

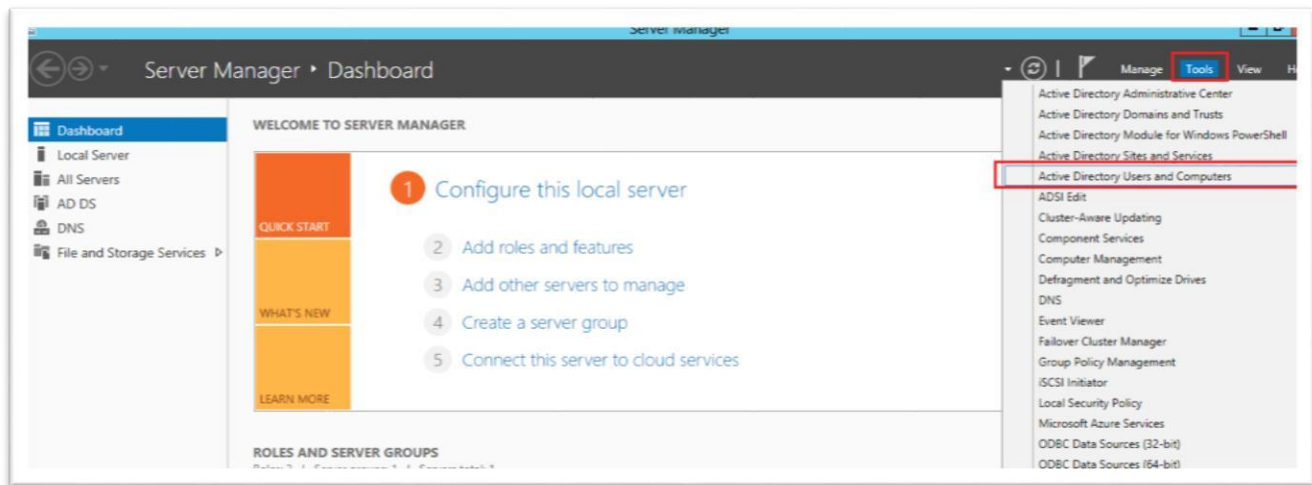
Configuring a SQL Server AlwaysOn High Availability Group

In this article, we will learn the step by step configuration of SQL Server AlwaysOn High availability Group for two nodes. Once nodes are added to the cluster group, we will be able to use the AlwaysOn feature in the SQL server.

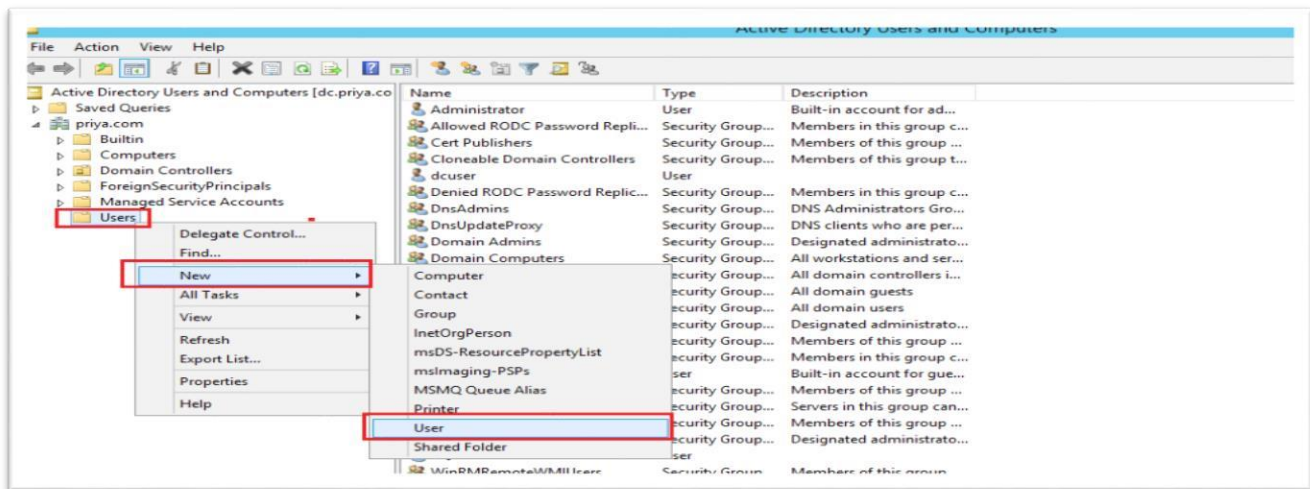
In the following steps, we will create a Service account to connect SQL server instances. The Service account will be used to connect the SQL instance from both the node servers.

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.



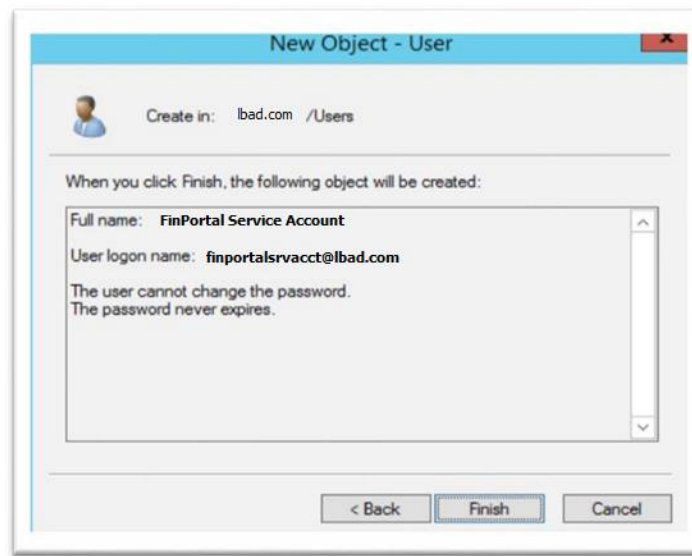
In this step, we will create a Service account username, Specify the service account login name and Click Next

The 'New Object - User' dialog box is shown. The 'Create in' field is set to 'priya.com/Users'. The 'First name' field contains 'FinPortal', and the 'Last name' field contains 'Service Account'. The 'User login name' field contains 'finportalsrvacct' and the domain dropdown is set to '@lbad.com'. The 'User login name (pre-Windows 2000)' field contains 'LBAD\finportalsrvacct'. The 'Next >' button is highlighted.

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.

The 'New Object - User' dialog box is shown, displaying the password and options section. The 'Password' and 'Confirm password' fields are filled with dots. The 'User must change password at next logon' checkbox is unchecked. The 'User cannot change password' checkbox is checked. The 'Password never expires' checkbox is checked. The 'Account is disabled' checkbox is unchecked. The 'Next >' button is highlighted.

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



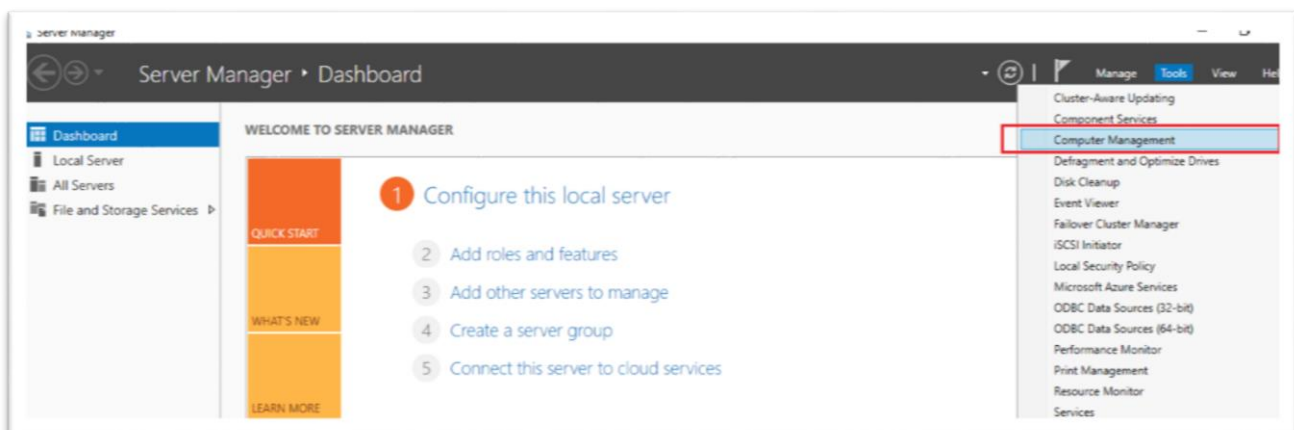
Configure Administrator permission, windows user and login for the service account

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.

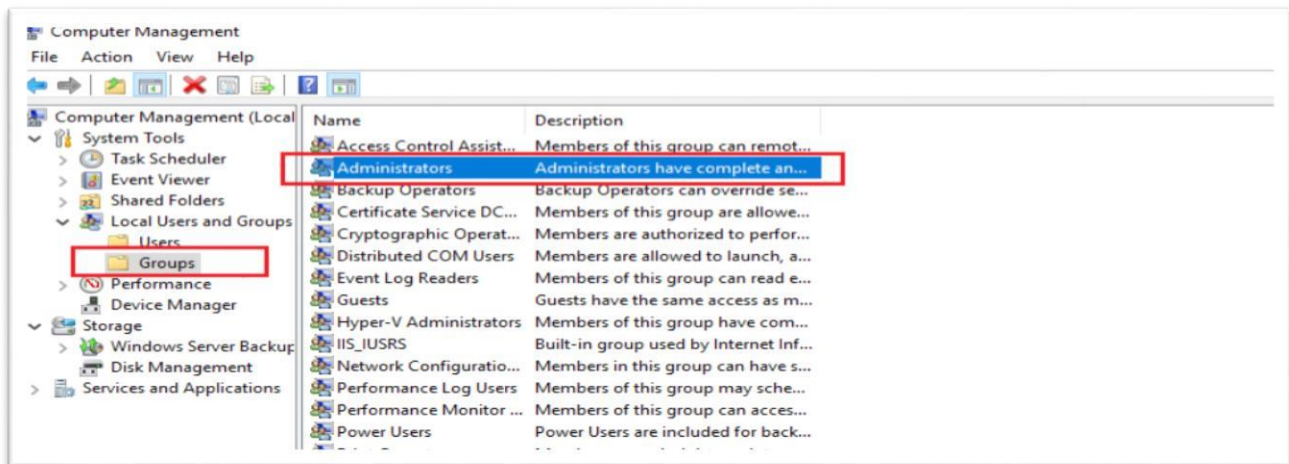
Connect to both node servers **FINPORTALDB** and **FINPORTALDB2** and perform below steps

- 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**

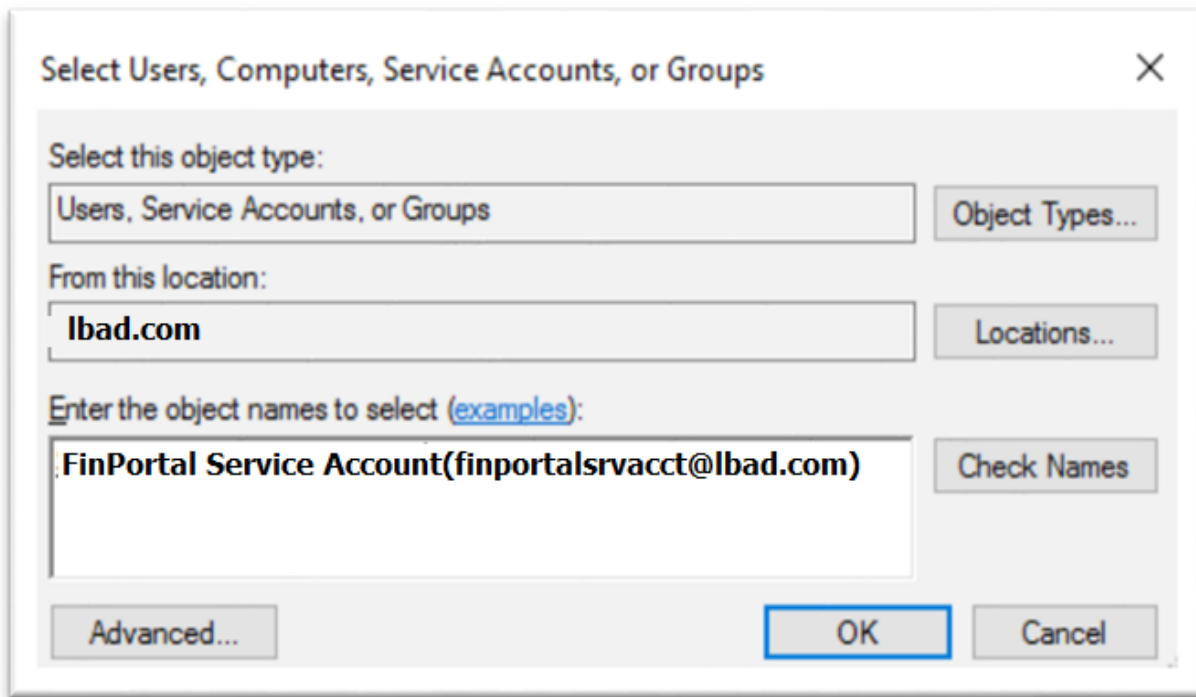


In the **Computer management** window, select **Groups** and Right-Click **Administrators** and Go to **Properties**

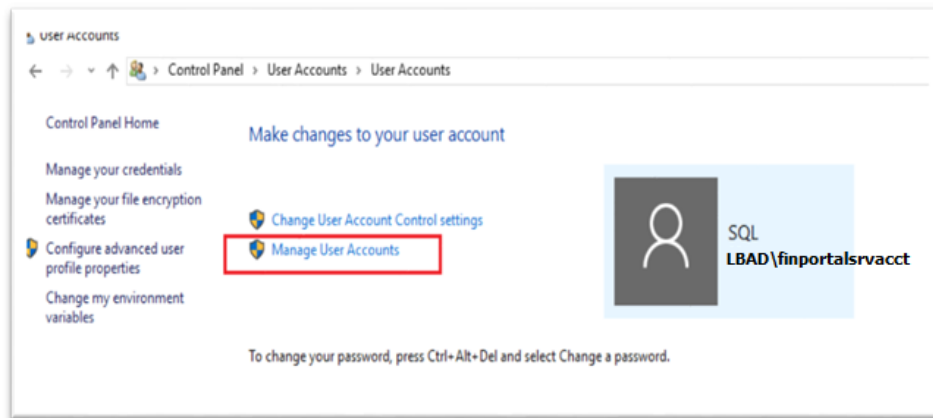


Add User, Enter the Service account login name and Check names and Click **OK**

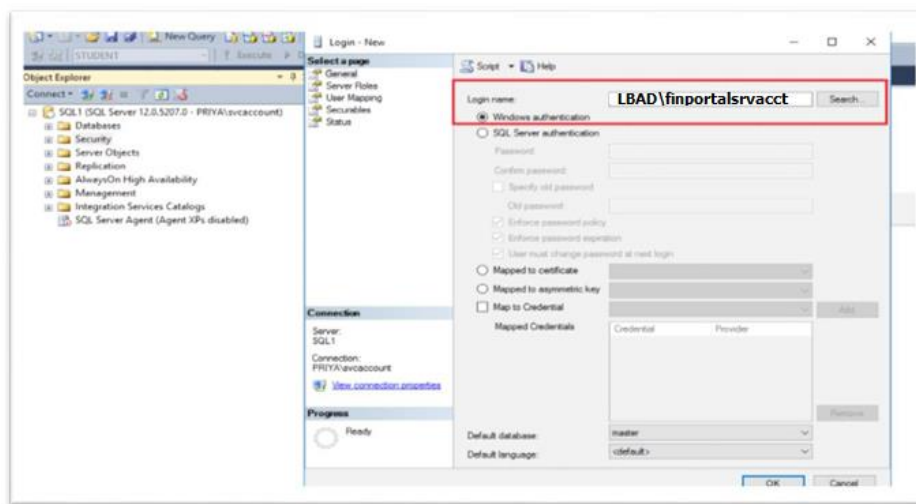
Note – Once clicked on the Check Names, it may ask to enter username and password pop-up window. In the pop-up window, we need to register a domain controller username and password and make sure the location should select as a directory.



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.



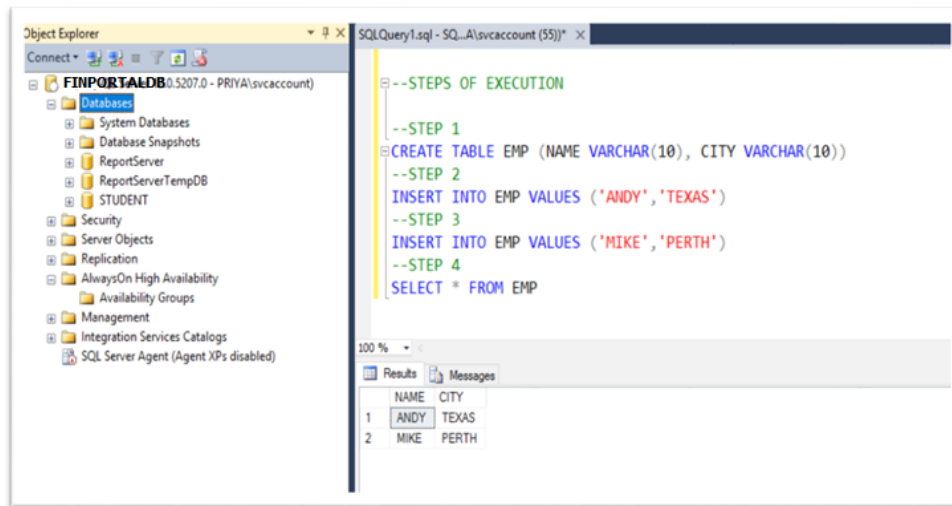
Take remote desktop connection of SQL1 with the service account windows user, connect to the SQL Server database engine instance SQL1 with default windows authentication login as shown in the following fig.

Enable AlwaysOn Feature



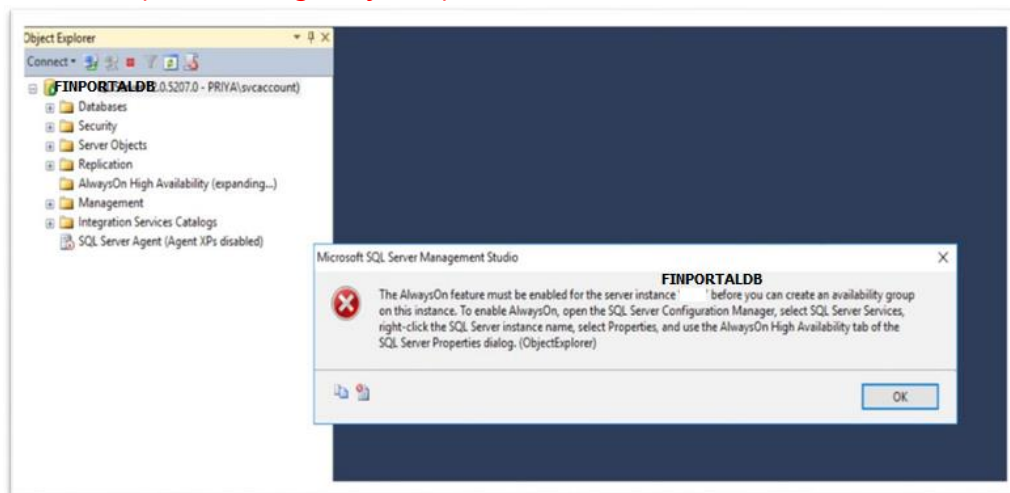
Once SQL Server instance is connected, we need to decide database which will take part in SQL Server AlwaysOn High Availability.

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.



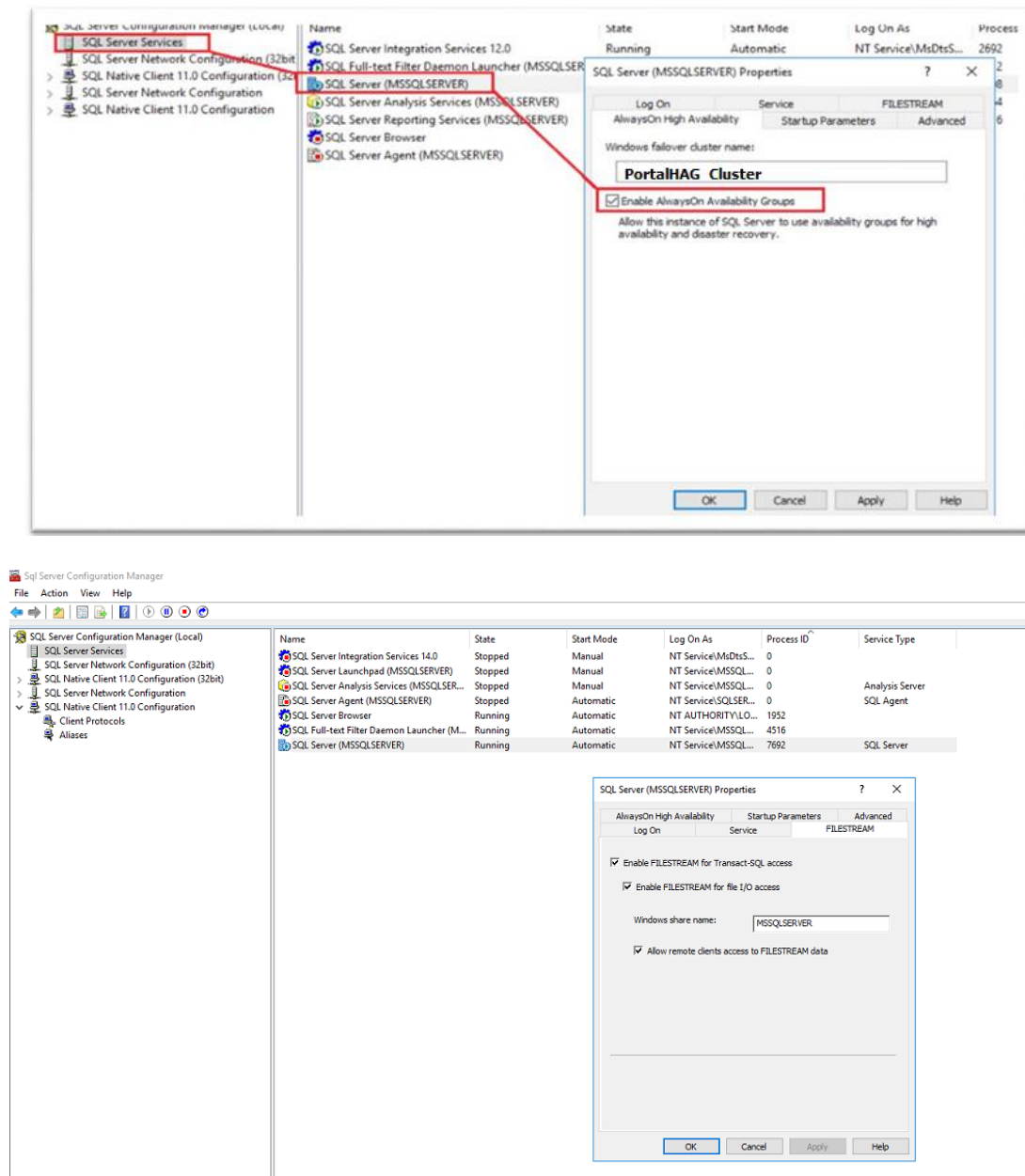
Next, Expand SQL Server AlwaysOn High Availability option, we will get an error while expanding the AlwaysOn option

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.

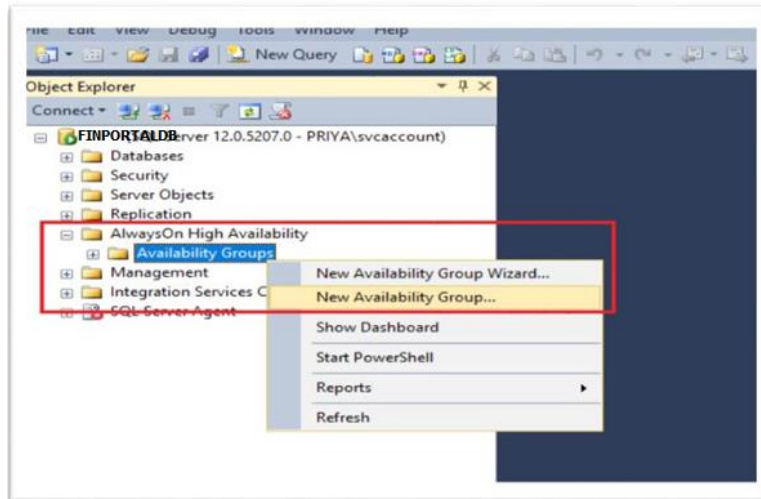


Once we enabled, Apply and Click **OK**. Restart the SQL Server services.

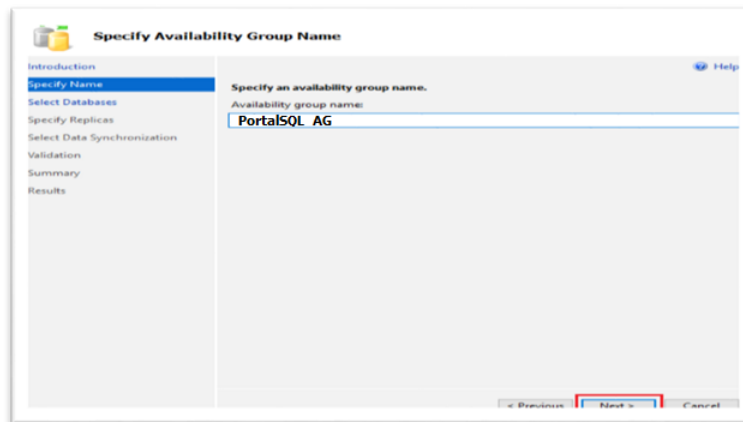
As shown in the following fig. we will be able to use SQL Server AlwaysOn High Availability feature. Right click **Availability Groups** and Click **New Availability Group**.

Configuring Availability Group

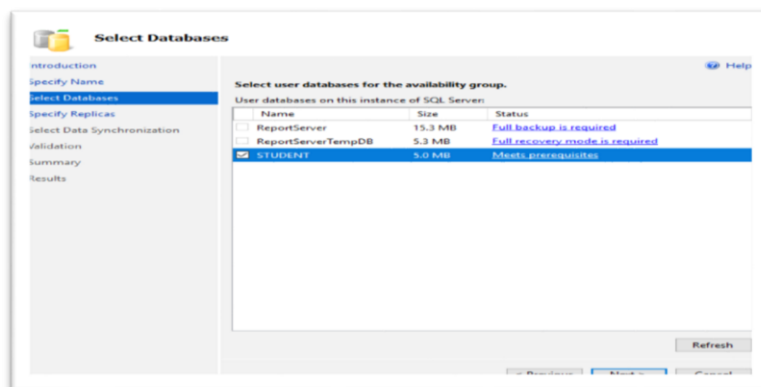
An AlwaysOn Availability Group contains a set of user databases; we need to select user databases which will be part of the Availability group. These databases called as an Availability database.



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 prerequisite 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.

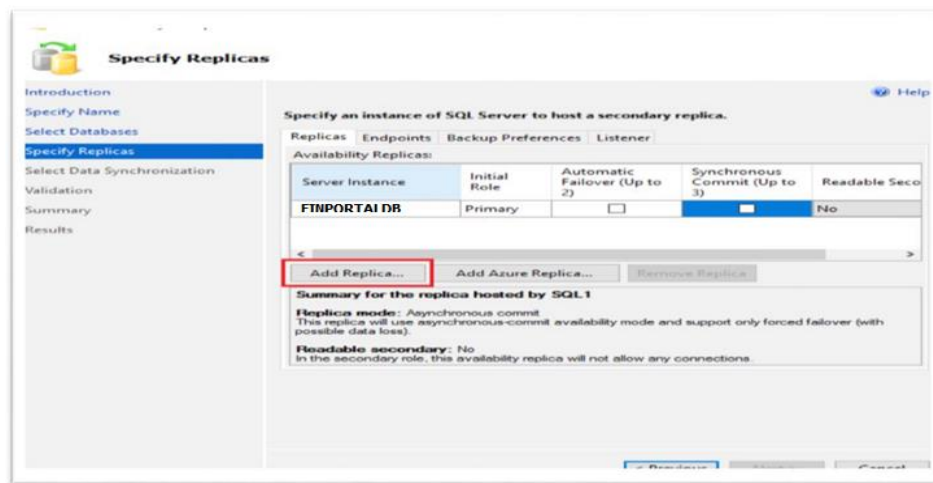
The primary replica is primary source server or production server. A secondary replica is a server which maintain a backup copy of the primary server availability database. On the Primary replica, it allows to perform Read and write operations while on the secondary replica only read operations.

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

Connect to Server

Microsoft SQL Server 2014

Server type: Database Engine

Server name: **FINPORTALDB2**

Authentication: Windows Authentication

User name: PRIYA\svccount

Password:

☐ Remember password

Connect Cancel Help Options >>

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

Specify Replicas

Specify an instance of SQL Server to host a secondary replica.

Replicas Endpoints Backup Preferences Listener

Server Instance	Initial Role	Automatic Failover (Up to 3)	Synchronous Commit (Up to 3)	Readable Secondary
FINPORTALDB	Primary	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	No
FINPORTALDB2	Secondary	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Yes

Add Replica... Add Azure Replica... Remove Replica

Summary for the replica hosted by FINPORTALDB2

Replica mode: Synchronous commit with automatic failover
This replica will use synchronous-commit availability mode and support both automatic failover and manual failover.

Readable secondary: Yes
In the secondary role, this availability replica will allow all connections for read access, including

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

Specify Replicas

Specify an instance of SQL Server to host a secondary replica.

Replicas Endpoints Backup Preferences Listener

Endpoint values:

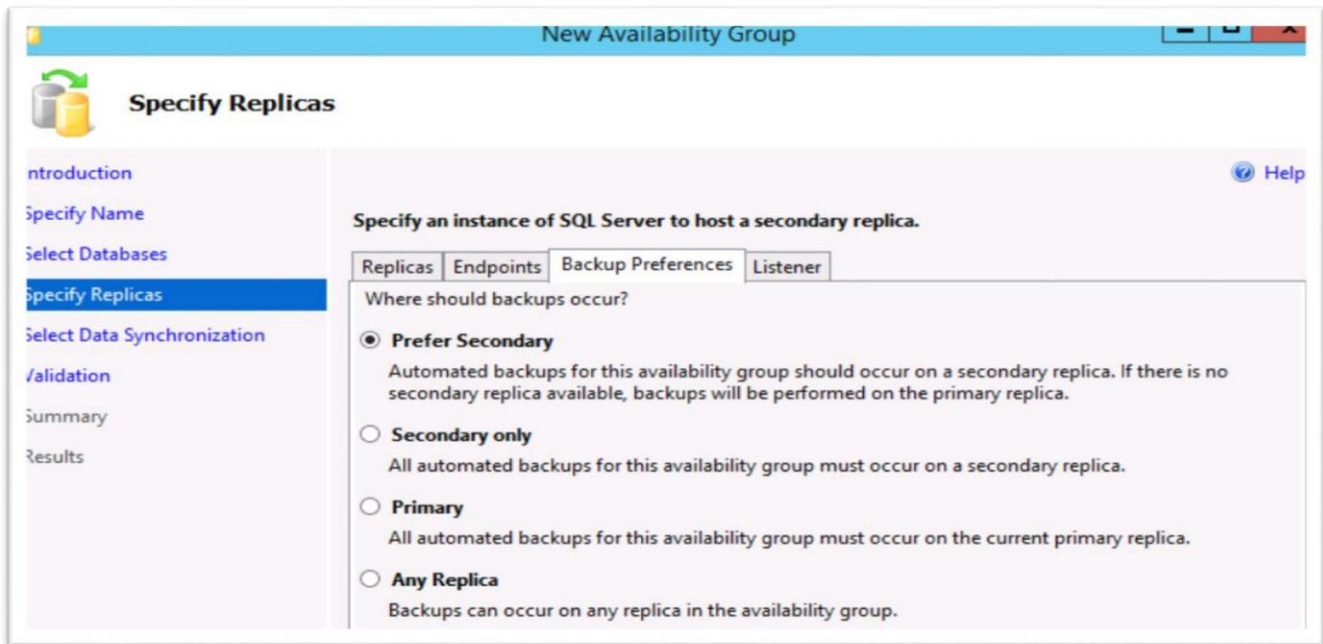
Server Name	Endpoint URL	Port Number	Endpoint Name	Encrypt Data	SQL Server Service Account
FINPORTALDB	TCP://finportaldb.lbad.com:5022	5022	Hadr_endpoint	<input checked="" type="checkbox"/>	LBAD\finportalsvcacct
FINPORTALDB2	TCP://finportaldb2.lbad.com:5022	5022	Hadr_endpoint	<input checked="" type="checkbox"/>	LBAD\finportalsvcacct

Status Refresh

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.



The screenshot shows the 'New Availability Group' wizard in SQL Server Enterprise Manager. The 'Specify Replicas' step is active, showing a list of steps on the left: Introduction, Specify Name, Select Databases, Specify Replicas (selected), Select Data Synchronization, Validation, Summary, and Results. The main area is titled 'Specify an instance of SQL Server to host a secondary replica.' and has tabs for Replicas, Endpoints, Backup Preferences (selected), and Listener. Under the 'Backup Preferences' tab, the question 'Where should backups occur?' is followed by four radio button options: 'Prefer Secondary' (selected), 'Secondary only', 'Primary', and 'Any Replica'. Each option has a descriptive text block below it.

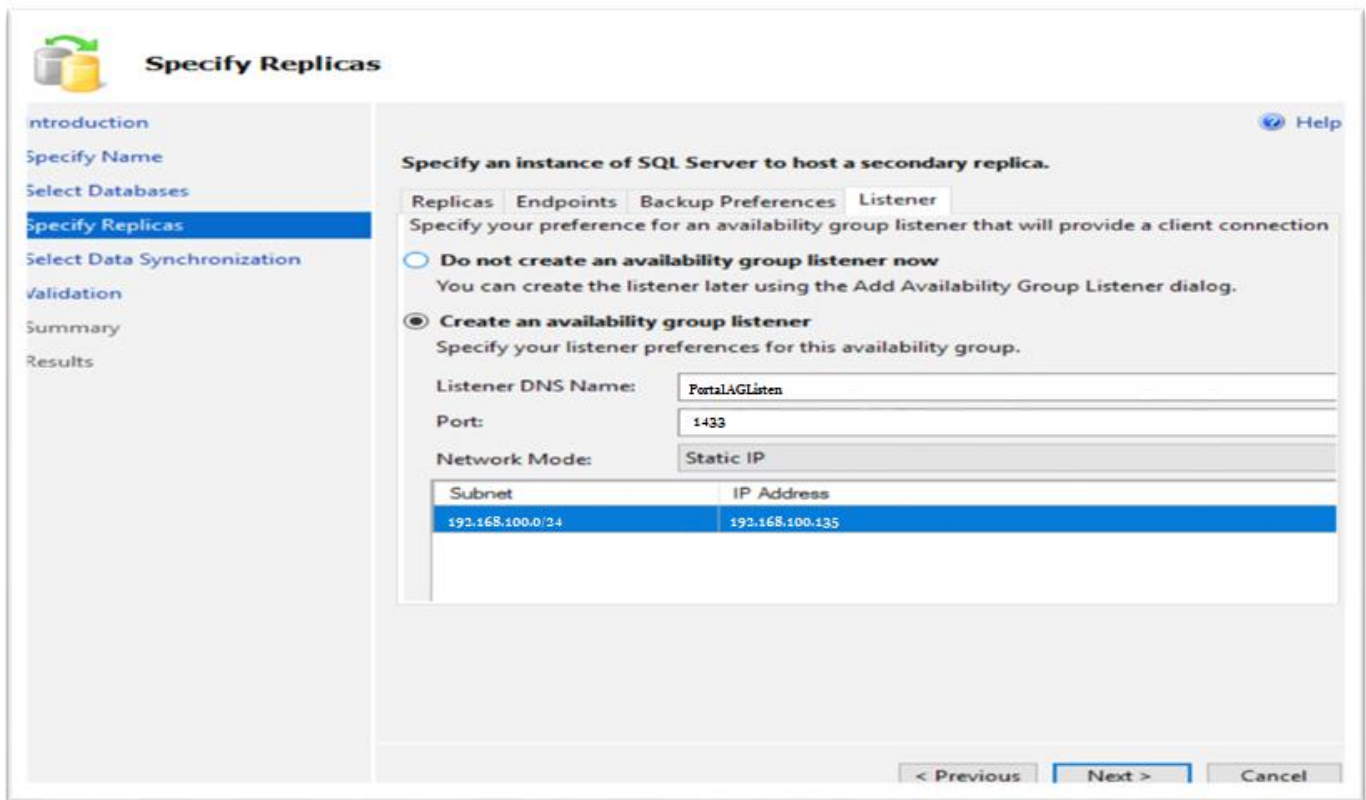
Specify an instance of SQL Server to host a secondary replica.

Replicas | Endpoints | **Backup Preferences** | Listener

Where should backups occur?

- ☒ **Prefer Secondary**
Automated backups for this availability group should occur on a secondary replica. If there is no secondary replica available, backups will be performed on the primary replica.
- ☐ **Secondary only**
All automated backups for this availability group must occur on a secondary replica.
- ☐ **Primary**
All automated backups for this availability group must occur on the current primary replica.
- ☐ **Any Replica**
Backups can occur on any replica in the availability group.

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.



The image shows a 'Specify Replicas' dialog box in SQL Server Enterprise Manager. The left sidebar contains a list of steps: Introduction, Specify Name, Select Databases, Specify Replicas (highlighted), Select Data Synchronization, Validation, Summary, and Results. The main area is titled 'Specify an instance of SQL Server to host a secondary replica.' and has four tabs: Replicas, Endpoints, Backup Preferences, and Listener. The 'Listener' tab is active, showing options to either not create an availability group listener now or to create one. The 'Create an availability group listener' option is selected. Below this, there are input fields for 'Listener DNS Name' (PortalAGListen), 'Port' (1433), and 'Network Mode' (Static IP). At the bottom, there is a table with two columns: 'Subnet' and 'IP Address'. The first row is highlighted in blue and contains the values '192.168.100.0/24' and '192.168.100.135'. At the bottom right of the dialog are buttons for '< Previous', 'Next >', and 'Cancel'.

Specify Replicas

Introduction
Specify Name
Select Databases
Specify Replicas
Select Data Synchronization
Validation
Summary
Results

Specify an instance of SQL Server to host a secondary replica.

Replicas Endpoints Backup Preferences **Listener**

Specify your preference for an availability group listener that will provide a client connection

☐ Do not create an availability group listener now
You can create the listener later using the Add Availability Group Listener dialog.

☒ **Create an availability group listener**
Specify your listener preferences for this availability group.

Listener DNS Name: PortalAGListen

Port: 1433

Network Mode: Static IP

Subnet	IP Address
192.168.100.0/24	192.168.100.135

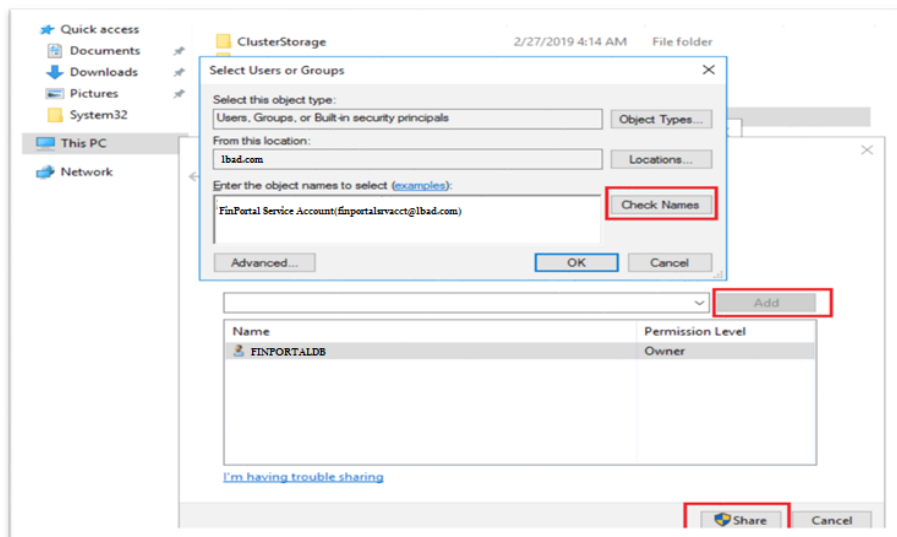
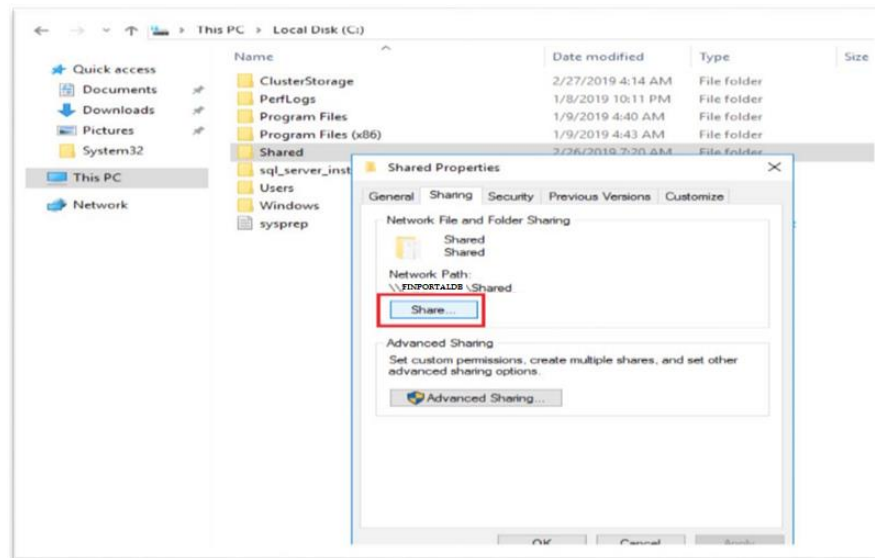
< Previous Next > Cancel

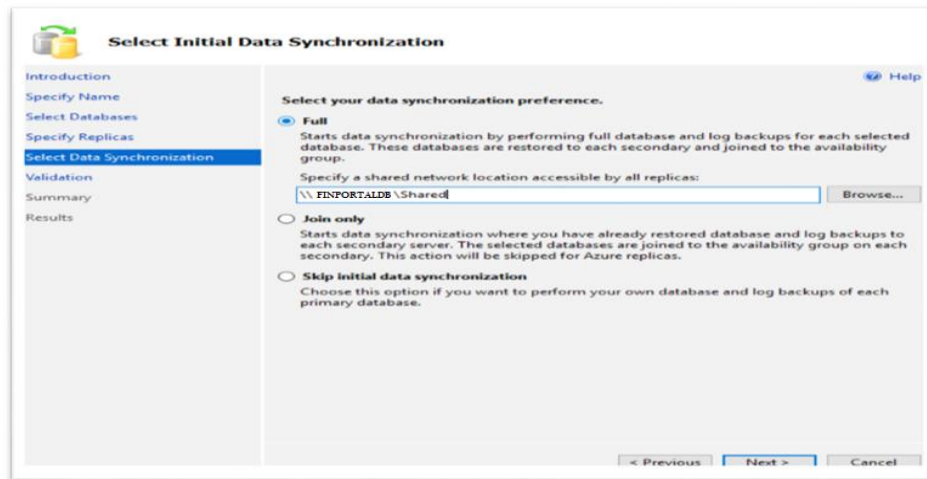
In **Data Synchronization step**, Select **Full** where it performs Full and Log backup and puts it into shared location path.

Create a Shared location

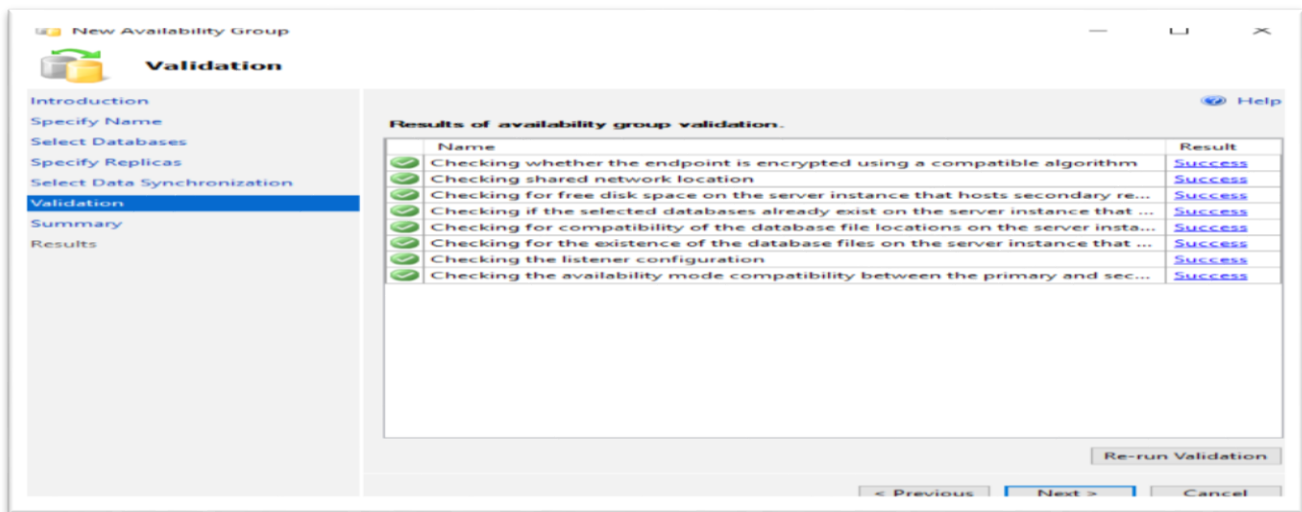
Connect to node SQL1 and choose desired drive location and folder, **Right-click folder** and **Go to Properties**, **Select Security** and click **Share**. It will show a pop-up window to add a user to whom, share permission need to grant.

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**.

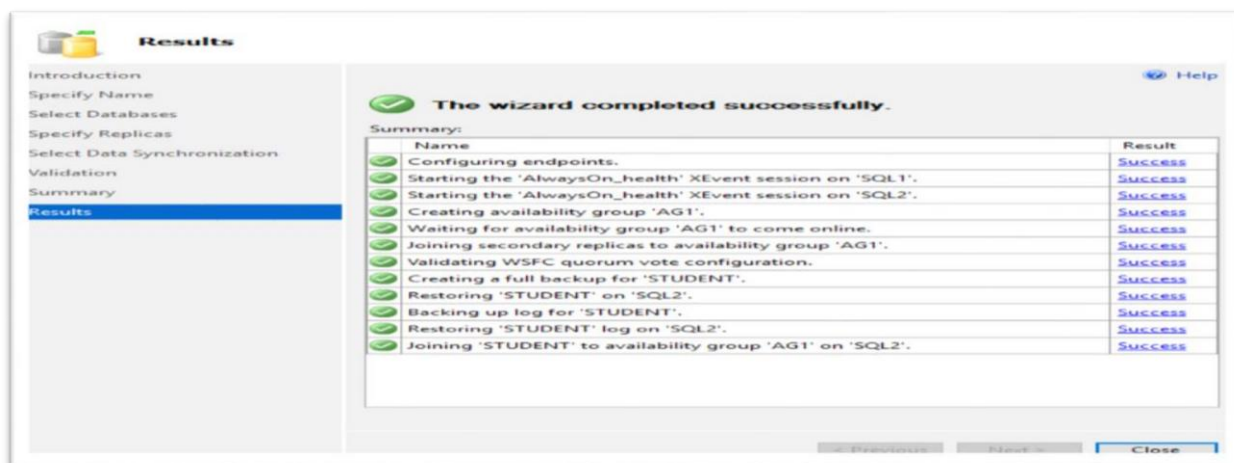




In **the Validation step**, SQL Server performs validations against any required actions, and it will provide warning or error messages for missing required steps.



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.

