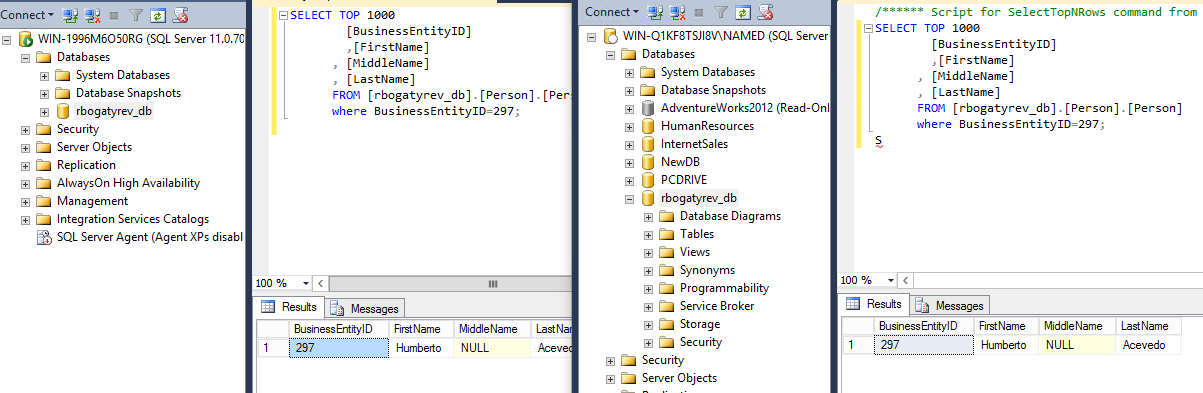
 Restore full backup, which was done in simple model, on the second server (the name should be custom, the same as on 1st server). Try to restore backup via network path from the first server. Skip network restoration if you cannot. Use SELECT statement to show, that data is not the same on both Servers – with the fresh DB and you customized one.



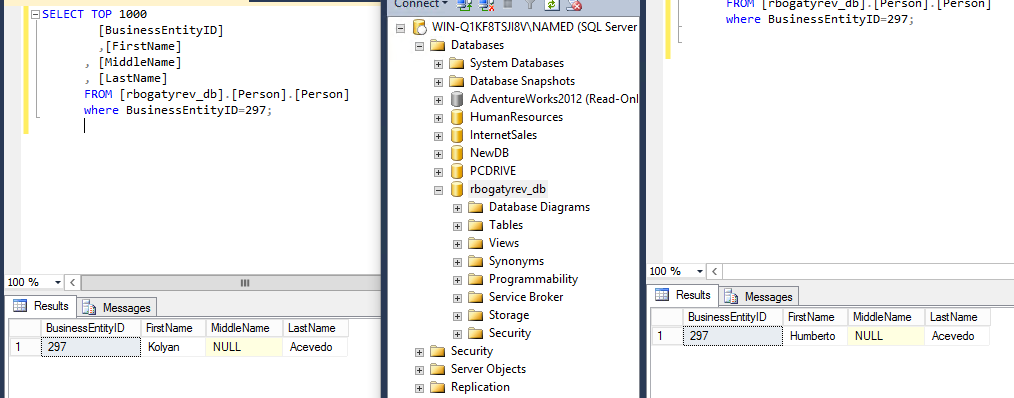


1. Restore differential backups in correct sequence on 2nd server, show the differences in tables step by step, according the order you changed them on first server.

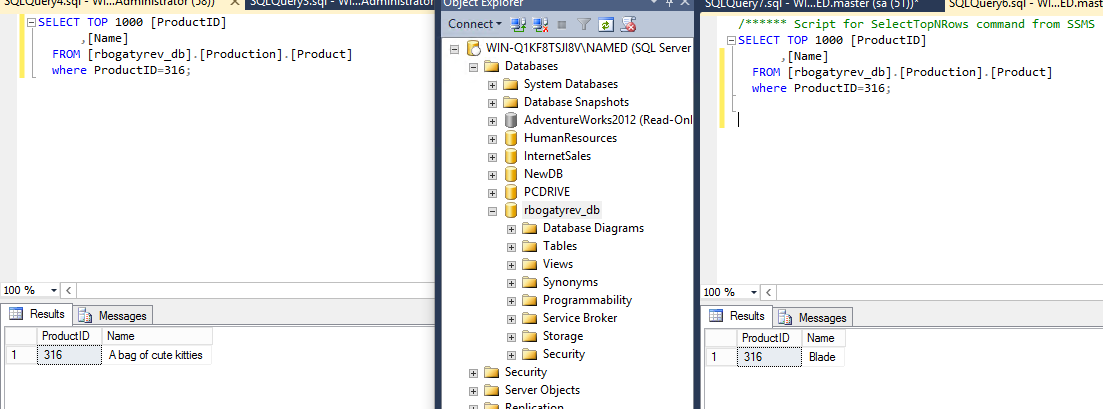
1 откат(все адекватно)(слева машина 2,справа машина 1)



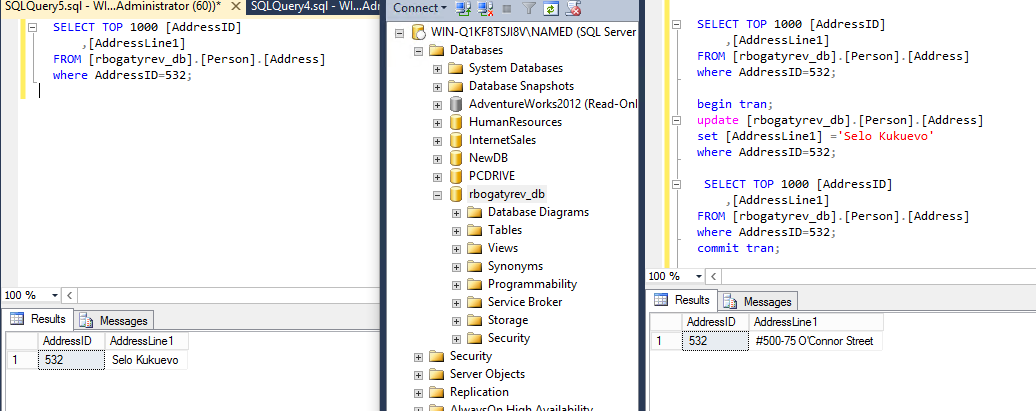
2 откат(меняется таблица)



3 откат(меняются 2 таблички)

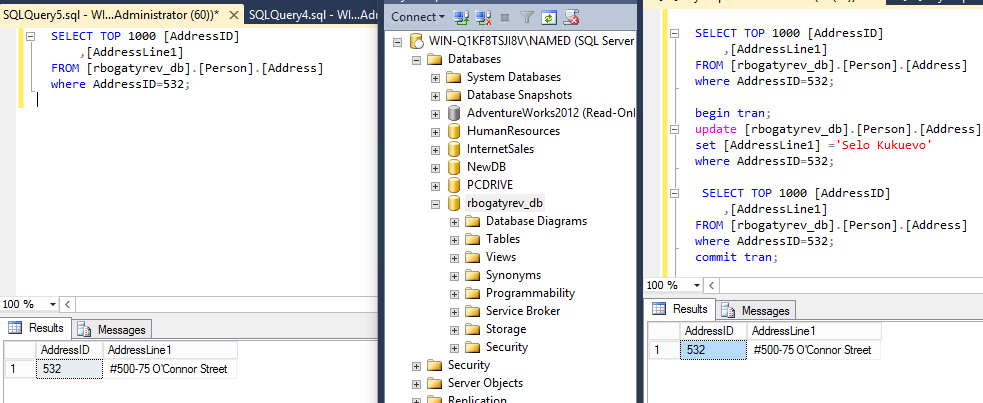


1. Copy full latest backup on second SQL server. Restore it with latest timeline point to newer created DB. Compare data with first server state.



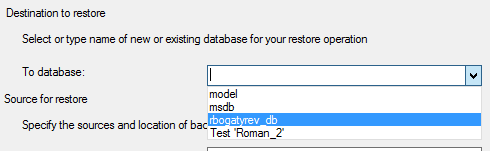
(добавилась 3 таблица к 2 предыдущим)

1. Restore N-1 timeline point. Show, that data is differ from the original server.



(Таблица 3 снова пропала)

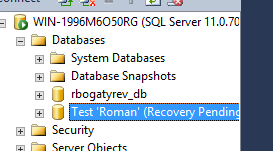
1. Create Database “Test\_’Student name’” on any instance. Make backup for master database. Create another database “Test\_’Student name’\_2”. Delete previously created database.

(мастер БД нет)

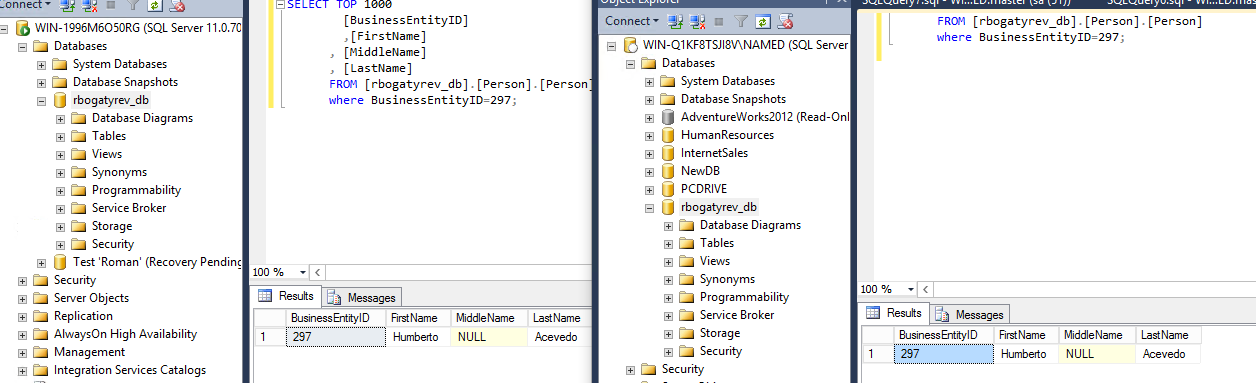
1. Recover DB Master from backup. Pay attention, what is the result after recovery and why. See additional materials for help.

Hint: in case of server will be busy by any identity, and you cannot login to it – you can exactly define application to use SQL in the single mode.

Hint 2: Everything is working using SQLCMD.EXE. That task could be done not using scripting, I hope you will never have to automate master DB recovering.

(2 БД пропала,вернулась 1,т.к мастер хранил данные о первой базе,а не о второй)

1. Make “Student\_name” db the same on both servers. It is the last basic task.



Extra tasks:

Task 1. Create a PowerShell wrapper for scenario, that should perform next:

1. Create a backup for Adventure Work DB, and restore it on the instance 1.
2. Make update of any table with SELECT Before and SELECT after.
3. Create full compressed backup of DB
4. Restore it on second Instance.
5. Write everything in log file, Log file should be Human readable, with commented steps. (Tee-Object can help)

Task 2. Perform Task from presentation. During task creating keep in mind and perform MS recommendation for better performance (Should be in script).