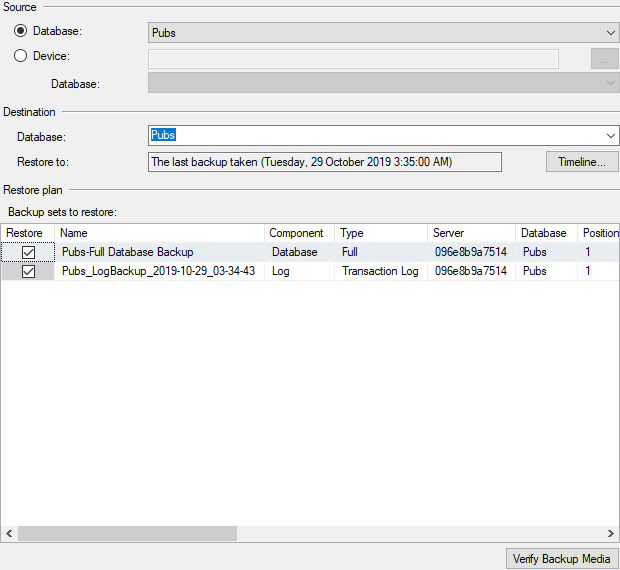
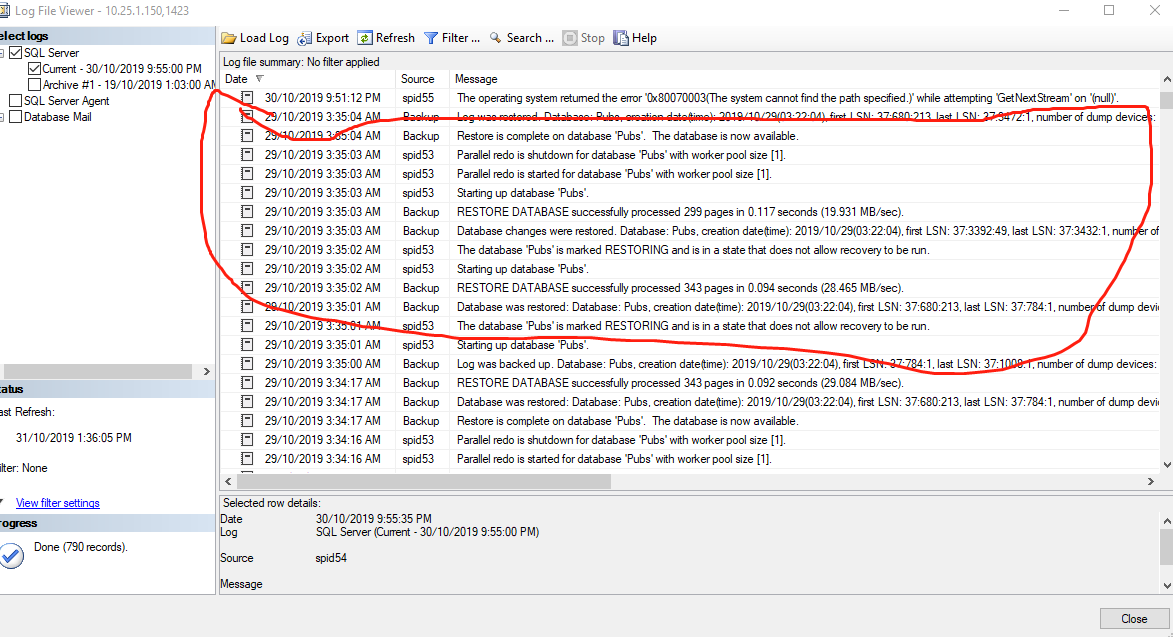
# Week 13 Practical

# Duties of a DBA

## Examine each of these carefully, screen captures and notes can be used as proof of completion. Perform any additional setup as required to complete these tasks.

Part 1:

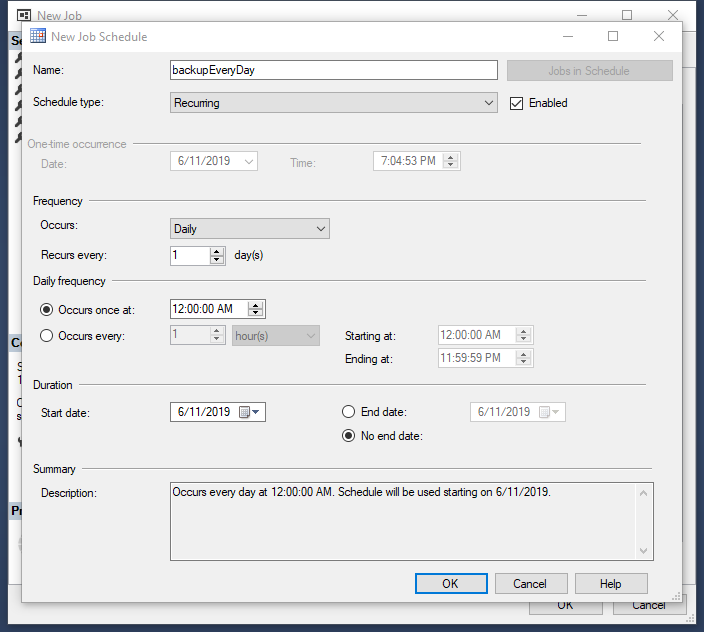
* Check the previous night’s SQL Server database and transaction log backups and SQL Server Agent jobs for errors.
* In this case, we are using the Ubuntu Docker, So by default the SQL Server Agent it does not enabled. We cannot start it by Microsoft SQL Server Management Studio, To solve this:
  1. Login into container with command docker exec -it hua /bin/bash
  2. Enable SQL Server Agent with command sudo /opt/mssql/bin/mssql-conf set sqlagent.enabled true
  3. Now, we can use SQL Server Agent to do some easy working
* 
* 
* Automate a daily backup schedule (Full and Differential)

There are three ways to backup:

* 1. User SQL Server Agent
  2. Under the Management Plans right click to go backup wizard
  3. Choose database, right click goes to task to backup

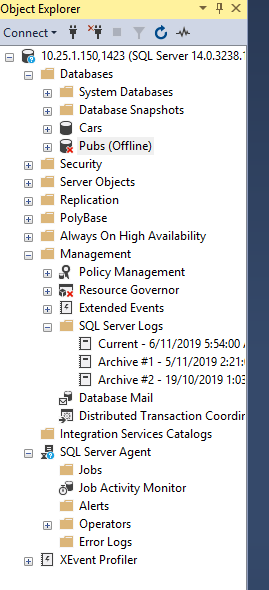
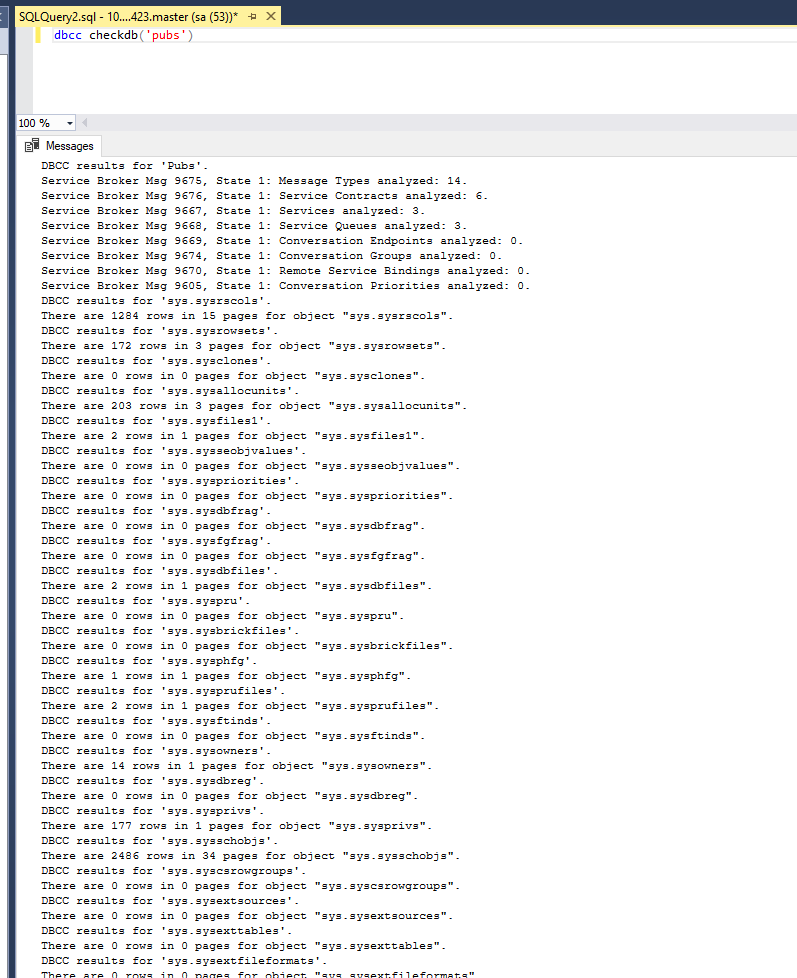
To set automate schedule using a SQL Server Agent job

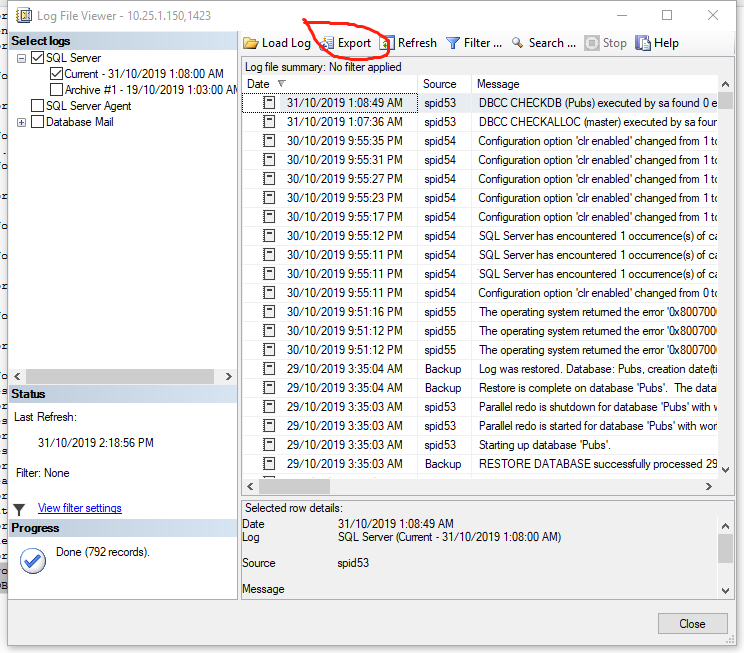
* 1. Create a new job
  2. Under the Schedule tab select an occurring frequency, duration and a start end date, Schedule type, name and click OK, done.

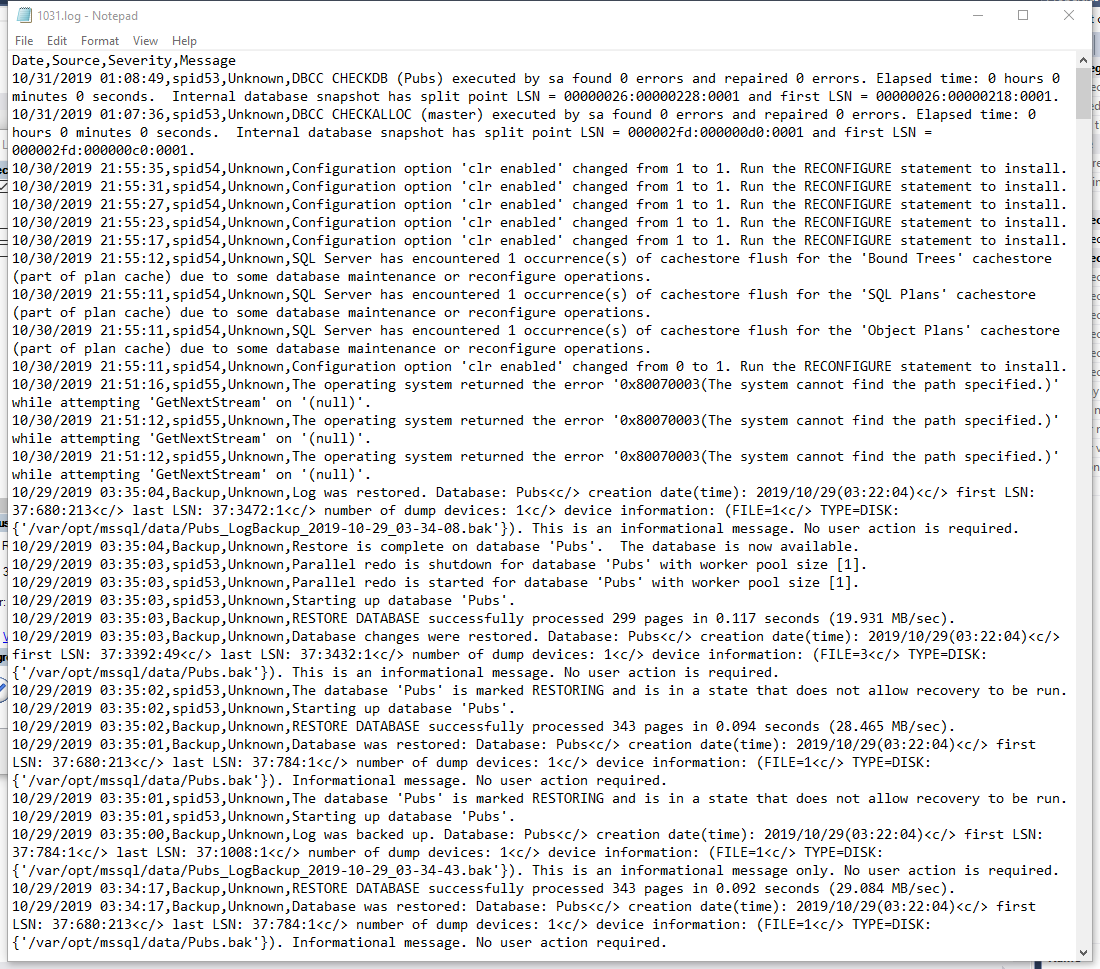


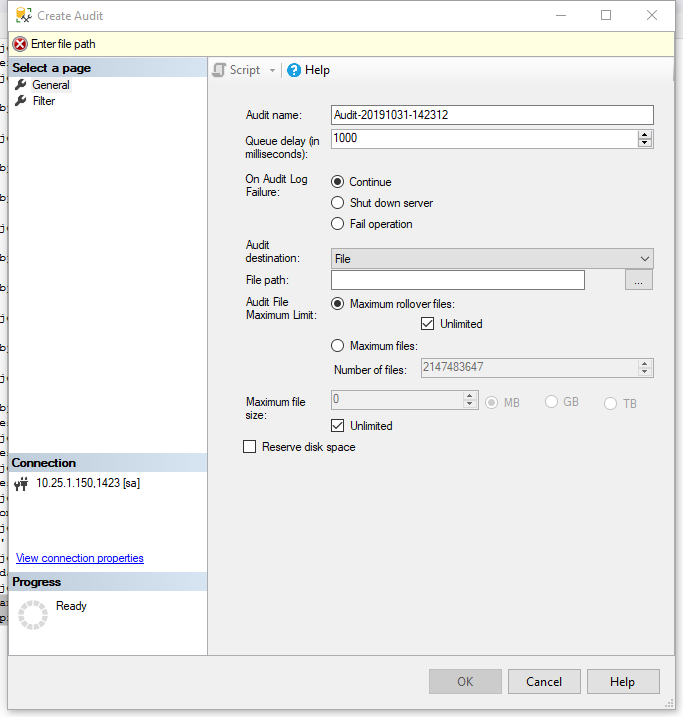
* Check all databases to make sure all are up and not marked as suspect. Check previous DBCC CHECKDB for errors.

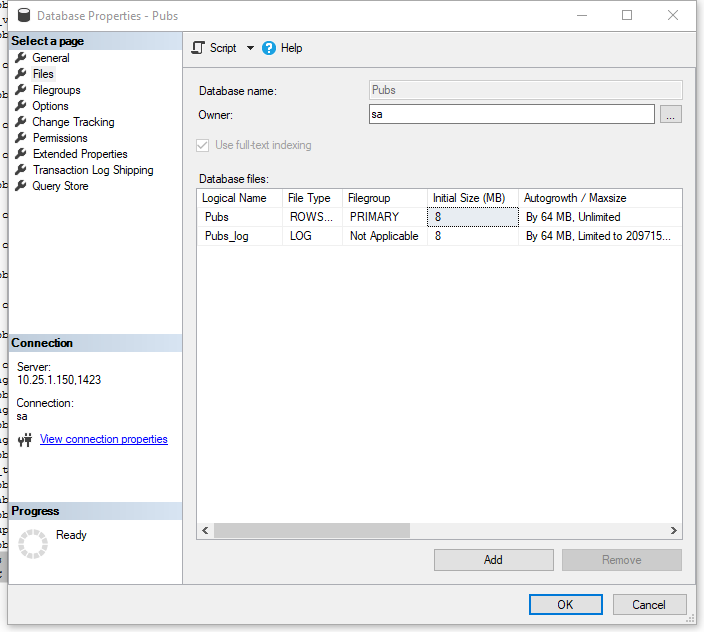
CHECKDB found 0 allocation errors and 0 consistency errors in database 'Pubs'.

* DBCC execution completed. If DBCC printed error messages, contact your system administrator.
* 
* 
* Check SQL Server Log File entries for warnings and errors and determine if any entries warrant further investigation. Export and save the current log file.

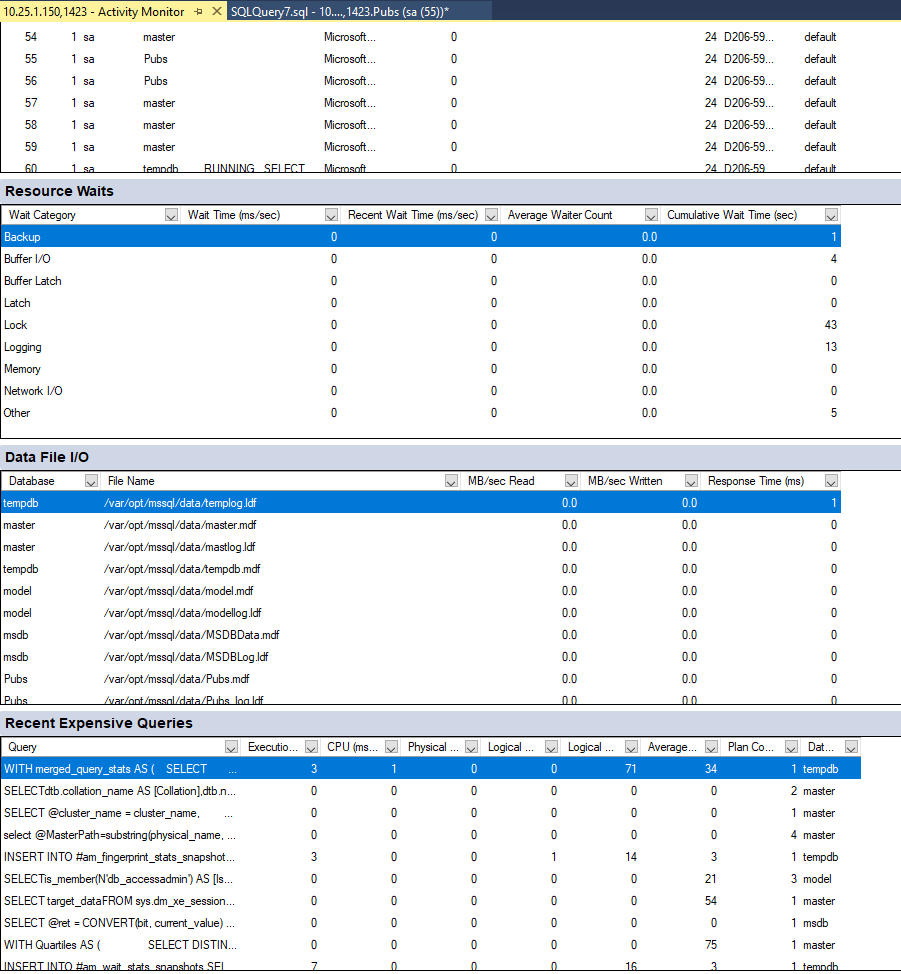


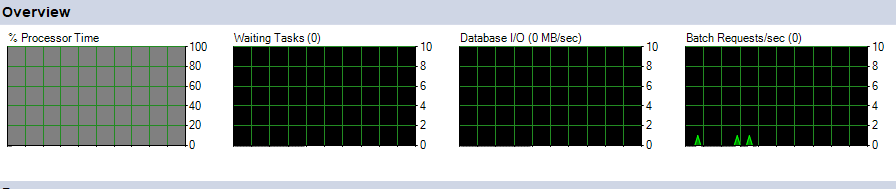


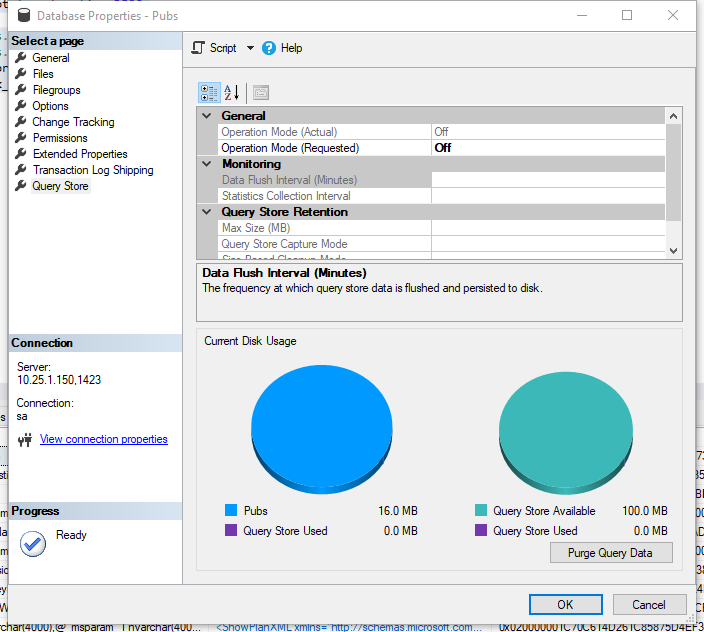
* Look for any security policy violations. Look for resources on the server, such as file sizes and disk space, and audit growth for long-term projections.
* 



* Explore using long-running queries or tasks, Perfmon, etc. to generate data. Set up a sensible logging report to monitor disk and memory usage.

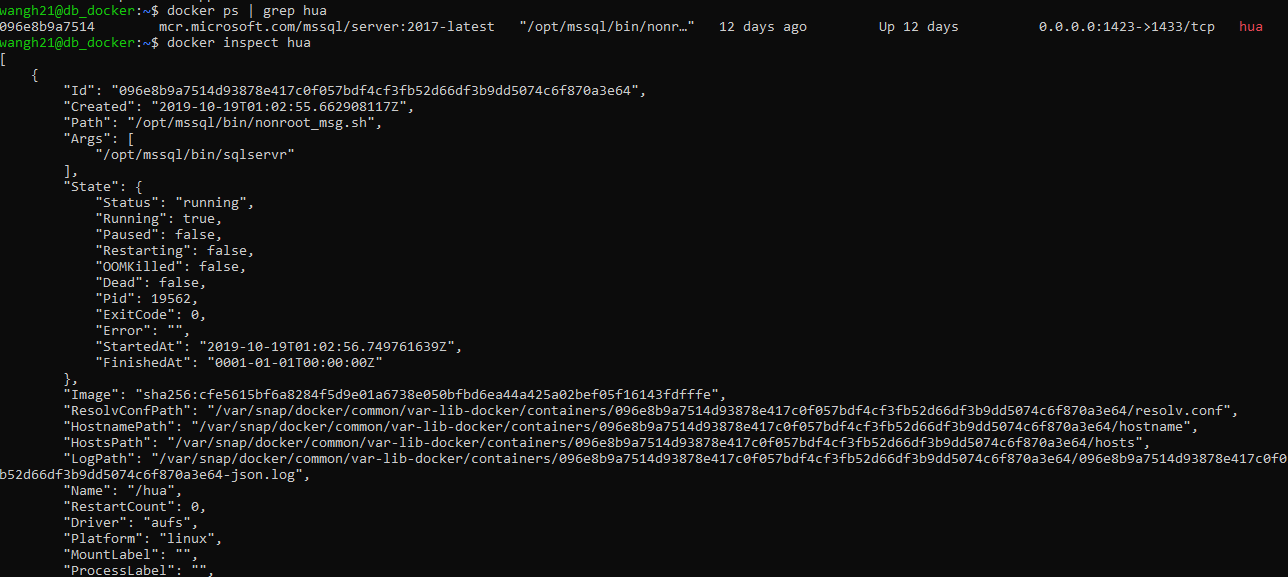


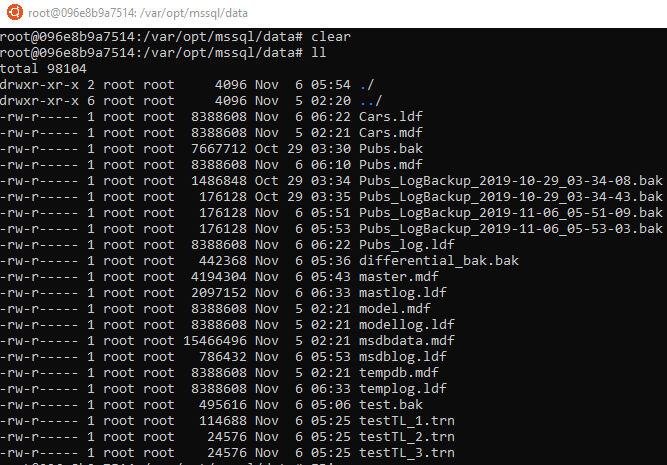




* What about your Container? What sort of usage data can you extract?

My Container working properly.





* 1. I have made this Docker image to my Docker Hub repository. So that I can still use this container without Otago Polytech virtual machine. To do this:  
       
     docker login  
     docker commit <my container id> aemooooon/mssqlserver:latest  
     docker push aemooooon/mssqlserver:latest
  2. About Database backup, normally just like the question above asking, Use Microsoft SQL Server Management Studio to back up, restore, create jobs, schedule and so on, also we can do development on there.
  3. About Docker database files backup. Because it is based on virtual machine, so sometimes we need concern more. So next time when I use container to run MS SQL SERVER, I will use volumes when I create my custom image, so that I can store the database mdf, ldf and backup file in volume. And also I can use WinScp tools to download the files to somewhere which is safer and stable place.

1. Create a file named as ‘Dockerfile’ and put the code below in it:

FROM mcr.microsoft.com/mssql/server:2017-latest

ENV ACCEPT\_EULA=Y

ENV SA\_PASSWORD=Test@12345

ENV MSSQL\_PID=Developer

ENV MSSQL\_TCP\_PORT=1433

WORKDIR /src

VOLUME ["/opt/project"]

RUN (/opt/mssql/bin/sqlservr --accept-eula & ) | grep -q "Service Broker manager has started" && /opt/mssql-tools/bin/sqlcmd -S127.0.0.1 -Usa -PTest@12345

2. Run command ‘docker build -t="aemooooon/huasqlserver" .’ to build new image.

### Use your pubs database

Part 2. Task 1 (Task 2 will follow next week)

You want to recover a single table from a database backup – why?

It is the only table effected by a recent data loss.

Restoring an entire backup can take a significant amount of time, and you are under a lot of pressure to get it done fast

Perform the necessary data adjustments to check your solution is correct (delete rows etc). Provide a script for each of the following scenarios:

1. The table still exists, but only some rows were deleted, restore the deleted data only.
2. Create a new database named as ‘temp\_backup’
3. Restore the Pubs backup files to temp\_backup database with command below:

USE [master]

RESTORE DATABASE [temp\_backup] FROM DISK = N'/var/opt/mssql/data/Pubs\_LogBackup\_2019-11-07.bak' WITH FILE = 1, MOVE N'Pubs' TO N'/var/opt/mssql/data/temp\_backup.mdf', MOVE N'Pubs\_log' TO N'/var/opt/mssql/data/temp\_backup\_log.ldf', RECOVERY, NOUNLOAD, REPLACE, STATS = 5

GO

1. Run command below to insert the data that we have missed:

USE Pubs

INSERT INTO Pubs.dbo.test\_signle\_table\_backup SELECT \* FROM temp\_backup.dbo.test\_signle\_table\_backup WHERE id NOT IN(SELECT id FROM Pubs.dbo.test\_signle\_table\_backup)

GO

1. The table has been too badly damaged; restore the table structure and all the data.
2. Create a new database named as ‘temp\_backup’
3. Restore the Pubs backup files to temp\_backup database with command below:

USE [master]

RESTORE DATABASE [temp\_backup] FROM DISK = N'/var/opt/mssql/data/Pubs\_LogBackup\_2019-11-07.bak' WITH FILE = 1, MOVE N'Pubs' TO N'/var/opt/mssql/data/temp\_backup.mdf', MOVE N'Pubs\_log' TO N'/var/opt/mssql/data/temp\_backup\_log.ldf', RECOVERY, NOUNLOAD, REPLACE, STATS = 5

GO

1. Whatever how the table damaged is? Delete the table and create a new one and grab the data from backup then insert to target table, the command below:

USE Pubs

IF EXISTS (SELECT \* FROM sys.tables WHERE name = 'test\_signle\_table\_backup' AND type = 'U') DROP TABLE test\_signle\_table\_backup;

GO

CREATE TABLE test\_signle\_table\_backup(

id bigint primary key not null,

name nchar(10) null,

password nchar(10) null,

email nvarchar(50) null

)

TRUNCATE TABLE Pubs.dbo.test\_signle\_table\_backup;

GO

INSERT INTO Pubs.dbo.test\_signle\_table\_backup

SELECT \* FROM temp\_backup.dbo.test\_signle\_table\_backup;

GO