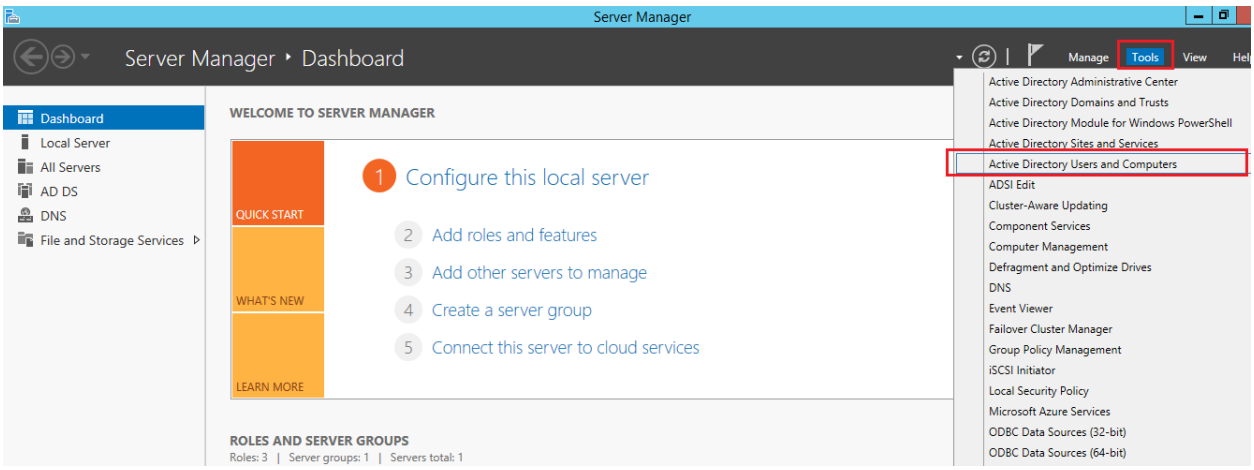
Configuring a SQL Server AlwaysOn High Availability Group

Configuring a SQL Server AlwaysOn High Availability Group

In the following steps, we will create a Service account to connect SQL server instances. The Service account will be used to connect the SQL instance from both the node servers.

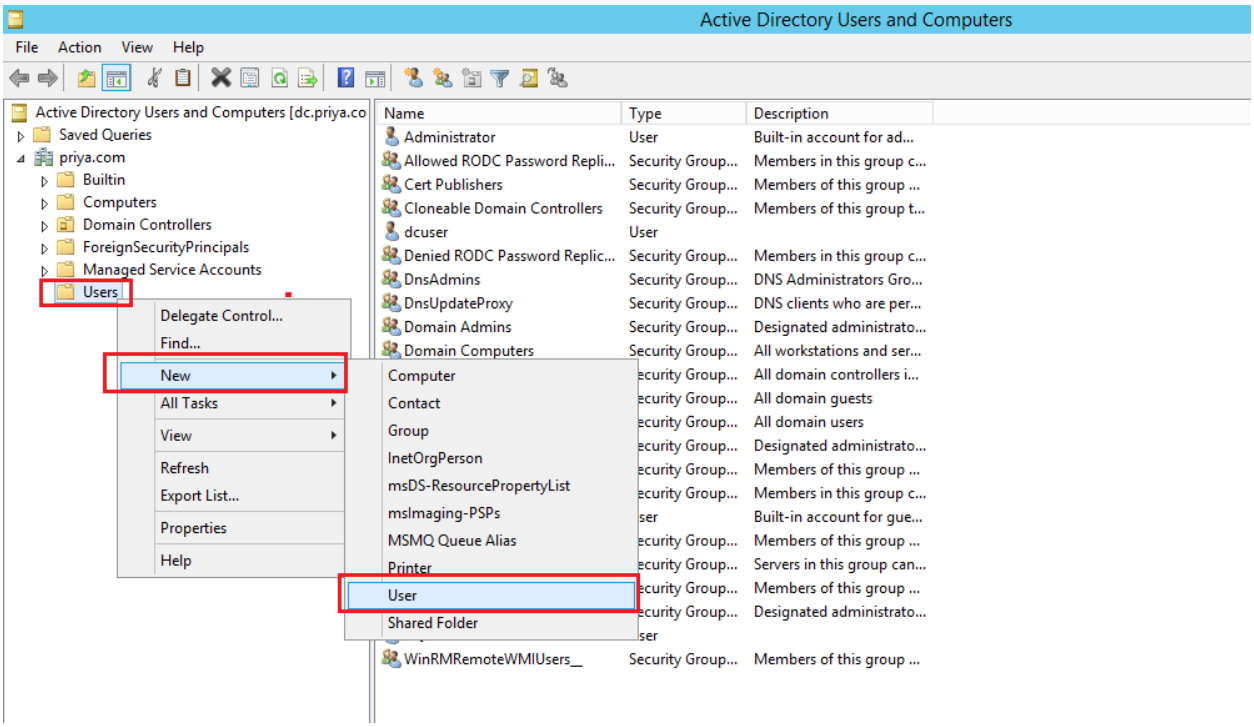


Create a Service Account

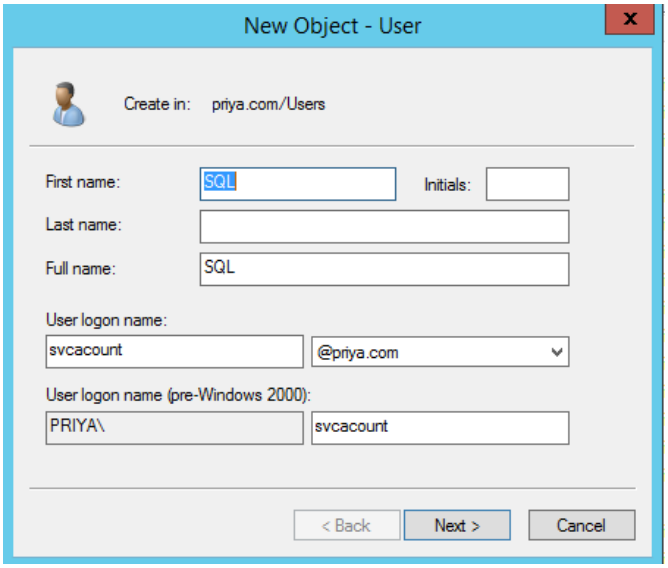
We will log in to the Domain controller and perform the following steps.

Open **Server Manager**, **Tools** and Click **Active Directory Users and Computers**.

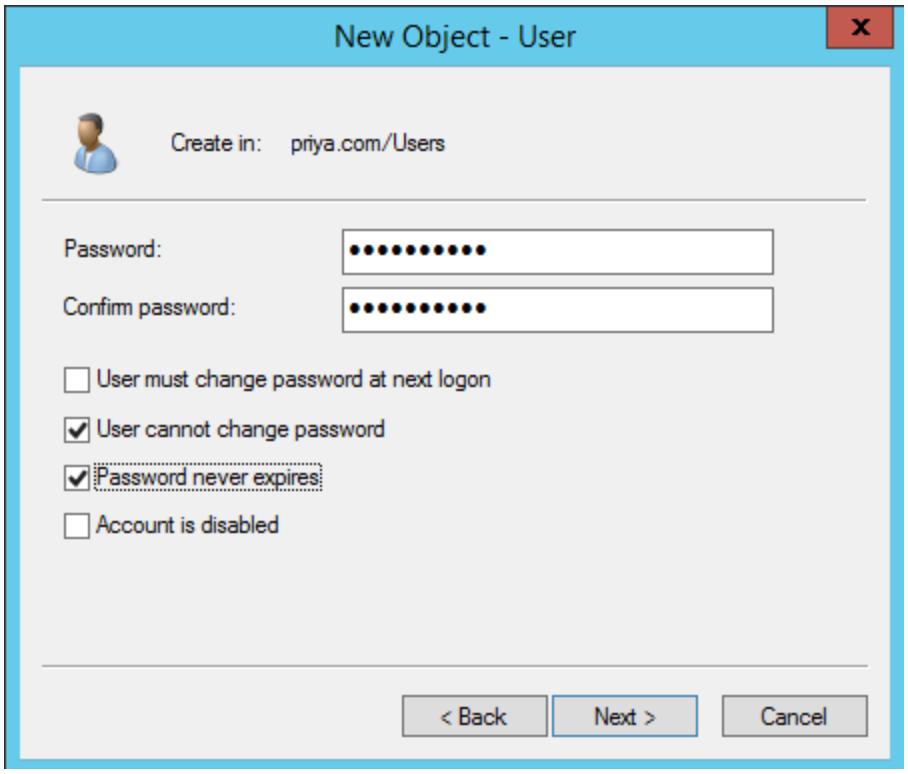
In Active Directory Users and Computers window, **Right-Click Users > New > User** as shown in the following Fig.



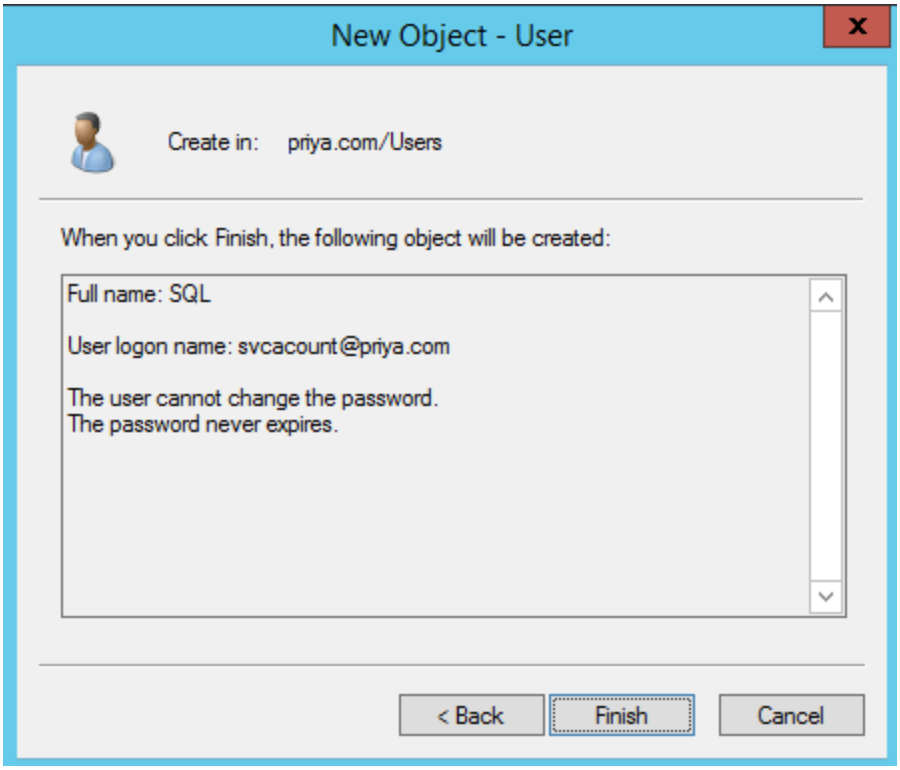
In this step, we will create a Service account username, Specify the service account login name and Click Next



Specify a Password and tick the options for **User cannot change password** and **Password never expires**. We need to select **Password never expires** option because, if the password expired, the service account login would fail, and it will cause a problem to the high availability configuration.



Click **Finish** to complete the service account creation process.



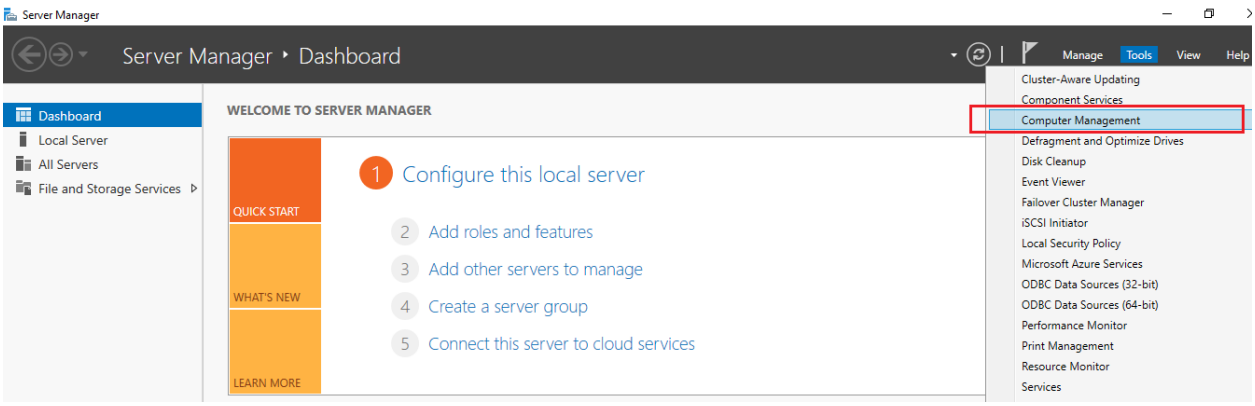
Configure Administrator permission, windows user and login for the service account

Once the Service account created on the Domain controller, we need to add service account as a member of the administrator account on both nodes.

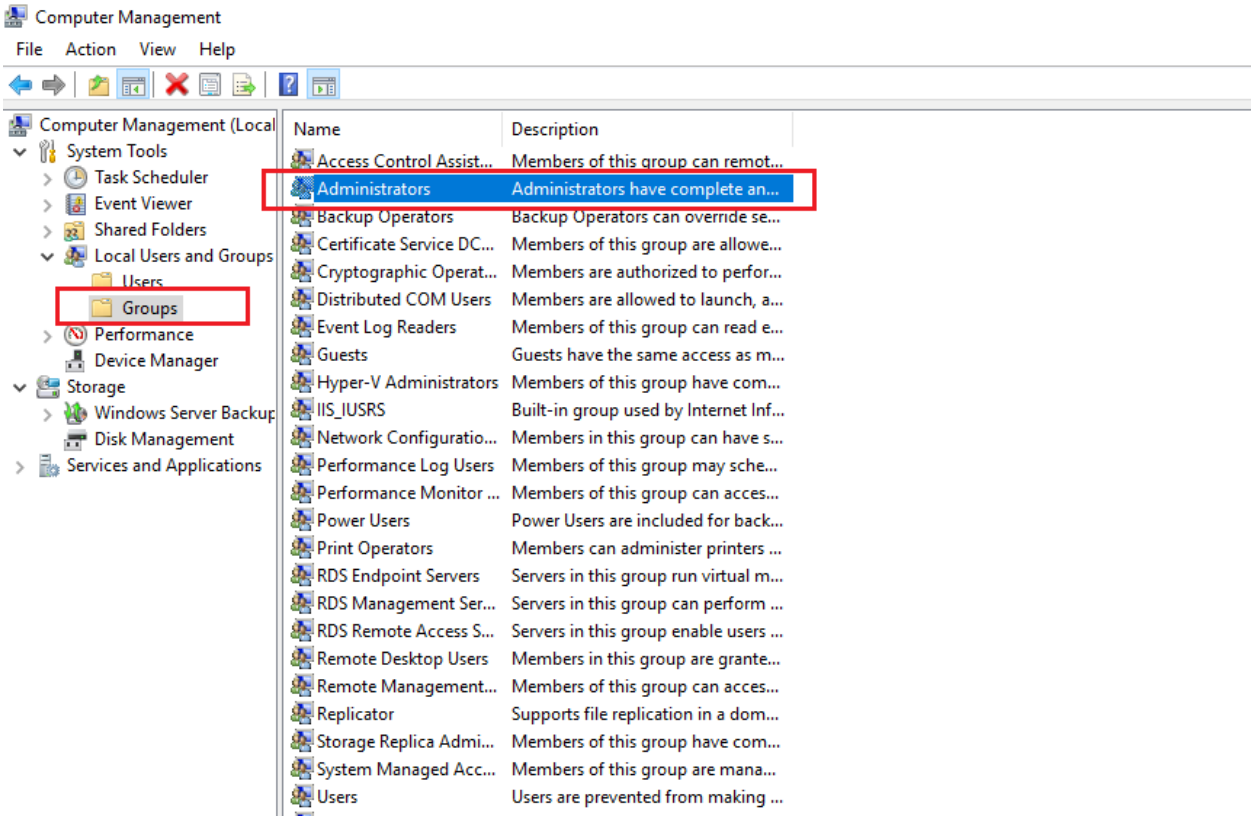
Connect to both node servers SQL1 and SQL2 and perform below steps

- Add Service account as an Administrator
- Create a Windows user for the Service account
- Create a Windows authentication login for a service account login in the SQL database instance

Open **Server manager** and Select **Tools** and **Computer Management**

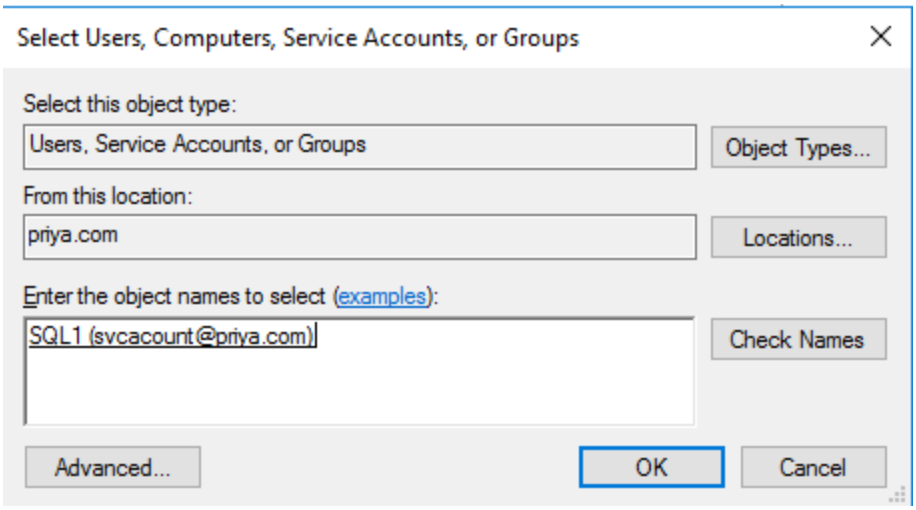


In **the Computer management window**, Select **Groups** and Right-Click **Administrators** and Go to **Properties**



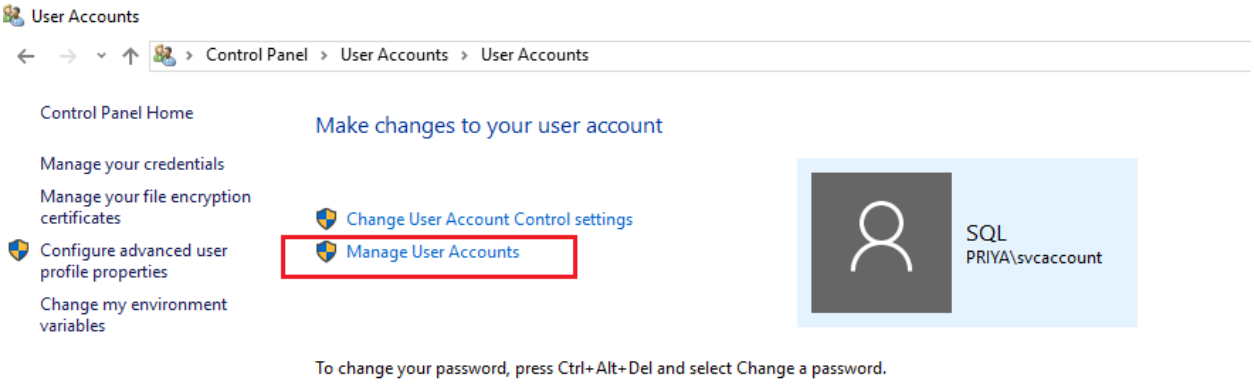
Add User, Enter the Service account login name and Check names and Click **OK**

Note – Once clicked on the Check Names, it may ask to enter username and password pop-up window. In the pop-up window, we need to register a domain controller username and password and make sure the location should select as a directory.

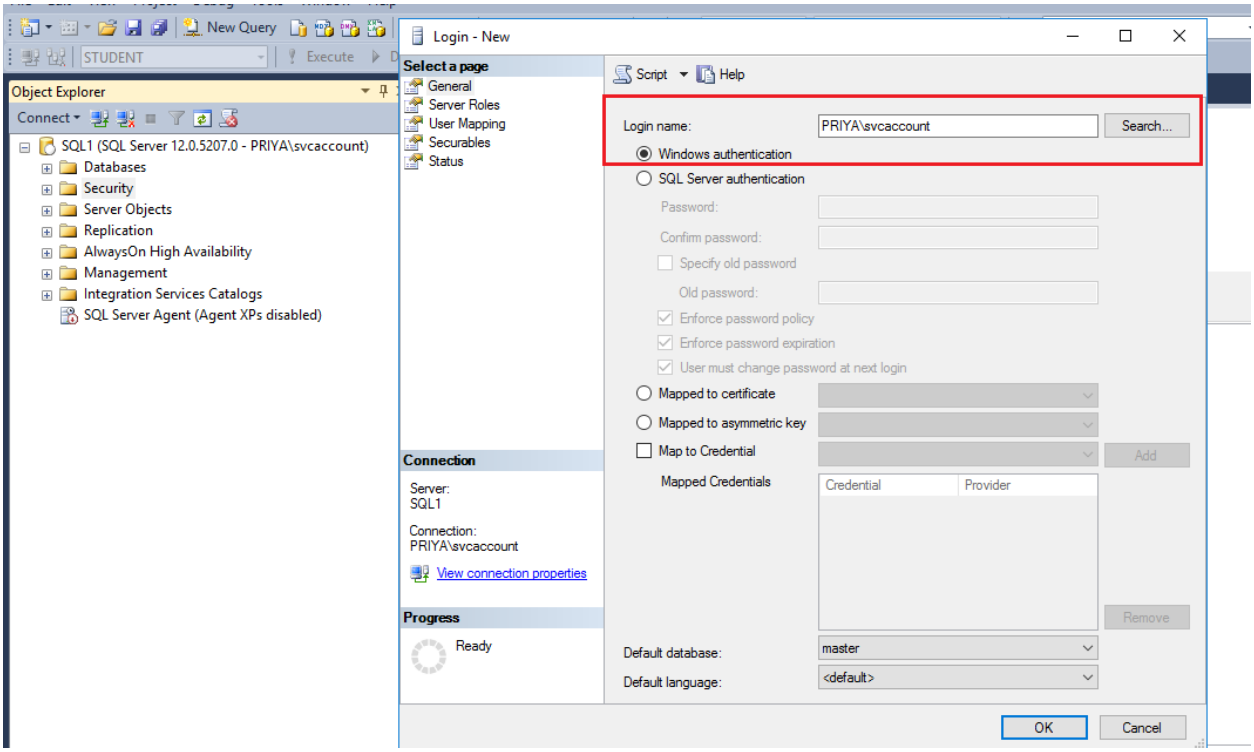


In the following Fig. We will create the Service account as a windows user in **Control Panel > User Account > Click on Manage User Accounts**.

Enter a **Service account name** and **domain name** and Click **Next**, Select **Administrator** and Finish

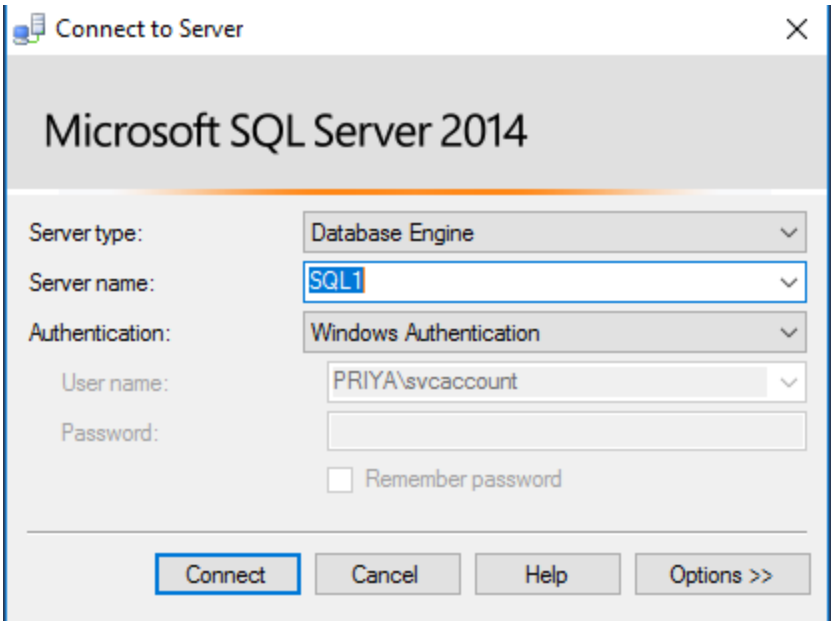


Next, we will create a windows authentication login for the service account as shown in the following fig.



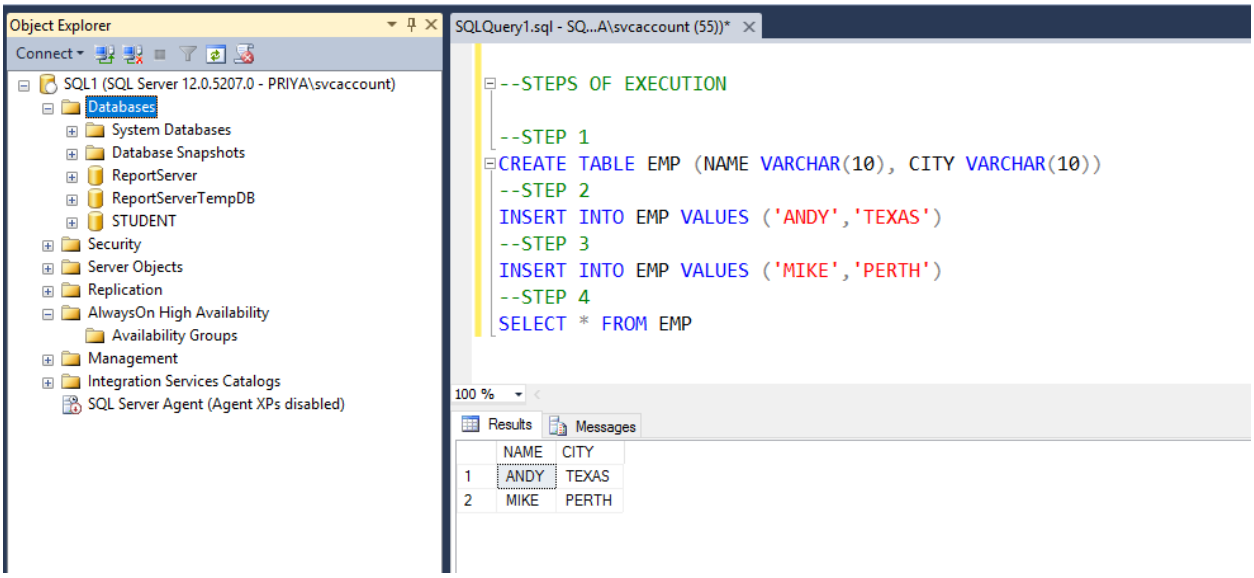
Take remote desktop connection of SQL1 with the service account windows user, connect to the SQL Server database engine instance SQL1 with default windows authentication login as shown in the following fig.

Enable AlwaysOn Feature



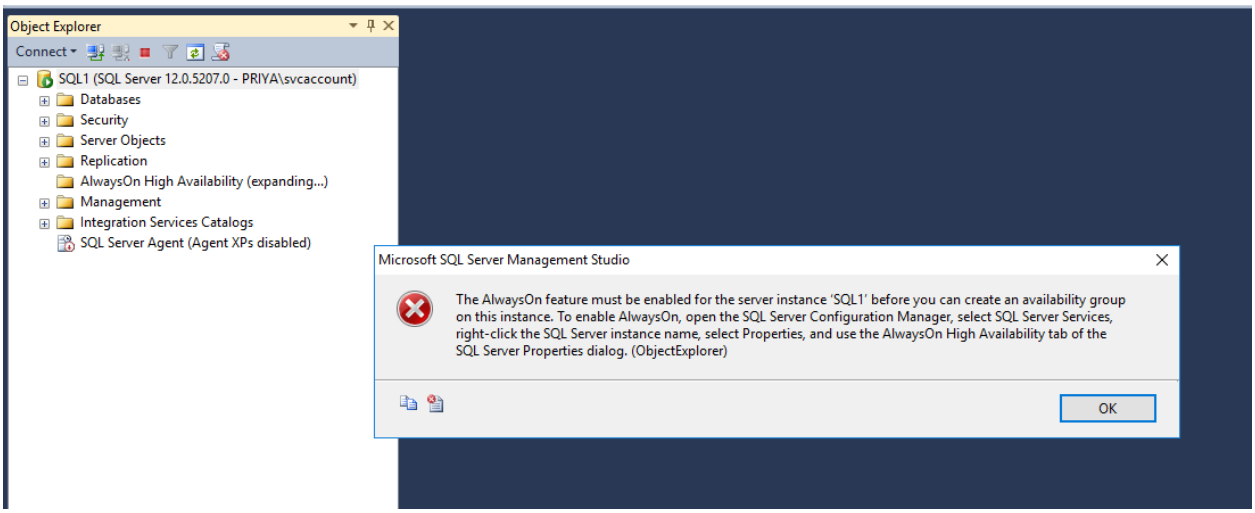
Once SQL Server instance is connected, we need to decide database which will take part in SQL Server AlwaysOn High Availability.

In this demonstration, we will create a new database **STUDENT**, and then we will create a table and add a few records into it as shown in the following fig.



Next, Expand SQL Server AlwaysOn High Availability option, we will get an error while expanding the AlwaysOn option

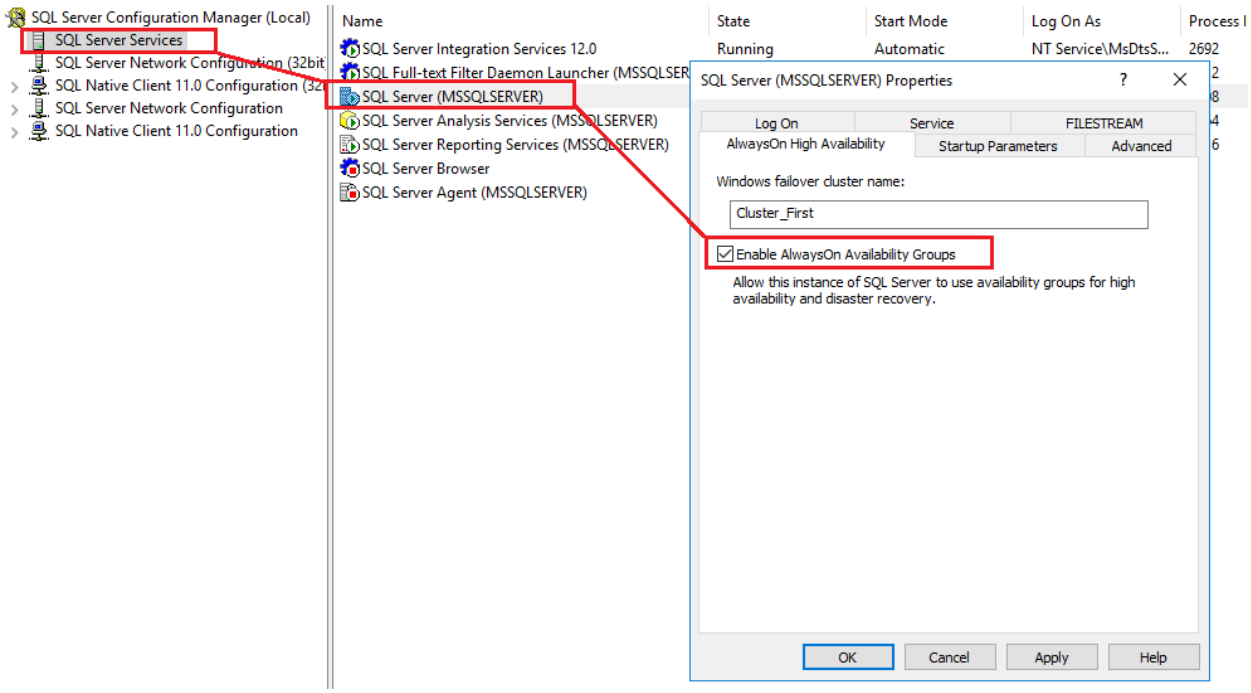
The AlwaysOn feature must be enabled for the server instance 'SQL1' before you can create an availability group on this instance. To enable AlwaysOn, open the SQL Server Configuration Manager, select SQL Server Services, right-click the SQL Server instance name, select Properties, and use the AlwaysOn High Availability tab of the SQL Server Properties dialog. (ObjectExplorer)



To enable AlwaysOn feature, **Open SQL Server Configuration Manager**, **Right-click SQL Server instance** and **go to properties**, Select **AlwaysOn High**

Availability tab section and tick checkbox for **Enable AlwaysOn AvailabilityGroups**

Note: – *We also need to make sure nodes where we will enable AlwaysOn feature is part of Windows Failover Cluster.*

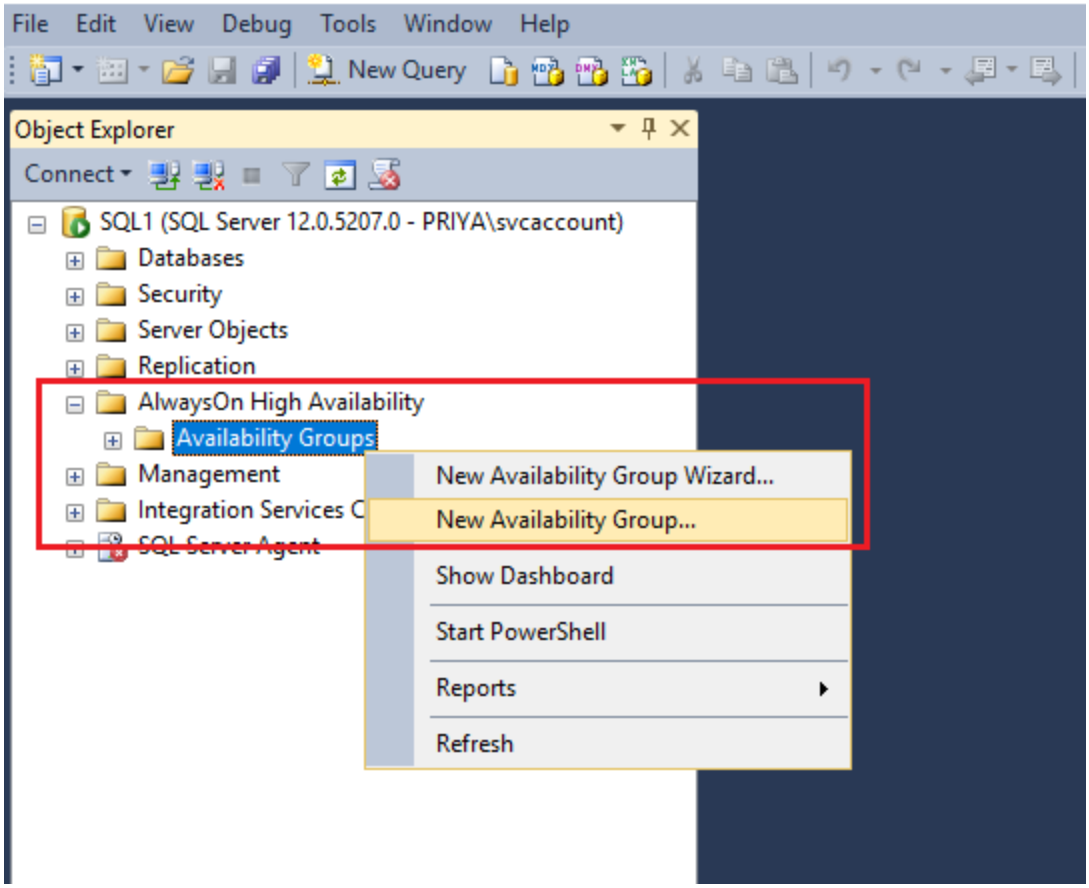


Once we enabled, Apply and Click **OK**. Restart the SQL Server services.

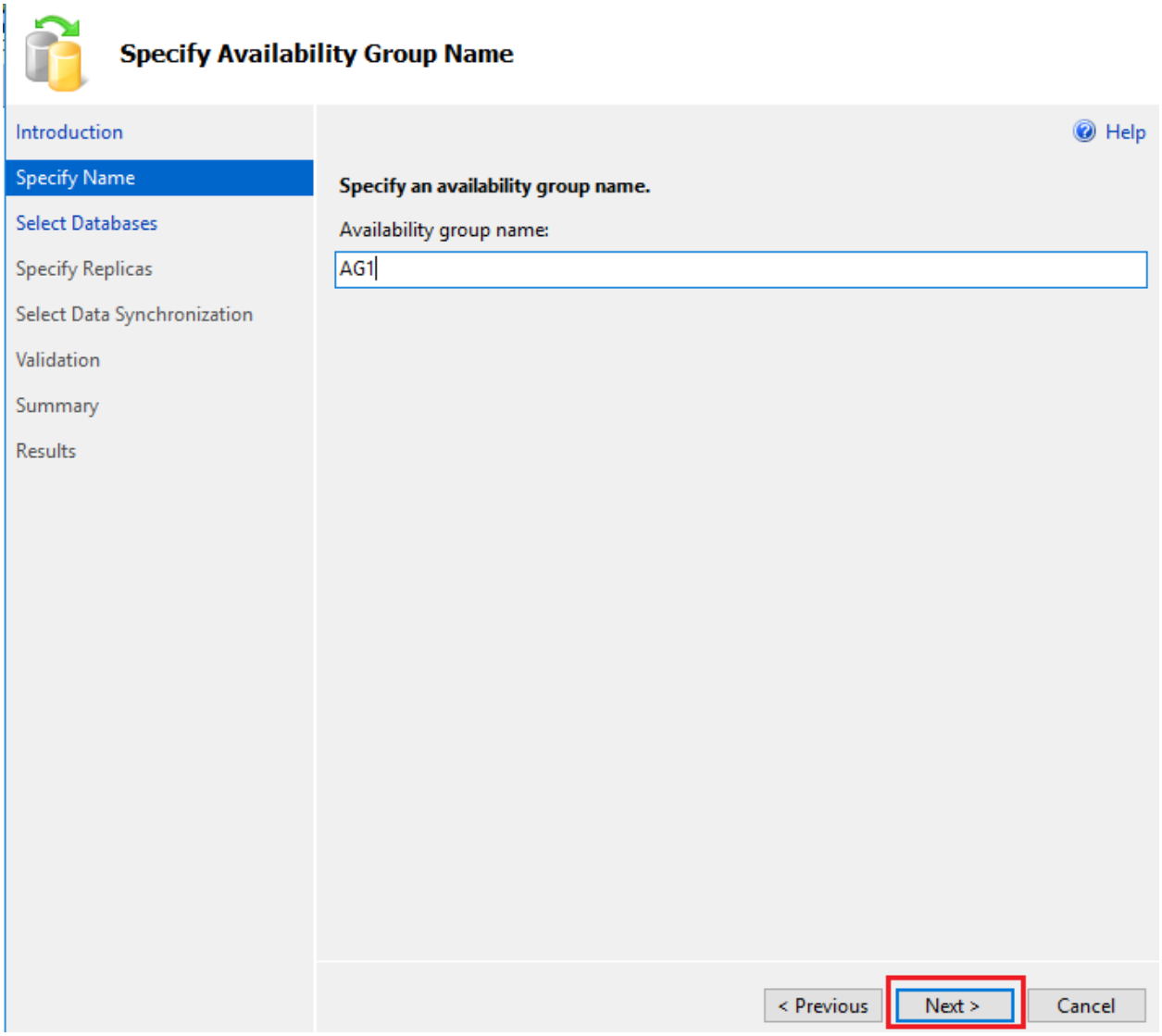
As shown in the following fig. we will able to use SQL Server AlwaysOn High Availability feature. Right-click **Availability Groups** and Click **New Availability Group**.

Configuring Availability Group

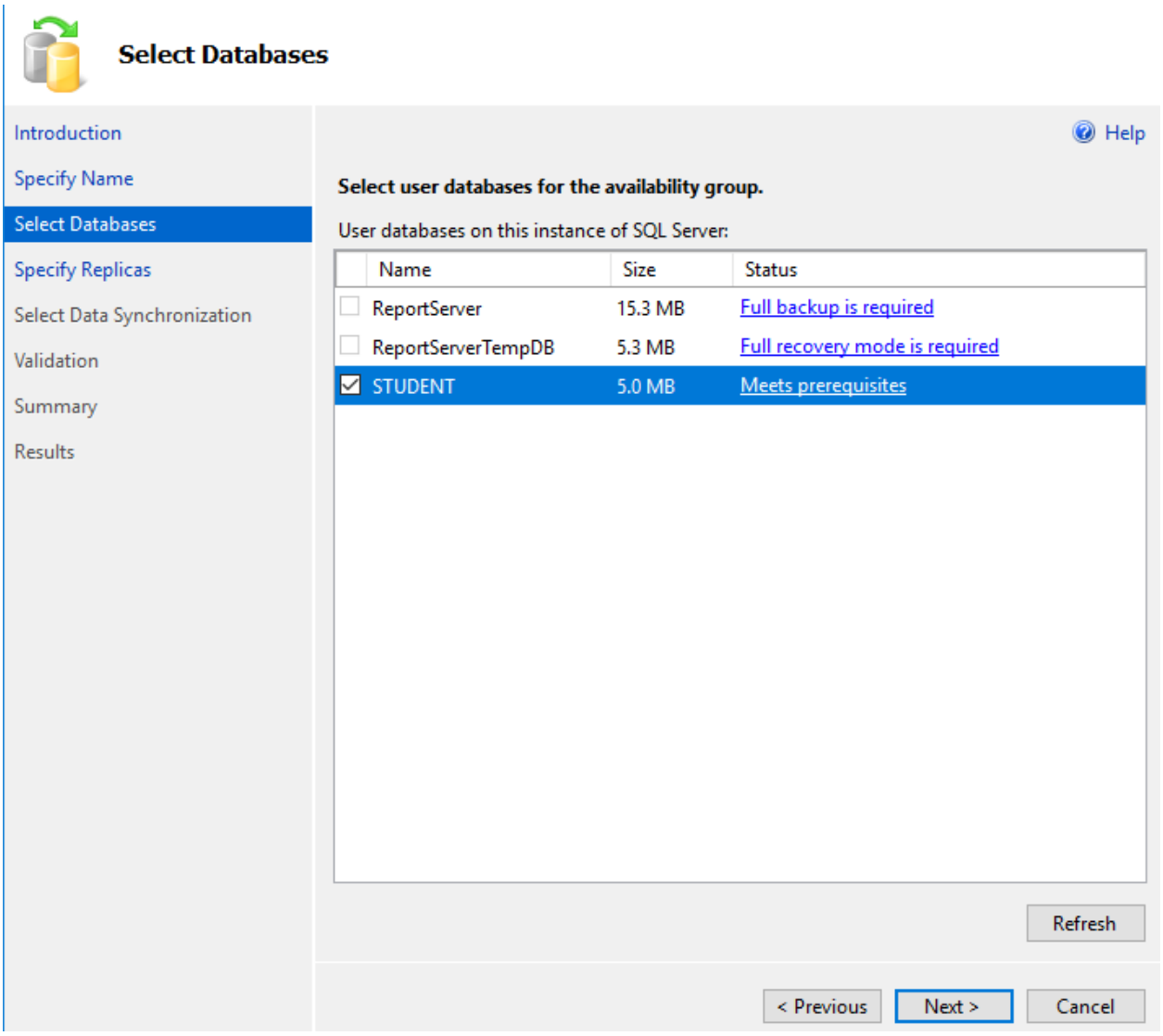
An AlwaysOn Availability Group contains a set of user databases; we need to select user databases which will be part of the Availability group. These databases called as an Availability database.



Specify **Availability Group name** and click **Next**



As shown in fig. we will use database **STUDENT**. To proceed further, we need to perform one essential pre-requisite step, a Full backup of the database. So, take a Full backup of STUDENT and click **Next**



Next, **Specify Replica** step has four sections. **Replica, Endpoint, Backup Preferences, and Listener**

Replica– Replica is a server. There are one primary replica and multiple replicas. In SQL server 2012, It supports up to 4 secondary replicas, while in SQL Server 2014, it supports up to 8 replicas.

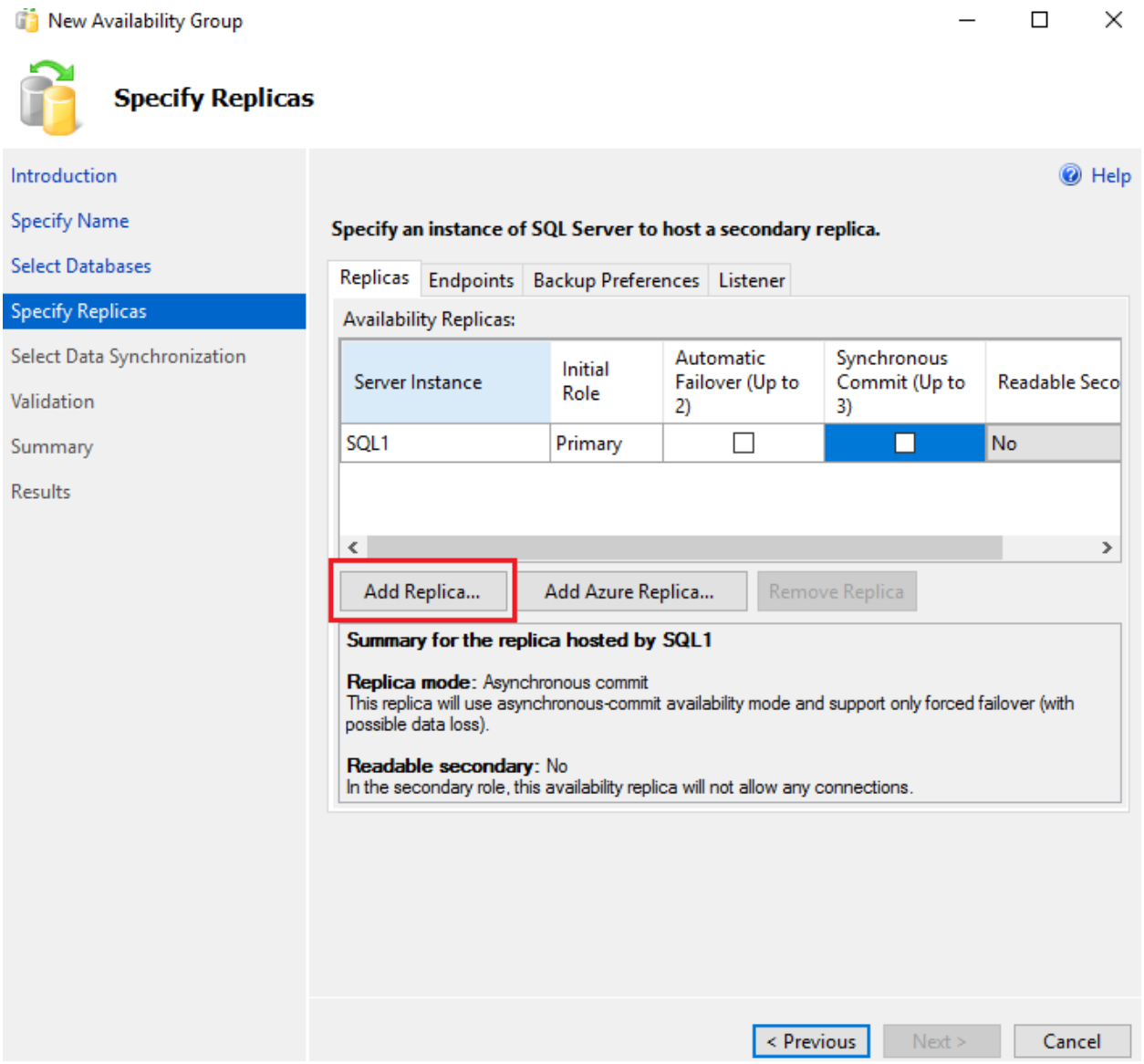
The primary replica is primary source server or production server. A secondary replica is a server which maintain a backup copy of the primary server availability database. On the Primary replica, it allows to perform Read and write operations while on the secondary replica only read operations.

In this case, SQL1 is our primary replica and SQL2 is a secondary replica. So, we will add SQL2 as a secondary replica as shown in the following fig.

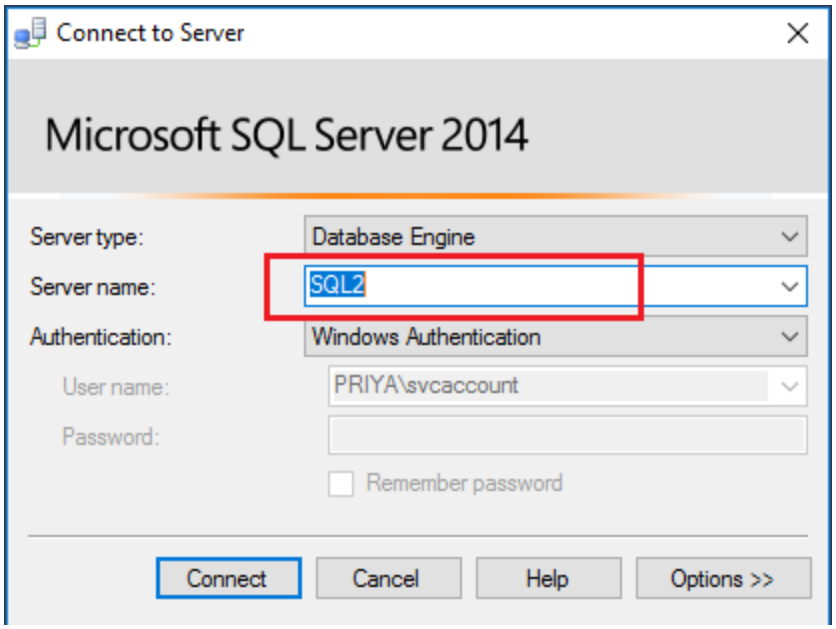
Initial Role – It specifies the role of replica whether Primary or secondary**.**

Automatic Failover – It failovers functional role from primary replica to secondary replica by an automated method with no data loss.

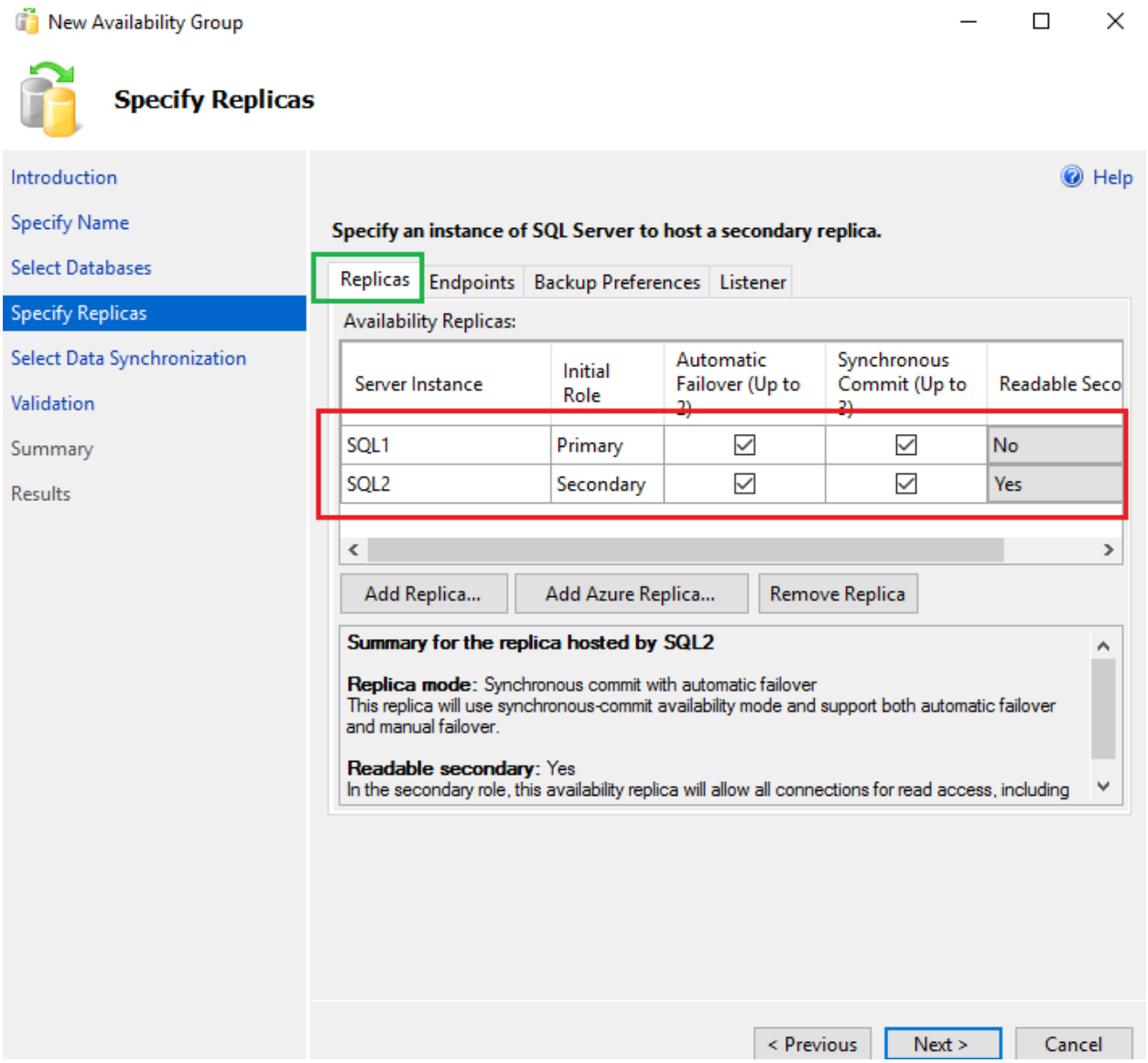
Synchronous Commit – In the Synchronous-commit mode, failover can happen by either Automatic or manual way with no data loss. So, in this case, we will use synchronous commit with automatic failover.



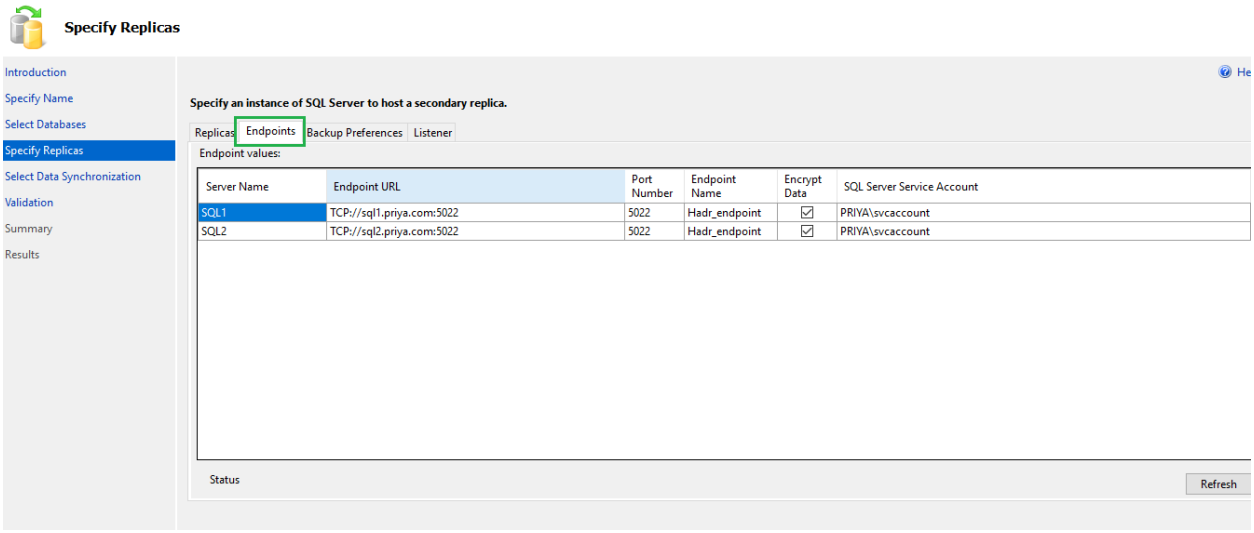
Once clicked **Add Replica**, connect to the server, Select SQL2 and Connect



Once we add secondary server SQL2 as a secondary replica, the primary role of SQL2 is secondary.



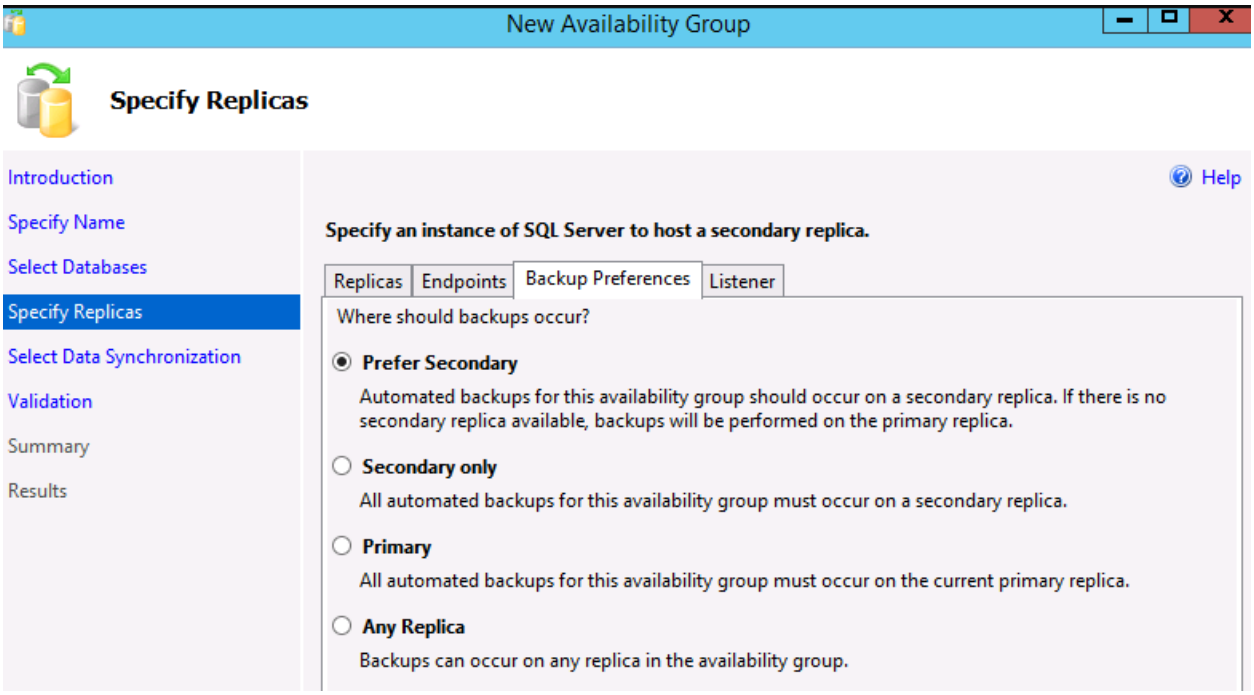
In the Endpoint section, it will show configured endpoint URL, port, endpoint name, and the service account name for SQL1 and SQL2 nodes.



Backup Preferences – Backup preferences indicate the backup location of the **availability group.** It provides four backup options.

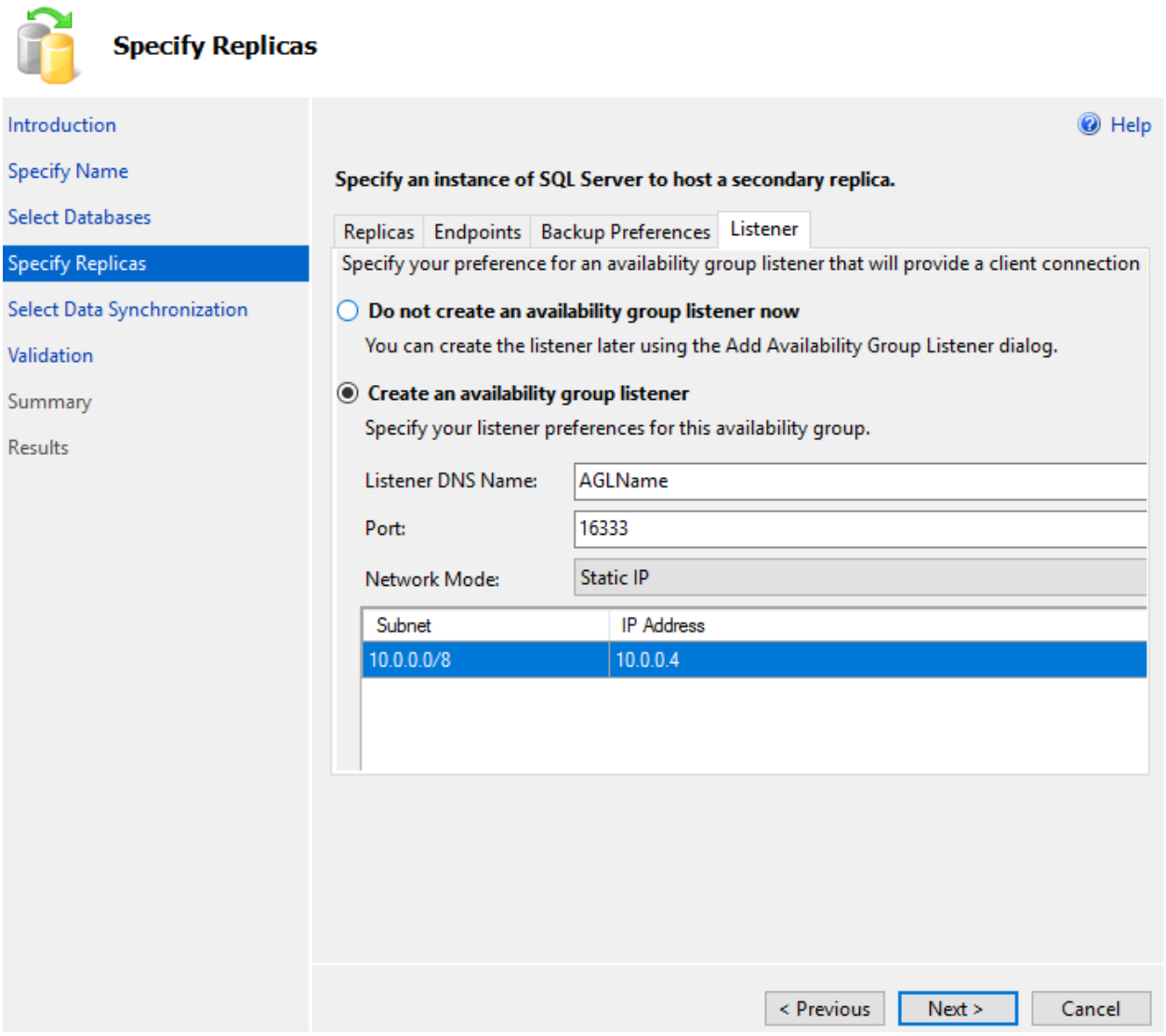
- **Prefer Secondary** – Automatic backup will occur on a secondary replica
- **Secondary only** – Automatic backup must occur on a secondary replica
- **Primary** – Automatic backup must occur on a primary replica
- **Any Replica** – Backup will occur on any replica

On the secondary replica, we can only take a log backup, and Copy-only backup, Differential, and Full back up are not allowed on a secondary replica. While on the primary replica, we can perform Full, log, and Differential backups.



Listener– Listener is a server through which nodes get to communicate with each other. The Listener has all information about availability group. We need to select **Create an Availability group listener** option and specify Name, Port and static IP as shown in the following fig.

usual port is 1433

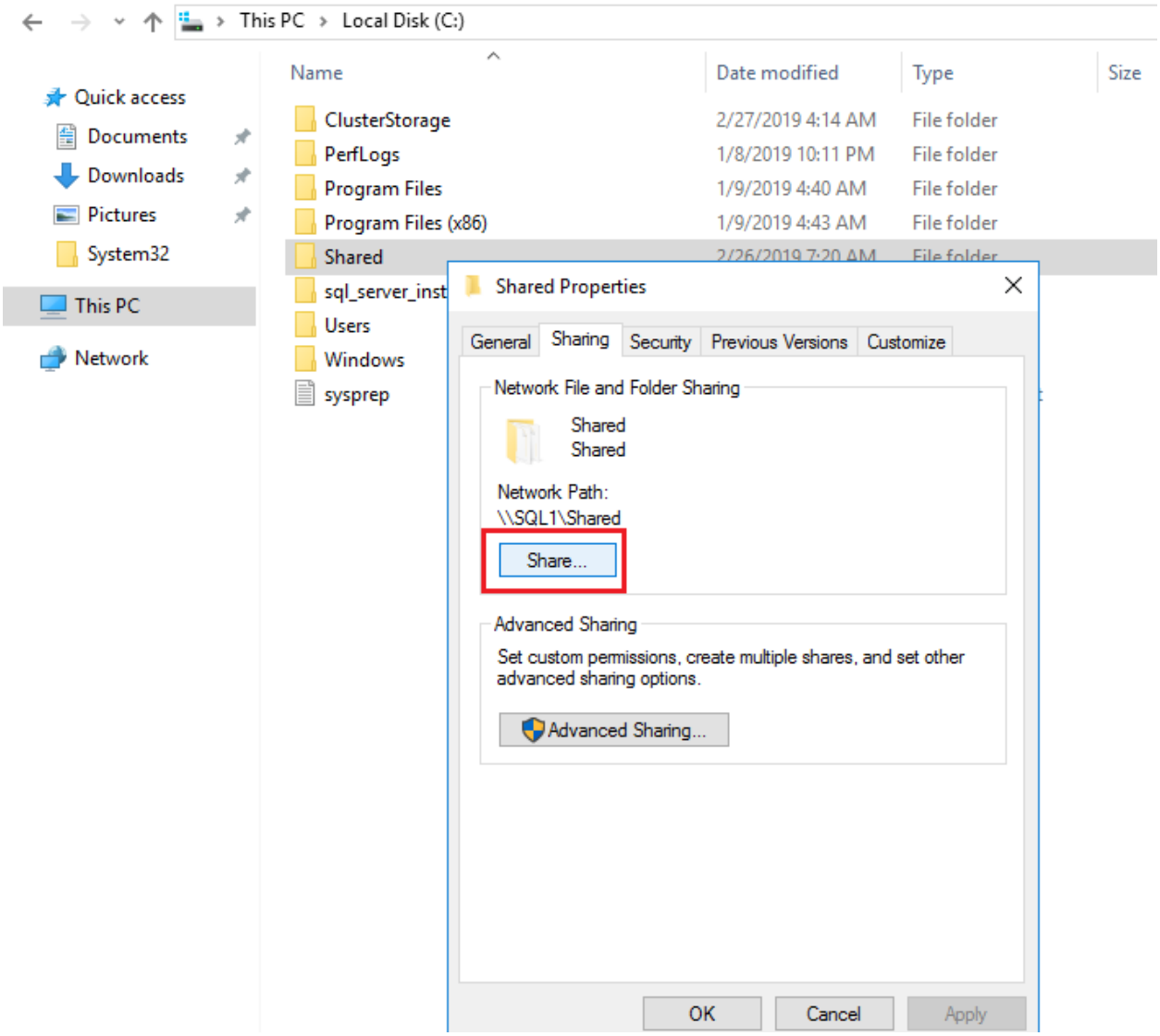


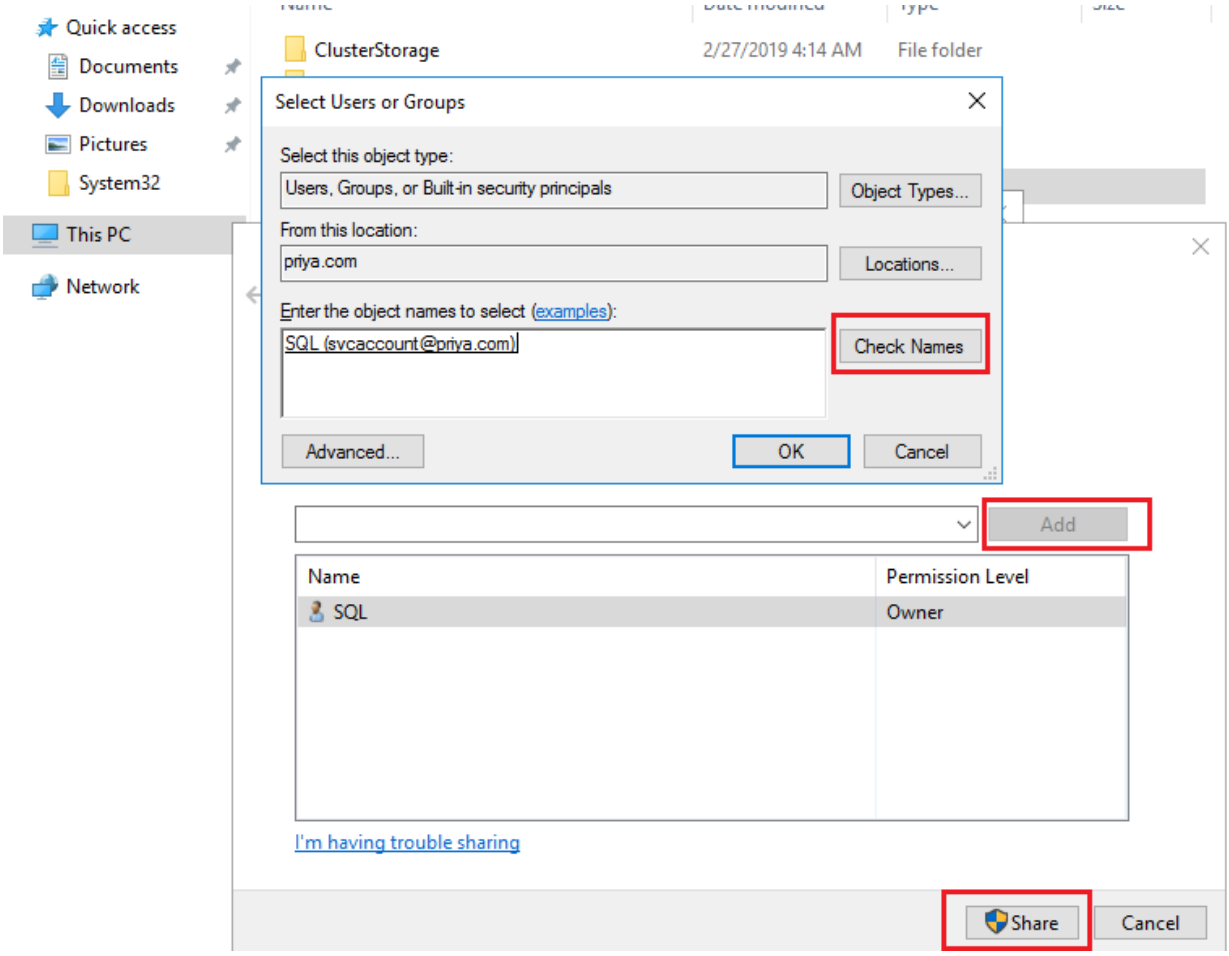
In **Data Synchronization step**, Select **Full** where it performs Full and Log backup and puts it into shared location path.

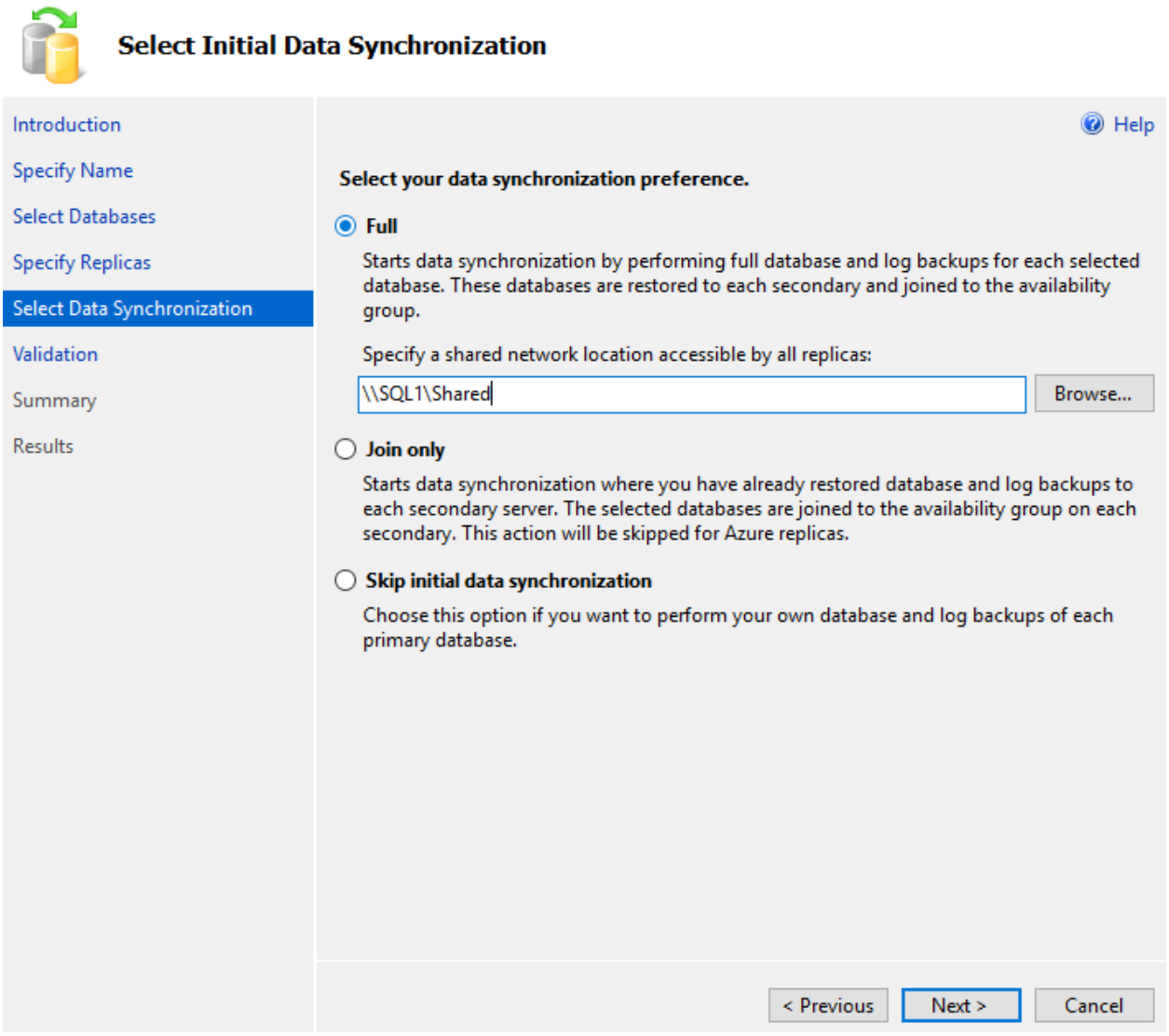
Create a Shared location

Connect to node SQL1 and choose desired drive location and folder, **Right-click folder** and **Go to Properties**, **Select Security** and click **Share**. It will show a pop-up window to add a user to whom, share permission need to grant.

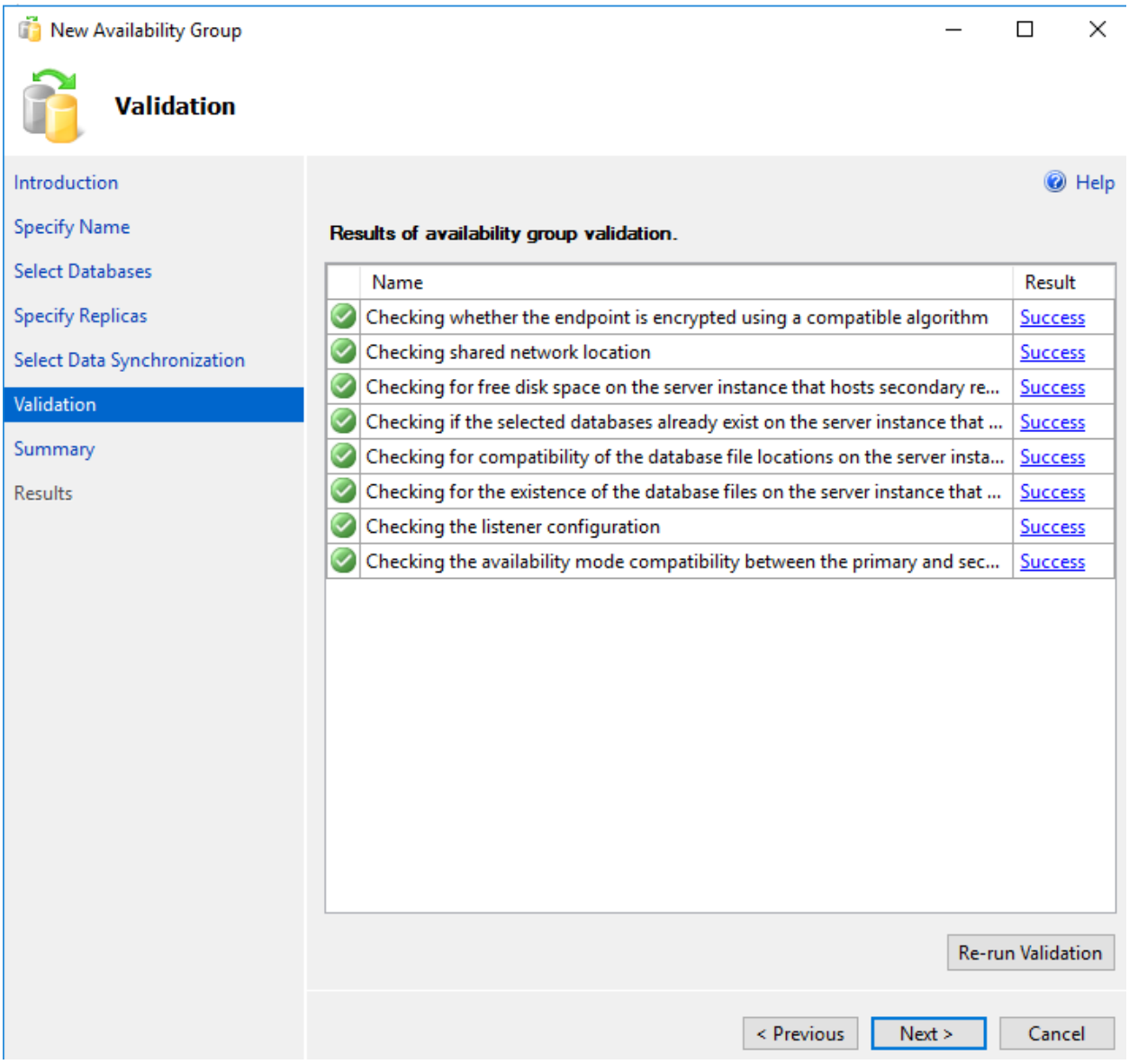
Provide the service account name and Click **Add**; it will show pop-up window to check names, once service account name found in the domain name, click **Share**.


This PC
Local Disk (C:)
Quick access
Documents
Downloads
Pictures
System32
This PC
Network
Name
Date modified
Type
Size
ClusterStorage
2/27/2019 4:14 AM
File folder
PerfLogs
1/8/2019 10:11 PM
File folder
Program Files
1/9/2019 4:40 AM
File folder
Program Files (x86)
1/9/2019 4:43 AM
File folder
Shared
File folder
sql_server_inst
Users
Windows
sysprep
Shared Properties
General
Sharing
Security
Previous Versions
Customize
Network File and Folder Sharing
Shared
Shared
Network Path:
\\SQL1\Shared
Share...
Advanced Sharing
Set custom permissions, create multiple shares, and set other advanced sharing options.
Advanced Sharing...
OK
Cancel
Apply

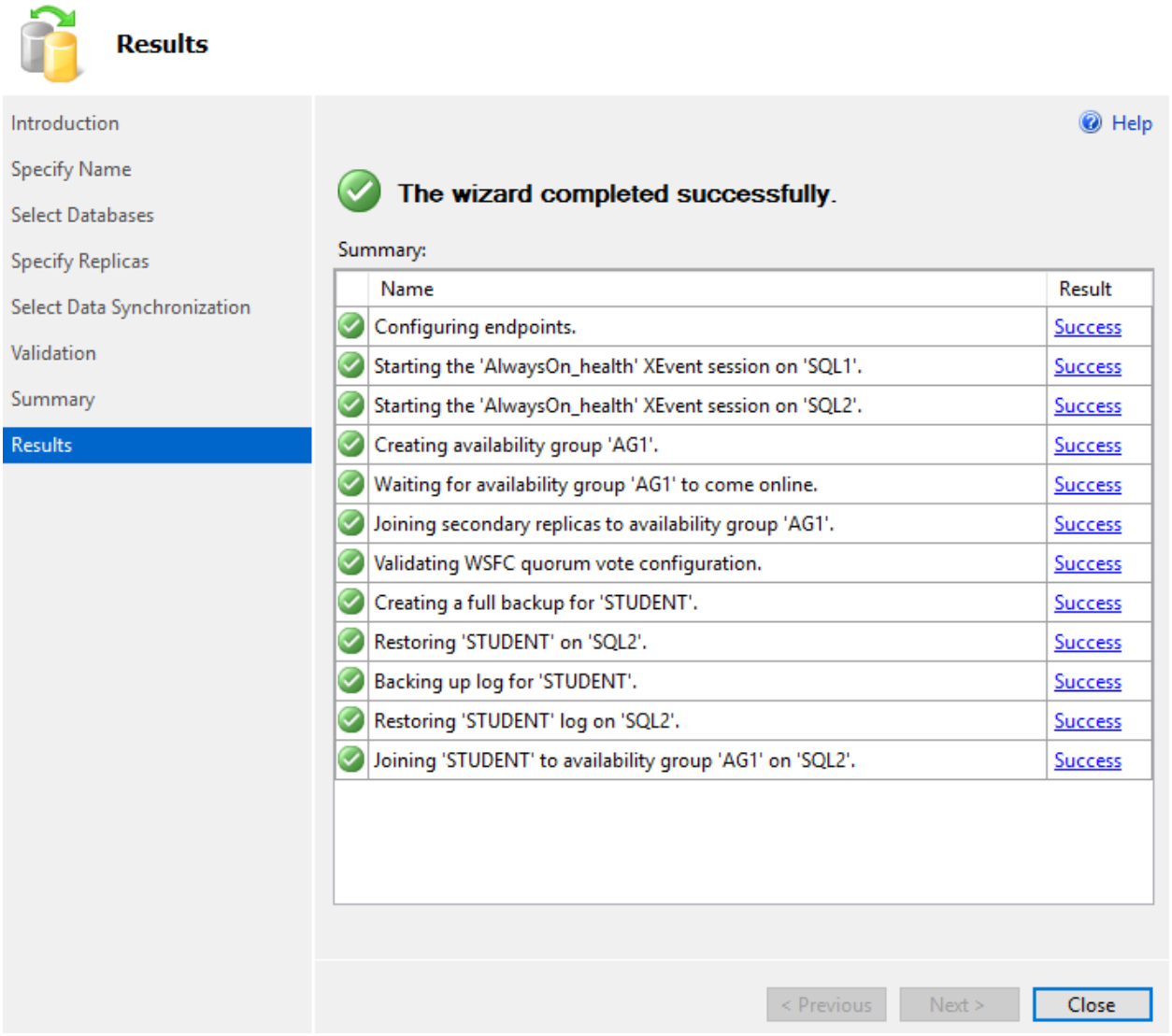

Quick access
Documents
Downloads
Pictures
System32
This PC
Network
ClusterStorage
2/27/2019 4:14 AM
File folder
Select Users or Groups
Select this object type:
Users, Groups, or Built-in security principals
Object Types...
From this location:
priya.com
Locations...
Enter the object names to select (examples):
SQL (svcaccount@priya.com)
Check Names
Advanced...
OK
Cancel
Add
Name
Permission Level
SQL
Owner
I'm having trouble sharing
Share
Cancel



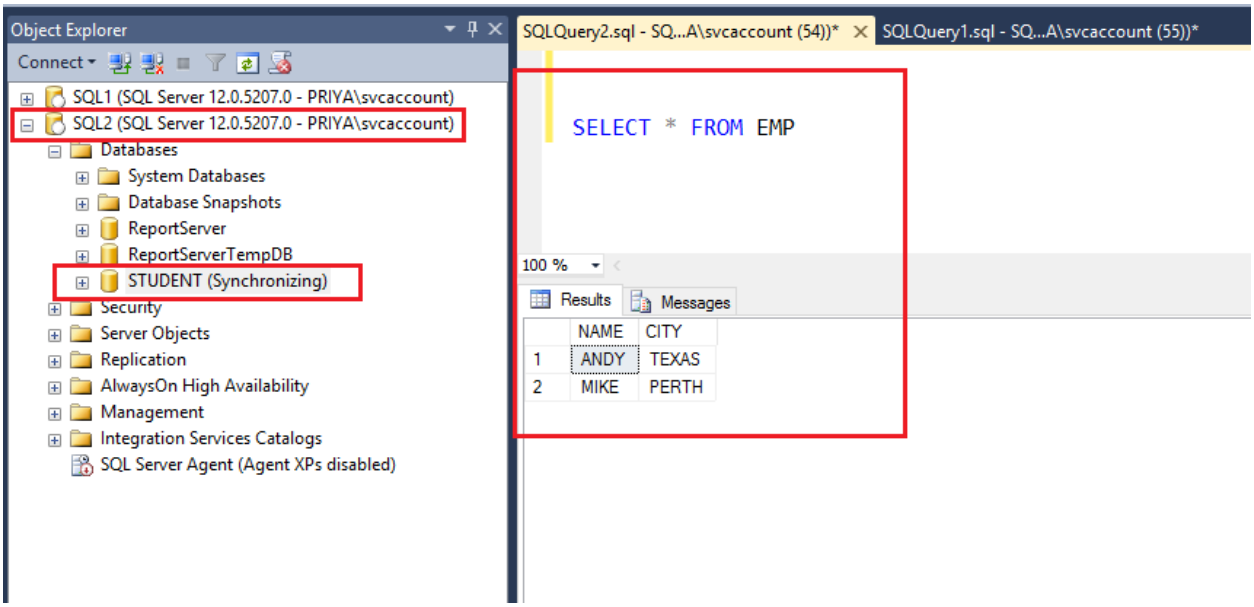
In **the Validation step**, SQL Server performs validations against any required actions, and It will provide warning or error messages for missing required steps.



It performed all validation steps successfully, and wizard operation completed as shown in the following fig.



After successful completion, we will verify the database at secondary replica SQL2. As we can observe, Database STUDENT is in Synchronizing state, and it is in sync with the primary replica, and we can read data successfully.


Object Explorer
Connect
SQL1 (SQL Server 12.0.5207.0 - PRIYA\svcaccount)
SQL2 (SQL Server 12.0.5207.0 - PRIYA\svcaccount)
Databases
System Databases
Database Snapshots
ReportServer
ReportServerTempDB
STUDENT (Synchronizing)
Security
Server Objects
Replication
AlwaysOn High Availability
Management
Integration Services Catalogs
SQL Server Agent (Agent XPs disabled)
SQLQuery2.sql - SQ...A\svcaccount (54))*
SQLQuery1.sql - SQ...A\svcaccount (55))*
SELECT * FROM EMP
100 %
Results
Messages
NAME
CITY
1
ANDY
TEXAS
2
MIKE
PERTH