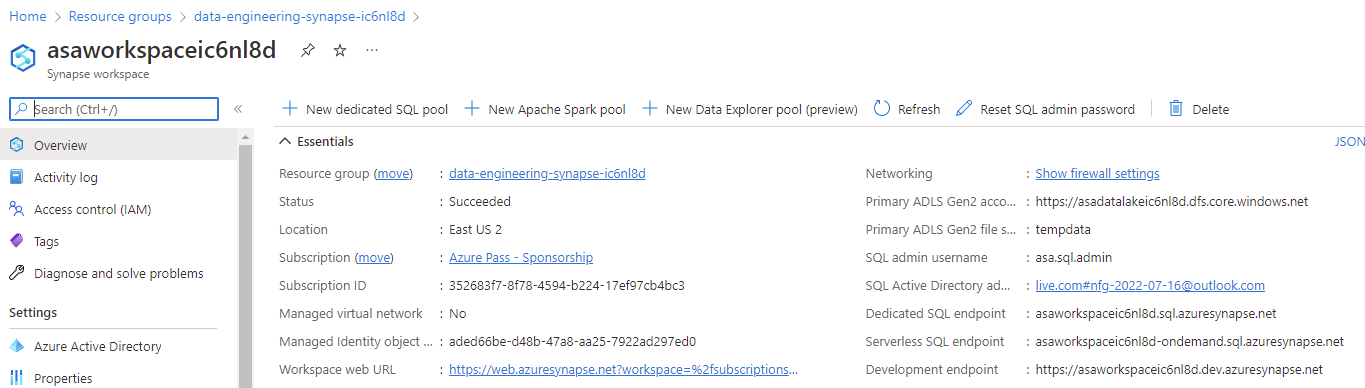
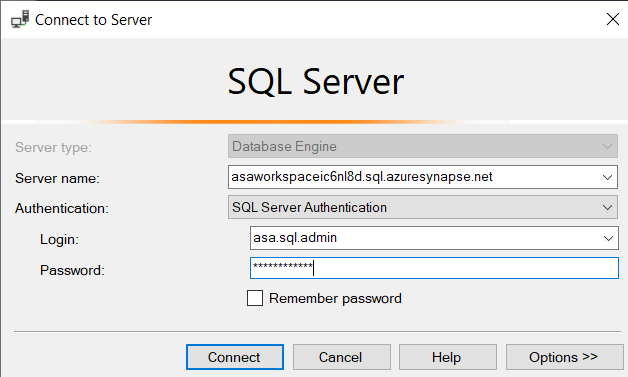
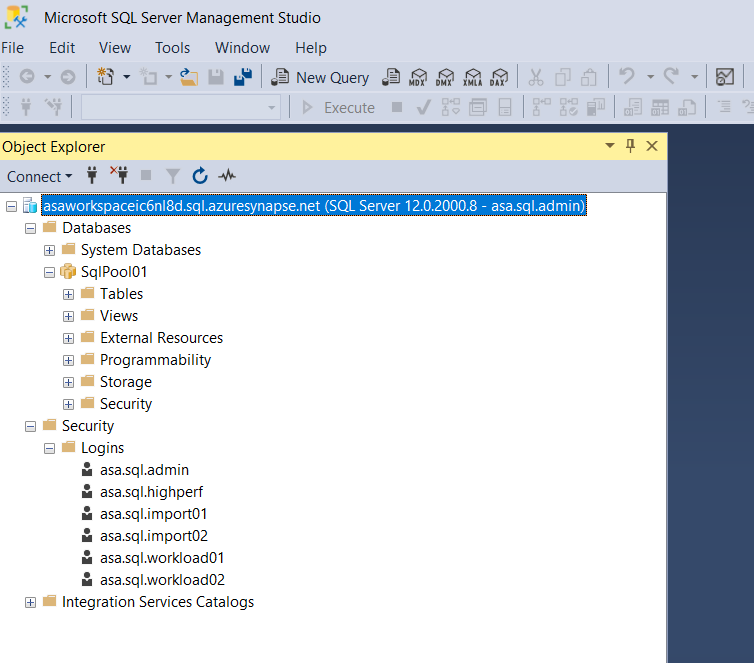
Lab4 Notes

Make sure you name the dedicated SQL Pool, **SqlPool01**. Please double check that it ends with 01 (**ZERO** ONE). A lot of other tasks in this and other labs depends on that.

You need to pass to the setup the name of the RG (asaworkspaceSUFFIX). The script has been modified to avoid asking for the SQL Password you assigned in the first script. My suggestion was P@ssw0rd1234. So, you probably don’t need to worry about what was the password. But if you want to open SSMS and connect to the SQL server service in Azure and check if everything was indeed created you will need the password. You can reset the password in Azure portal.

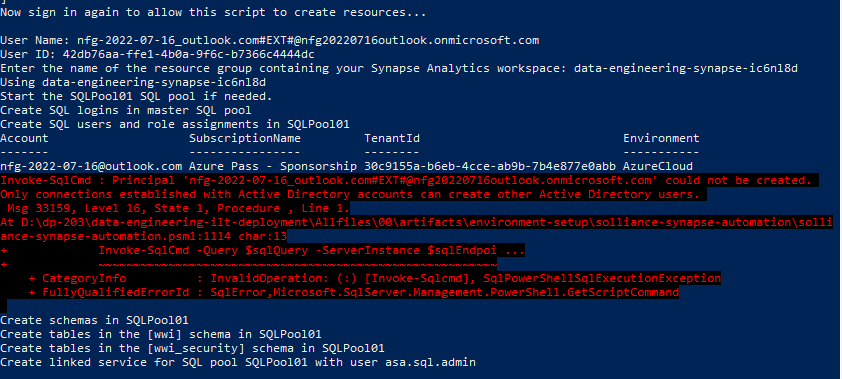


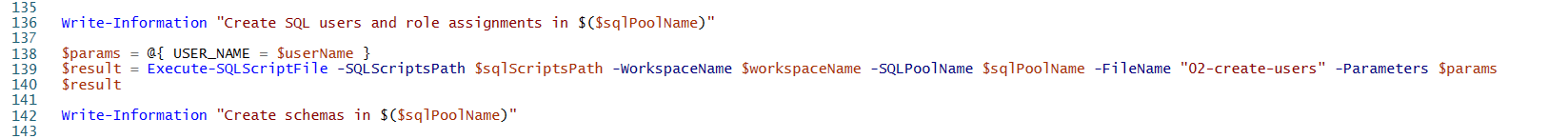
 

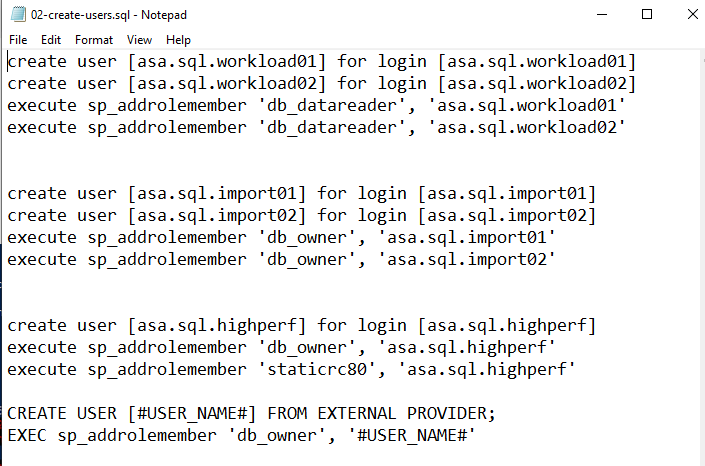
If the AZ CLI is updated, the objectId on line 55 should be replaced by id  
  
But this error seems to have no consequences.

Some errors occurred, but the setup-sql.ps1 completed.

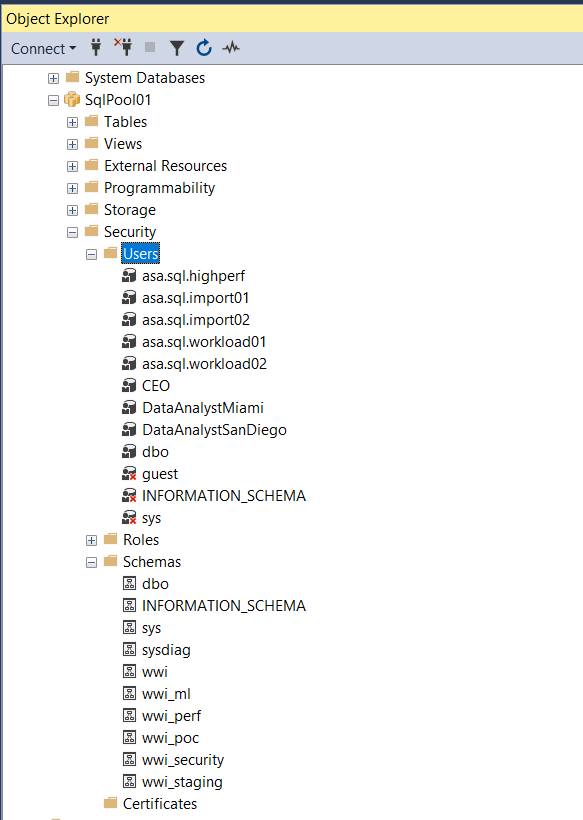
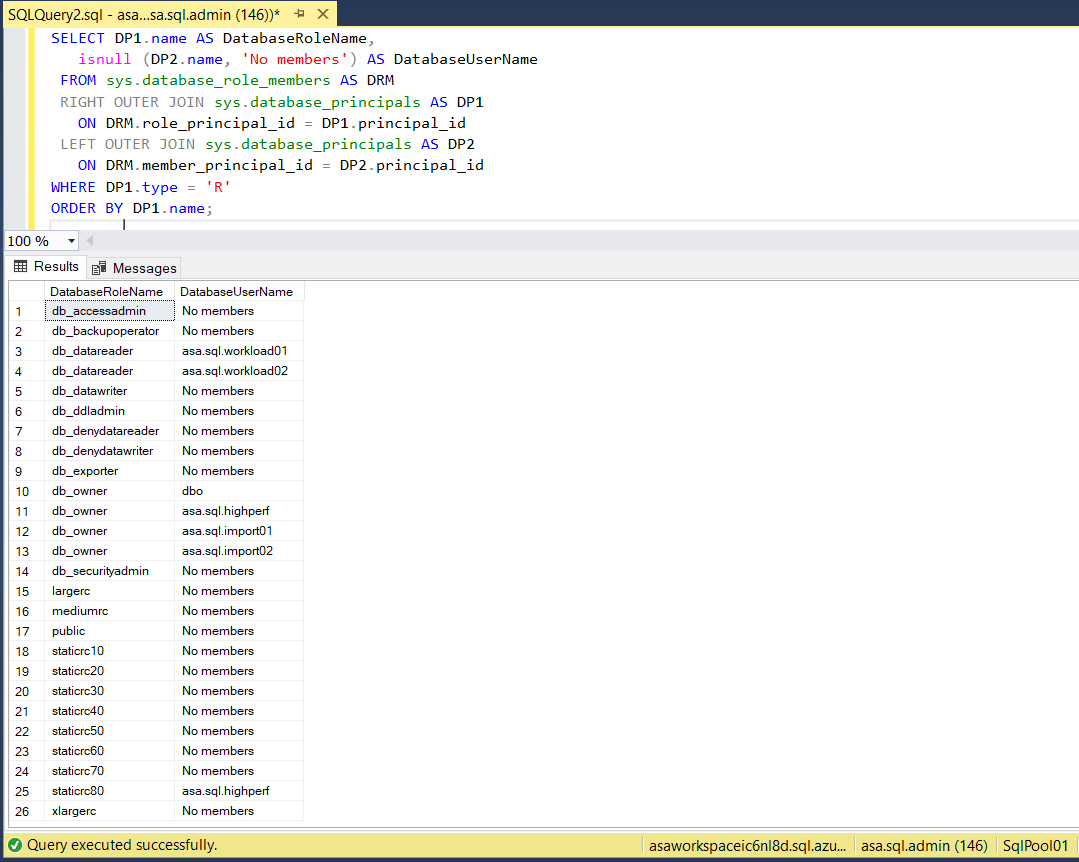
The first was a principal that could not be created.



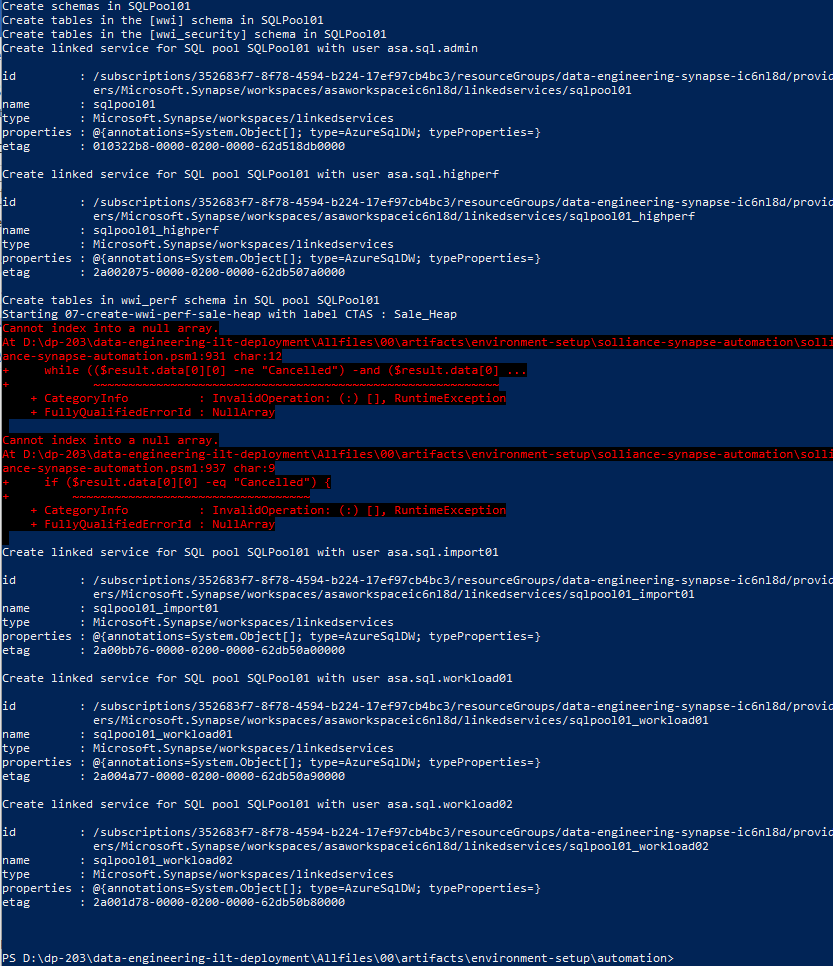




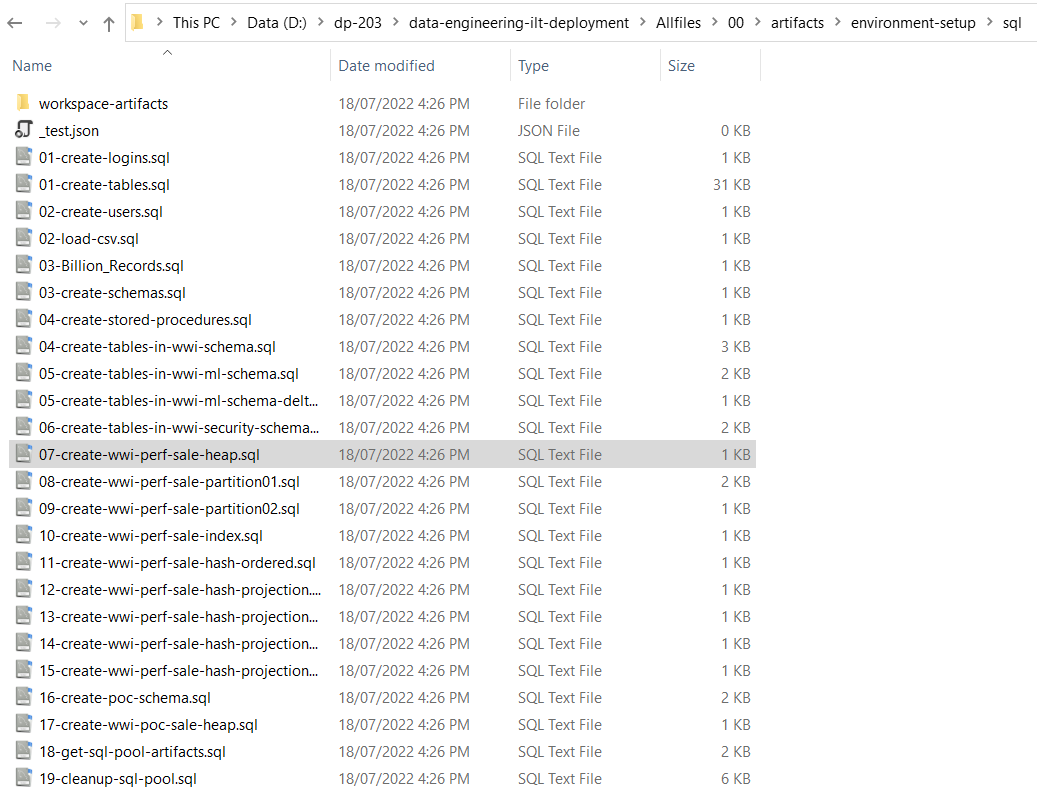
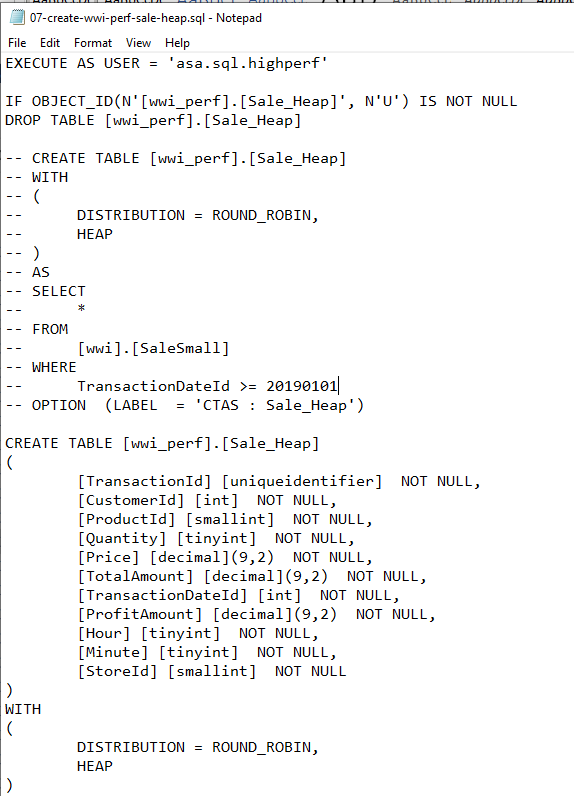
It was the last CREATE USER that caused the error. This error has no apparent consequences.

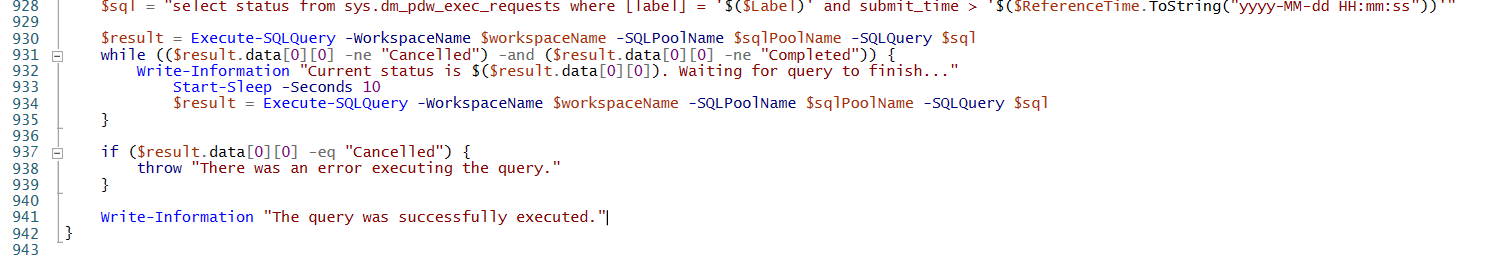
Then the same error “Cannot index a null array” happened twice:



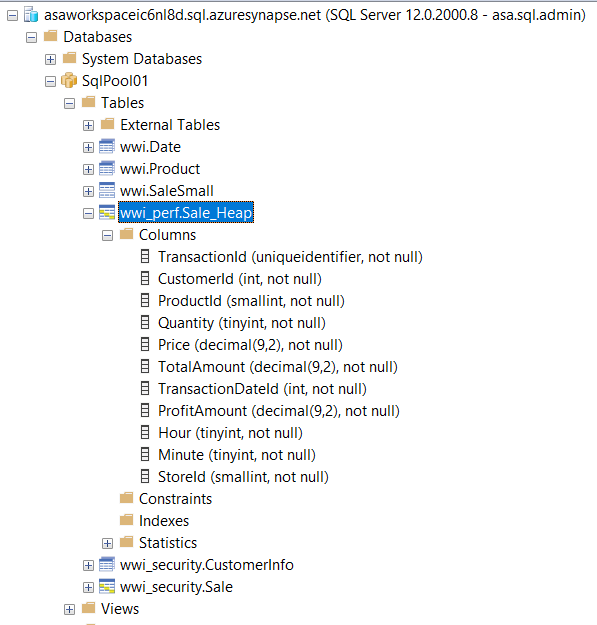


The SQL script run without any issues. The problem was the call to **Wait-ForSQLQuery** that is in the imported module "..\solliance-synapse-automation" lines 931 and 937.

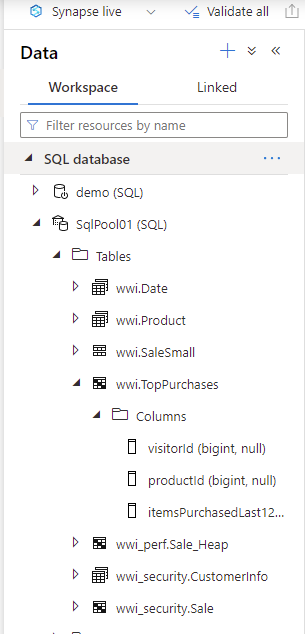


The reason is that in the SQL script the code that was assigning a label was commented out. So, there is no label and $result is empty. There are no consequences from this error. The important step was the CREATE TABLE, and that was fine as you see below:



So, we can safely ignore the 3 errors in **setup-sql.ps1**

Exerc. 4



If you want 2 decimal places, as in the screenshot in the lab steps, you have to modify the code as below:

