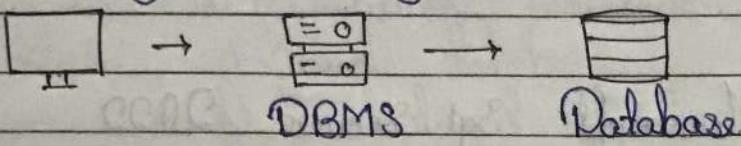




SQL

Database: It is organized collection of data so that it can be easily accessed.

To manage these database, DBMS (database management system) are used.



Types of DBMS :-

- Relational DBMS
- Non-Relational DBMS

Relational DBMS :- In this DBMS, data stored in table format.

Roll No.	Name	Class
1.	Ankit	5 th
2.	Aayti	6 th
3.	Anshika	7 th

Ex:- MySQL, Oracle

Non-Relational DBMS:- In this DBMS data is stored in Key-Value pair.

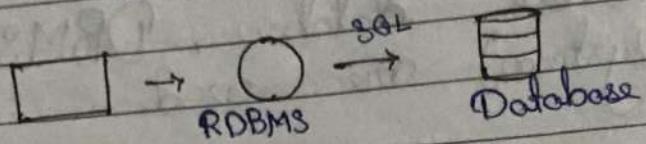
```

{
  "Roll no": 1,
  "Class": "5th",
  "Name": "Ankit"
}
  
```

Date: / /

Ex:- MongoDB, Redis

SQL:- Structured Query Language is used for update, delete, insert data in table or relational Database.



* Download Microsoft Sql Server 2022

Download Developer edition

Select an installation type: Download media

Select iso and download

Mount the iso file.

Database is an organized collection of structured information, or data, typically stored electronically in a computer system.

Types:-

- a) Hierarchical database
- b) Network database
- c) Object oriented database
- d) Relational database
- e) Cloud database
- f) Centralized database
- g) Operational database
- h) No SQL database

Hierarchical database look like similar to a family tree. A single object (Parent) has one or more object beneath it.

Relational database use SQL for CRUD (Creating, reading, Updating, deleting).

Object Oriented database store and manage object on a database server's disk. It is unique because associations between objects can persist.
Ex:- MongoDB realm.

* Installation of SQL Server.

- i) After mounting the iso select setup and run as administrator.
- ii) Select Installation and click new SQL installation or add features to an existing installation.
- iii) In edition section specify developer.
- iv) In license terms accept the license.
- v) In Microsoft update section click update.
- vi) In install rules we get warning and press.
- vii) In Azure extension Uncheck Azure.
- viii) In instance Configure we can use default instance or we can proceed with name.
- ix) In database engine configuration we can proceed with windows authentication mode but for safety we will go with mixed mode with SA account (System administrator) and enter password and click add current User.

x) In Analysis services configuration select tabular mode and add user.

Now start the installation.

Note:- In latest SQL versions we do not get SSMS (SQL Server management Studio). So, we have to download and install for using GUI to our database.

Connect to server

In SSMS Server type will be database engine and Server name we will get user by browse.

we can connect with windows authentication but for safety purpose we will authenticate with SQL Server authentication and login with Sa.

Note:-

32767 databases can be created in one instance.
 mdf :- meta data file.
 ldf :- log data file.

Date: / /



MySQL

Backup database using Cmd

C:\xampp
mysqldump

ANKIT@DESKTOP C:\xampp

mysqldump -u root -p Admin@123 Colour > C:\
users\ANKIT\Colour.sql

Enter password: *****

ANKIT@DESKTOP C:\xampp

mysqldump -u root -p Colour > C:\users\

ANKIT\Colour.log

Enter password: *****

C:\xampp\mysql\bin> mysqldump.exe -u root
-p Colour > Colour.sql

Enter password:



- we can create new database by right click on database or by creating in new query.
- In SQL Server Configuration manager we can start or run the services.

There are two databases:-
(i) System Database
(ii) User Database

System Databases

- (a) Master
- (b) Model
- (c) Msdb
- (d) Tempdb

Master database contain all information of db. of system-level information.

Model database contain information to create database's size. It used as template of all database created on the instance server.

Modification made to model database such as database size, collation, recovery model and other database option, are applied to any database created afterwards.

Msdb database is used by SQL server agent for scheduling alerts and jobs.

Tempdb database is a workspace for holding temporary objects or intermediate result sets.



tick the open the properties and finish.

iii) Tick Run whether user is logged on or not.
 Do not store password Run with highest
priviledge in conditions and uncheck
 Start the task only if the computer is
on AC power.

Now start backup 2pm.



My SQL Auto Backup Scheduling.

- Make a folder and create backup.bat and edit this text file and write.

```
@echo off
For If "tokens = 1-2" delims = /: "%%" a in
("TIME%") Do (set mytime=%a% %b%)
```

```
Set user = root
```

```
Set pwd =
```

```
Set backupdir = "C:\mysql\AutoBackup\backup"
```

```
Set dbname = Colour
```

```
Set dmpdir = "C:\xampp\mysql\bin\mysqldump.exe"
```

```
%.dmpdir% --host=localhost --user=%user% --password=%pwd% %dbname% > %backupdir%\%dbname%.mytime%.sql
%dbname%.mydate%.mytime%.log
```

By clicking this batfile we can get backup manually or we can schedule time for autobackup.

Scheduling

i) Open Task St Scheduler

ii) Create basic task

Name : backup 2pm > Daily > Select time
 Start programme > Browse this bat file >



My SQL using cmd

open xampp powershell

Administrator @ Ankit C:\xampp

mysql -u root -p

Enter password:

MariaDB [(none)] > show databases;

use database name;

Show tables;

desc cities;

Select * from cities;

Change password

Config.inc.php

'Config' > 'Cookie'

true > false

Password = ''

User account > root localhost : edit privileges
Change password.



ON (NAME = N'ANKIT' FILENAME = N'D:\SQLFile\Ankit.mdf', size = 8 MB,
FILEGROWTH = 8 MB)

LOG ON (NAME = N'ANKITlog' FILENAME = N'D:\SQLFile\Ankitlog.mdf', size = 8 MB,
FILEGROWTH = 8 MB)

- * How to assign a particular database to a particular user.

Security > Logins (new login) > SQL authentication
(fill name & password) > User Mapping
(check a database name and check
db owner & public).

- * How to assign permission of data to a user.

Login > User name (Property) > User mapping >
Select database name and give permission.

1> Select name from sys.databases
2> go

Now create Database

1> Create Database Ankit
2> go

Check Version of Database

Select @@Version

Check Connection

sp_who or sp_who2 (in query & execute)

exec sp_who2 or exec sp_who (in cmd)

go

Use Database in query

USE Database name (Execute)

Create Database in query

Create database Ankit;
CREATE

Change location of database

CREATE DATABASE Ankit



Create DB in SQL Server.

We can login Server name as :-

- i.
- ii) localhost
- iii) 127.0.0.1
- iv) Server name

Disable window Authentication

• Security > Logins > Ankit\Administrators (property) > Status > check Permission Deny & login disabled

Open Database using CMD (Default DB)

Sqlcmd

- 1) Select name from sys.databases;
- 2) go

Open Database using CMD (window authentication)

Sqlcmd -S Desktop\SQL2014 -F

- 1) Select name from sys.databases;
- 2) go

Open Database using CMD (User)

Sqlcmd -S Ankit\SQL2014 -U sa
password :



In backup options we can compress db.

Backup Database using T-SQL Script.

- Full Backup

```
BACKUP DATABASE SalesOrder
TO DISK = N'SalesOrder', N'D:\SQL Backup\1
STAT      SalesOrder.bak' WITH INIT,
NAME = N'SalesOrder',
STATS = 10,
COMPRESSION
```

go

- Differential Backup

```
Backup DATABASE ANKIT
TO DISK = N'D:\SQL\1\ANKIT.dif' WITH DIFFERENTIAL,
NAME = N'ANKIT',
INIT,
STATS = 10,
COMPRESSION
GO
```

- Transactional Backup

```
Backup DATABASE ANKIT
TO DISK = N'D:\SQL\1\ANKIT.trn' WITH INIT,
NAME = N'ANKIT',
STATS = 10,
COMPRESSION
GO
```



1> Select name from sys.databases;
2> go

3> drop database Ankit;
4> go

Backup Database

- Full Backup
- Differential Backup
- Transactional Backup

we can backup a database by using Full Backup. When the database is processing full back, the time between the process we can start differential backup. So we do not have any database left for backup.

If a database taking 1 hr for full backup then Differential backup get that 1hr of backup.

After differential we get transactional backup which keep record of transaction happen after differential. like delete.

Backup in GUI

DB > Tasks > Back up > add path > file name

- * Full backup extension is .bak
- * Differential backup extension is .dif
- * Transactional backup extension is .trn



Create database in different path.

CREATE DATABASE RegTestDB3

ON (NAME = N'RegTestDB3',
 FILENAME = N'D:\Database\RegTestDB3.mdf',
 SIZE = 4,
 MAXSIZE = 64)

Log ON

(NAME = N'RegTestDB3_log',
 FILENAME = N'D:\Database\RegTestDB3_log.ldf',
 SIZE = 4,
 MAXSIZE = 64)

GO

* Extension will be .mdf and for log .ldf.

Delete Database

we can simply delete database by right click on database or run query.

DROP DATABASE RegTestDB3;

we can detect is database if database is there or not.

DROP DATABASE IF EXISTS RegTestDB3;

Delete database using CMD

Sqlcmd -S ANKITISQL -U sa
 password:



Auto Backup / Schedule Backup

Management > Maintenance plans (New maint. plan Wizard) > Changes (Frequency: daily)

Set daily frequency time.
we can modify time in maintenance plan.

Error Check

SQL Server Agent > Error logs > Current (View agent log)



Restore

Database > Restore Database > Device (location) > Destination (New database name) & Options (Uncheck fail-log backup).

Select a database and restore full, differential and transactional backup, respectively.

In options we can choose Overwrite the existing database (with Replace) & close existing connections to destination database.

In options if we restore with nonrecover then check fail-log backup. (makes log)

Restore in T-SQL Script

Restore Database Ankit with Recovery;

Insert a data in database.

Select Top (1000) [user_id], [name]
From [SalesOrder].[dbo].[users]

INSERT INTO [SalesOrder].[dbo].[users] ('user_id',
'name') Value ('5', 'Ankit')



Script

we can not restore a database created in higher version to lower version.

Ex:- If i created a database in SQL 2022 and restore in SQL 2014 then it is not possible. Then we use Script.

But we can restore lower to higher version

Generate Script

Any Database > Tasks > Generate Script >
Select specific database objects and check
Tables & dba users > Save as script file >
Browse location > Advanced (Types of data to
script : Schema and data)

Now Copy the script and open in
SQL Server where we want to restore
and open new query and paste it. And
write on first line "CREATE DATABASE
[Database Name]". Now Execute.

Replication

Replication is a set of technologies for copying and distributing data and database objects from one database to another and then database to another and then synchronizing between database to maintain consistency.

- a) Transactional Replication
- b) Merge Replication
- c) Snapshot Replication

Replication Part

- Distributor
- Publisher
- Subscriber

Distributor is a database instance that acts as a store for replication specific data associated with one or more publishers.

Publisher is a collection of one or more articles from one database.

Publication is a database instance that makes data available in other locations through replication.

Subscriber is a database instance that receives replicated data.



Jobs > Plan Plan Planname (Properties) > Notification
(Enable Email & Select operator), Select when
the job completes.

Now Execute

Management > Maintenance plan > Plan name
Execute

Create new Job

Jobs > new Job > Name > Steps (new) > Step name
Command: Select name from sys.database: >
Notification > E-mail.

Email Configure

Management > Database Mail > Configure database mail > New profile & name, description and Add > Account name, description, E-mail address, Reply e-mail (from sender), Server name: smtp.gmail.com, port no.: 587. Check SSL, User name: (Sender gmail), Password: (generated in gmail 'app password'). Select profile.

Now test the Configuration

Database name > Send Test E-mail > To & Body.

Send Email Scheduling

Create a backup maintenance plan.

SQL Server Agent (Properties) > Alert System > Enable mail profile (Mail System: Database Mail, Mail profile: DBMailProfile) > Check Replace tokens for all job responses to alerts.

Now make operator

Operator > New Operator > Name: SQLBackupJobs, Email name: ankit@gmail.com, arpit@gmail.com.

Now integrate jobs



Check Distribution is created in System Database or not.

Add User

Logins (New login) → Search → Advance → Find Now
 Select all four users. (Snapshot) → User mapping (Select the db you want to replicate) → DB owner → Check distribution and make DB owner.

Now add all four users respectively.

Create Publication

Replication → Local Publication (new publication)
 Select DB we want to replicate → Transactional Publication → Check Tables, Stored, Views, indexed, users defined → check Create Snapshot → Snapshot agent (Security) → Process account: ~~→~~
 Paste the Logins user name (Desktop\repl → Snapshot as well as Snapshot) add password → Uncheck use the logon user. Security setting from the snapshot agent. → Publication Name.

Add Distribution

Replication → Local publication → DBname (Properties) → Publication Access List → Add (repl distribution)



Note:- If we are replicating database in single server then we have to create all four users, But if we process in two different server then we only create distribution & merge.

Add Subscriber Transactional Replication

Create folder

This PC > C:drive > Program files > Microsoft SQL Server > MSSQL12.MSSQLSERVER > MSSQL > Data > Create folder (Repldata)

Repldata (Property) > Sharing > Advance Sharing > Check share this folder. > Permission (Add) → Advance → Find Now → Select four user we have created)

Give Read Permission to distribution, logreader, merge and give full permission to Snapshot.

Now Copy the Network path: \\Desktop\repldata

Repldata (Property) > Security > Edit > Add > Advance) Find Now → Select four user we have created.

Give Read Permission to distribution, logreader, merge and full permission to Snapshot.

Configure distribution in first Server

Replication (Configure distribution) > Desktop > Paste the Network path: \\Desktop\repldata in Snapshot folder.



1. Transactional Replication is a process of copying database from a source server to a destination server. This type of replication is used when the data in the source and destination databases need to be kept sync.
2. Merge Replication is a process of copying data between two or more databases and merging them into a single database. The source and destination database can be located on different servers.
3. Snapshot Replication is a process of copying data from source database to a destination database. The process creates a snapshot of the data at a particular point in time and transfers the data to the destination.

Create User

Computer Management \rightarrow Local Users (Users)
 New User (Set Username & Password and check password never expires).

Now repeat and create users :-

- (i) repl_distrib
- (ii) repl_logread
- (iii) repl_Snapshot
- (iv) repl_merge



Monitor the replication

Replication > Local Publications > SalesOrder (Launch replication monitor)

Insert into table and replicate

Database > SalesOrder > Tables > dbo.users (Select 1000 rows)

```
SELECT TOP (1000) [user_id], [name]  
FROM [SalesOrder].[dbo].[users]
```

```
insert into users (name, user_id, name) Values  
(6, 'Ankit');
```

Now check in Server 2 DB replicated.

Check Snapshot agent status

Replication > Local publication > SalesOrder (View Snapshot agent status)

Check Logreader agent status

Replication > Local publication > SalesOrder (View Logreader)



SalesOrder Trans.

Replication > Local Publications > SalesOrder (View snapshot agent)

Replication > Local Publications > SalesOrder (new subscription) > Add Subscriber > login (new Database) SalesOrder_Rep > Connection to subscriber > Process account: Desktop\repl distribution and add password.

Before Connection to subscriber make sure to follow this steps:-

2nd Server > Security > logins > Desktop\repl distribution (property) > User mapping > SalesOrder_Rep (db. owner)

1st Server > Replication > Local publications > SalesOrder (Property) > Publication access list Add > Desktop\repl distribution.

Now check the replication

1st Server > Databases > SalesOrder > Tables > dbo. users (Edit 1000) > write (insert into users (user_id, name) Values (4, 'Ankit')) Select and execute.

Db will be replicated in 2nd Server.

Add Subscriber

Replication > Local publications > DB Name (New subscription) > Add Subscribers in subscribers > Add SQL Server Subscriber. New Connect and check Server and select new database.

Security > Logins (New login) > Search > Advanced > Find New (Repl distribution) > User mapping (Select the database we created and check db owner) > Connection to subscriber > Process account: Desktop > repl distribution and add password.

Note: Changes not permitted

Tools > Options > Designers > Table > Uncheck Prevent saving.

Example :-

Security > Logins > Desktop > repl Snapshot (Property) User mapping > SalesOrder (db owner).
Repeat this with repl distribution, repl logreader, repl merge.

Replication > Local Publication (New publication) > SalesOrder > Transactional Publication > check tables > check create snapshot > Snapshot agent (Security) > Process account: Paste the login user name (Desktop > repl snapshot) > Uncheck Use the security > Security setting > Process account: Paste login user (Desktop > repl logreader) > Publication name:



Merge Replication

Create new database.

Database (new database) > name: ReliMerge

Create table

Database > ReliMerge > Tables (new → table)

Column Name	Data Type	Allow Null
User_id [PK]	int	
Name	nvarchar(50)	✓

Table name users.

Create another table

Database > ReliMerge > Tables (new → table)

Column Name	Data Type	Allow Null
Product_id	int	
Product_name	nvarchar(50)	✓

Table name products.

For Replication assign Login

Security > Logins > Desktop | repldistribution (Properties)
User Mapping (Select Relimerg and db owner)



Articles \rightarrow uncheck show only and check all the tables.

Replication \rightarrow Local publications \rightarrow SalesOrder (Reinitialize all subscriptions) \rightarrow make for reinitialize.

Replication \rightarrow Local publications \rightarrow SalesOrder (view snapshot agent) \rightarrow Start.

Note: We can schedule the replication.

New Table will replicated in 2nd Server.



Add New Table

Database > SalesOrder > Table (New + Table)

Note :- For replication we need a primary key

Column Name	Data Type	Allow Nulls
Primary Key	Address id	int
State	nvarchar (50)	✓
City	nvarchar (50)	✓
Country	nvarchar (50)	✓
Identity Specification (Is Identity)	Yes	
Identity Increment	1	
Identity Seed	1	

Table Name will be address.

Edited 200 rows of dbo.address

Address id	State	City	Country
NULL	Jharkhand	Ranchi	India

Note: Without primary key we can not perform replication.

Now add this table in replication.

Replication > Local publications > SalesBro (Property)



Snapshot Replication

Create new database

Database (new database) > name: Donation

Create Table

Database > Donation > Tables (new + table)

For Replication assign login

Security > Logins > Desktop \ repl distribution (properly)
User mapping (Select Donation and db owner)

* Same step follow for Desktop \ repl logreader, Desktop \ repl merge and Desktop \ repl snapshot.

Create publication on local

Replication > Local publication (new publication) >
Select Repl merge > M Donation > Merge publication
Snapshot publication > Select all tables > Snapshot
agent change and schedule daily > Snapshot
agent: Security settings (account: Desktop)\
repl snapshot, and add password) > Publication
on name: Donation_Snapshot.

Add Subscriber

Replication > Local publication > Donation_Snapshot (new)



Give Permission on 2nd Server

Security > Logins > Desktop\repl_distribution (property) > User mapping → Select ReliMerge_mRep and db owner

>Login (new login) > Search > Advanced > Find Now (Desktop\repl_merge) > User Mapping (Select ReliMerge_mRep and db owner)

In 1st Server

Replication > Local Publications > ReliMerge (Properties) Publication Access List → Add (distribution)

Finish Subscription

Replication > Local publication > ReliMerge (view Snapshot agent status).



Security > Logins > Desktop1 repl1greader (Property) > User mapping (Select Relimerge and db owner).

Security > Logins > Desktop1 repl1merge (property) > User mapping (Select Relimerge and db owner).

Security > Logins > Desktop1 repl1snapshot (Property) > User mapping (Select Relimerge and db owner).

Create publication on Local

Replication > Local publication (new publication) > Select Relimerge > Merge publication > Select all tables > Snapshot agent change and schedule daily > Snapshot agent: Security settings (account: Desktop1 repl1snapshot and add password) > Publication name: Relimerge-Merge.

Note: In Transactional Replication we need primary key for replication but in merge replication we can replicate without primary key.

Add Subscriber

Replication > Local publication > Relimerge (New subscription) > Subscribers > Add SQL Server Subscriber > Login and Connect server > Subscription db > new database (name: ReliMerge-mRep1) > Merge agent Security > Clicks on 3 dots > Process account: Desktop1.repl1merge and password. Finish after permission.



This PC > E: > SQL > Shiplog (Property) > Sharing
 Advance Sharing > Check Share this folder >
 Permission: full

Copy the Network Path: \\Desktop\Shiplog

Now Go back back to LogShipping Steps

Check Enable this as primary database > Backup
 Setting > Paste the path in Network path with
 1 > Schedule > Occur every 5 min

Add Secondary Instance in Database properties -
 LogShipping

Secondary Database (Add) > Connect (Connect
 Secondary Database) > Check on No, the
 Secondary Database is initialized.

Now Back up the Database

Database > LogShipping > Tasks > Backups > Add Path

Now Restore the Database in Secondary Server.

Database > Restore Database > Device > Add (Select the
 path of LogShipping) > Options > Restore with no
 recovery

Share Secondary Server's folder and copy network
 path \\DENNYCARLOS\Shiplogreceive

Log Shipping

SQL Server log shipping allow us to automatically send transactions log backups from a primary database on a primary server instance to one or more secondary databases on separate secondary server instance.

- Back up the transactional log at the primary server instance.
- Copy the transactional log file to the secondary server instance.
- Restore the log backup on the secondary server instance.

Create a new database.

Let we create database with name LogShipping and then create table as name users.

LogShipping > Tasks > Ship Transaction Logs

Now create folder in Primary Server as Shiplog and in secondary server as Shiplogreceive.

Share The paths



Subscriber) > Subscriber → Add SQL Server Subscribers
 Login and connect 2nd Server > Subscription db →
 new database (name: Donation_snapshot) > Merge agent
 Security → Click on 3 dots > process account:
 Desktop\repl_snapshot and password.
 *Finish after giving permission on 2nd Server.

Give Permission on 2nd Server.

Security > Logins > Desktop\repl_distribution (property)
 User mapping → Select RepMerge Donation_snapshot
 and db owner.

Login (new login) > search > Advanced > find New
 (Desktop\Donation_repl_snapshot) > User mapping
 (Select Donation_snapshot and db owner)

1st Server

Replication > Local publications > ^{Donation} RepMerge (property))
 Publication Access List → Add (distribution)

* Now Finish the step of Add Subscriber.

Replication > Local publication > Donation (View
 snapshot agent status).



Mirroring

Database mirroring maintains two copies of a single database that must reside on different server instance of SQL Server database engine.

Create a new user as name 'Mirroring'. And add in administrator group.

Create a new database as name 'mirrorDB'.

Create new table as 'users'.

Backup this database (full, transactional).

Restore the database in Secondary instance (full as well as transactional). with norecovery.

Steps for mirroring.

Database > mirrorDB > Task > Mirror > Configure security > check no > Mirror server instance (connect) > Connect secondary server.

Note:- After creating user we must change SQL Configuration.

Brows new user and restart SQL Server

Start SQL Server agent.

Login this user in Security, and give server roles (sysadmin) in both instance. & restart.



```
SELECT TOP (1000) [userid], [username]
FROM [LogShipping].[dbo].[users]
```

```
insert into users (userid, username) Values (1, 'Chirag')
```

```
insert into users (userid, username) Values (2, 'Ankit')
```

```
insert into users (userid, username) Values (3, 'Aspit')
```

```
insert into users (userid, username) Value (4, 'Shweta')
```

How to close LogShipping

Database > LogShipping > Tasks > Ship Trans Logs > Uncheck
Enable this as a primary database

2nd Server

New Query

```
Restore Database LogShipping with RECOVERY  
GO
```

Start Jobs in both server

Configure Log Shipping in Server 1st

Database > Log Shipping (Tasks) > Ship Trans log >
 Check enable > Backup Setting > Paste 1st Server
 folder's network path (\\DESKTOP1\Shplog1) >
 Schedule > Occurs every 5 min > Compression:
 Compress backup > Secondary database (Add) > Connect
 (Connect 2nd Server) > Check No, the secondary >
 Secondary database: Log Shipping > Copy Files > Paste
 the secondary folder network path \\DENNY1
 Shplogreceive1 > Schedule in 5 min > Restore trans-
 action log > Schedule in 5 min. OK

Start JOB (1st Server)

SQL Server agent > Jobs > 1 Backup Log Shipping
 (Start job at step)

Start JOB (2nd Server)

SQL Server agent > Jobs > 1SCopy DESKTOP\LogShipping
 (Start job at step) Same for 1S Restore

Note:- In Configure Logshipping we have to
 paste location not network path.

Insert Users in Table in Server 1

Database > Logshipping > Tables > dbo.users (Select top
 100)

Rename your pc of VM AD Server 1.

Change IP of 1st Server

Run → ncpa.cpl → Ethernet (property) → IPv4

Add role and features in VM1

Server Manager → Dashboard → Add role and feature → Server roles (Add Active Directory Domain Services, DHCP Server, DNS Server) → Confirmation (Check restart)

Promote the server as domain controller

Server manager → Dashboard → click on (F5) → Promote this server to a domain controller: Deployment (Add a new forest) → Root domain name: ankit.in → add password

Remote Desktop as domain controller
User name: ankit. ankit\administrator

Remove the gateway of IPv4

Reserve Lookup Zone

Server Manager → Local Server → Tools → DNS

AD Server → Reserve Lookup Zone (new zone) → Reverse Lookup Zone: 192.168.29.



Cluster

SQL Server clustering is the term used to describe a collection of two or more physical servers (nodes), connected via a LAN, each of which host a SQL server instance and have the same access to shared storage.

Cluster in SQL are used to store data that is from different tables in the same physical disk blocks.

- For clustering we need 3 Servers we have to make 3 VMs.

AD → Active Directory

FQDN → Fully Qualified domain name.

DNS → Domain Name System.

After creating 3 VMs we have to check IP address with IPConfig. and for ping with one server.

Let we create a VMs of windows server the we need to turn off the firewall and check IP address.
Now ping from main server. If it ping successfully we need to Remote Desktop Connection with mstsc from main server.

First of all we have to allow Remote remote access of VM Server.



Check jobs

Create table and view on 2nd instance.

How to stop mirroring.

Database mirrorDB > Task > Mirror > remove mirroring

ONLINE Database

new Query

Restore Database mirrorDB with recovery.

Note: If we want to start mirroring again we must restore again with no recovery in 2nd instance.



Configure failover cluster

- * Take remote of Node 2

Server Manager → Dashboard → Add roles and features → Features (Select failover clustering) → Confirmation (check restart) → Install

- * Take remote of Node 1 and Configure same as above.

• Ping : SQL Node 1 ↔ SQL Node 2

In Ethernet properties uncheck IPv6 (Disable IPv6).
In both node and ping.

Open failover cluster

Server manager → Dashboard → Tools → Failover cluster manager →

- * First of all we need to add both nodes in domain.

Server manager → Local server → Workgroup → Change
click on domain : ankit.in

Computer name : adadmin

Now Remote desktop using domain not administrator.
ankit\adadmin

We need One AD Server and Two Node.

We will make Three Vm of windows Server.

Domain Name : ankit.in

Change Name of 2nd Vm (Node 1)

- First we create 1st Vm (Ad Server)
- Second we create 2nd Vm (SQL Node 1)
- Third we create 3rd Vm (SQL Node 2)
- SID Should not be same.

Command : wmic useraccount get sid

If SID is same then we must change it :- Run → Sysprep

C:\Windows\System32\Sysprep press OK
Generalize → OK

Change Name :- This PC → Property → Rename
(SQL Node 1)

Change Name of 3rd Vm (Node 2)

This PC → Property → Rename (SQL Node 2)

- Note:
- Change IPV4 address of both Node with same series and different no.
 - Enable remote desktop
 - Disable firewall.
 - DNS of both node's IP is AD Server IP.

Active Directory Users on VM

Run dsa.msc

ankit.in → Users → New + User (Name: ADAdmin) (User: adadmin) → Domain Admins properties → Member → Add (ADAdmin)

Remote desktop

ANKIT\admindadadmin

IPV4 of AD Server : 192.168.29.176

IPV4 of Node 1 : 192.168.29.177

IPV4 of Node 2 : 192.168.29.178

Above IPV4 is only example.



Add Disks

Login with domain: amkit\adadmin in all servers

Node 1: Server manager → Dashboard → Tools → Failover cluster manager > Disk Storage → Disks (Add disk).

Check same in Node 2.

Configure Quorum Disk

Node 1

Server manager → Dashboard → Tools → Failover cluster manager → SCVAGCUI (More actions) → Configure cluster quorum settings
 Select quorum configuration (Select Advanced):
 Select quorum witness (Select configure a file share witness) → Share Path (Share the network path of folder which we have to create before this process in AdServer: FShare) → Finish

Check AdServer there is file created as witness name in Fshare folder we have created.

Configuration Completed for Cluster.



Server Manager → Dashboard → File and storage
 Services → iSCSI → To create on iSCSI Virtual
 disk → Select by Volume: C Volume → Name:
 Data → Size: 5 GiB → Target name: S01 → Add →
 Type: IP Address, Value: 192.168.29.177 → Add →
 Type: IP Address, Value: 192.168.29.178 → Create

uster

Initiate iSCSI in Node 1

7.9
 Server manager → Dashboard → Tools → iSCSI initiator

Target: 192.168.29.176 → Connect → Done

Server manager → Dashboard → Tools → Computer
 management → Disk management → 5 GiB initialize
 disk (GPT) → New Volume

Initiate iSCSI in Node 2

Server manager → Dashboard → Tools → iSCSI initiator
 Target: 192.168.29.176 → Connect → Done

Server manager → Dashboard → Tools → Computer
 management → Disk management → 5 GiB initialize
 disk (GPT) → New Volume

Configure Node and create Cluster

Node 1 : Server manager → Dashboard → Tools → failover cluster manager → Validate configuration

Select Server or a cluster → Browse → Check names (Select both node) → Finish with create cluster

Failover cluster manager → Create cluster wizard → Access point :- Name: SQLAGCLU
Click here to type an address: 192.168.29.179

After creating new cluster check the status of Nodes in node 1.

Failover cluster > SQLAGCLU > Nodes

Now check the status of Nodes in Node 2.

Server manager → Dashboard → Tools → Failover cluster manager → SQLAGCLU → Nodes

iSCSI = Internet Small Computer Systems Interface

Add iSCSI in AdServer

Server manager → Dashboard → Add roles and features → Server Roles (Select File and Storage > File and iSCSI Services > Select iSCSI Target Server) → Install.

Cluster the database

If we have installed SSMS in both nodes and AdServer then we can proceed for clustering

Login as domain and open SSMS in node 1 and connect as SQLNODE1\SQLINS1

For our convenient we need both instance Node 1 & Node 2 on same screen so we give permission to access both instance in same database server.

Node 1 instance : SQLNODE1\SQLINS1

Node 2 instance : SQLNODE2\SQLINS2

Node 2.

MS SQL Server Configuration manager → SQL Server Network → Protocols for SQLINS2 → TCP\IP → Enable → Name pipes → Enable

SQL Server Configuration manager → SQL Server Services + SQL Server (Restart)

Follow Same Steps in Node 1 also.

Firewall must be disabled.

Connection access test in both node

Enable Extension → Create new file as name : new text document.udl (open) → Server name Enter :



Paste in SQL Server agent database engine, SQL Server integration services 16.0 and enter password and startup type will be Automatic.

Now account name is changed in SQL Server Agent, SQL Server Database, SQL Server integration services 16.0.

Check on grant perform volume → Database engine Configuration: Mixed mode and enter password and Add current user (ANKIT\adedm). Tabular mode and add current user → Install.

Node 2

Server manager → Dashboard → Tools → Computer manager → Local users → Groups → administrator + add → Sql-node1 (check names)

Now install SQL Server and SSMS in Node 2
Same as Node 1.



Install SQL Server and SSMS in Server.

After completing configuration we need to install SQL Server and SQL Server management studio in our windows server.

SQL Server Installation in Node 1

SQL-Server (Mount) → Setup (Run as administrator) → Installation → New SQL Server → Developer → Accept terms → Feature Selection (Select all) → Instance Configuration (Name instance: SQLINS1) →

in ADServer

Note: We need to create user for further steps. We must login as domain.

Ad Server:

Create account: Run (DSA.MSC) → ankit.in → New → User → Name: SQL-Node 1, User name: SQL-Node 1 → Password (Password never expired).

ankit.in → Users → administrator → Member of

Node 1: Computer Management → Local user + Groups → administrator → Add → SQL-node 1 (check names)

Now go back to installation process in Node 1

Server Configuration → SQL Server Agent: Account name (Browse → SQL-node 1 <check names>) → Password → Automatic
Copy the account name: ANKIT\SQL-node 1 and



we can access any database created in instance 1 or instance 2.

Test Failover Cluster

Open both instance in node 1

To test failover we need to open SSMS in AdServer

Server Name: SQLSN-MyAB-202, 1496

Always on High Availability \rightarrow Availability groups \rightarrow SQLAB2022 (Start failover wizard) \rightarrow Connect to replica (Connect) \rightarrow Server name: SQLNode2\SQLINS2 (Connect) \rightarrow Finish

Now check Instance 2 had become primary & Instance 1 become secondary

Create table in db (cluster test DB) and access it from Instance 2.

Now Stop SQL Server of Node 2 to check clustering.

After Node 2 got stopped then Node 1 become primary and cluster goes on. And we can edit db without any difficulty.

No data will lose if any one node stopped.

SQLNODE1\SQLINST1

Database → New database → DB Name : Cluster Test DB

Now backup this database.

Configure always on high availability.

Always on high availability → New availability group →
New availability group wizard → Group Name :
SQLAG2022 → Select database → Add replica →
Connect Instance 2 (SQLNODE2\SQLINST2) → Check
Automatic failover on primary as well as secondary →
Listener (Name Create an availability [DNS Name :
SQLSN-MyAG-202, Port : 1496, Add IPV4 : 192.168.
29.180}) Copy the Listener DNS Name which we
write and paste on notepad → In validation we
must success all now and then go to next →
Finish.

After configuration finish, check the availability
group of both instance there is availability
created SQLAG2022

Node 1

Always on high availability → Availability groups →
SQLAG2022 → Availability group Listener → SQLSN
(Start powershell)

We can connect SSMS in Ad Server Connect
Server name : SQLSN-MyAG-202, 1496



SQLNODE2 | SQLINST2 → Test Connection.

Enter user name and password.

Enter Server name of Second node in first node and name of first node in Second node to check the connection in udl file.

To switch node.1 & node.2 we need to always on :-

Node.1: Connect both instance (Login both instance in SQL Server node 1).

SQL Server Configuration manager → SQL Server Service. SQL Server (Properties) → Always on availability groups → Check enable and apply. And restart SQL Server.

Node 2: Same as node 1.