**High availability**

This high availability is based on the popular Always On technology, which is also available in on-premises SQL servers. However, you would have to configure, manage, and maintain Always On in an on-premises environment. In Azure SQL Database, it's configured, managed, and maintained by Microsoft

Azure SQL Database offers several high availability (HA) features to ensure your database remains accessible and resilient in the face of failures. Here’s an overview of the high availability options and features in Azure SQL Database:

### **Built-in High Availability**

#### **1. Service Tiers**

Azure SQL Database provides different service tiers, each offering varying levels of performance, features, and availability:

* **Basic, Standard, and Premium Tiers (DTU-based):** These tiers come with built-in HA features. The Premium tier provides the highest availability with three replicas (one primary and two secondaries) using synchronous replication.
* **General Purpose and Business Critical Tiers (vCore-based):**
  + **General Purpose:** Uses a separation of compute and storage. The compute layer uses local SSDs, while the storage is backed by Azure Premium Storage, providing redundancy.
  + **Business Critical:** Uses Always on Availability Groups to provide multiple synchronous replicas for high availability and to ensure zero data loss.

#### **2. Geo-Redundant Backups**

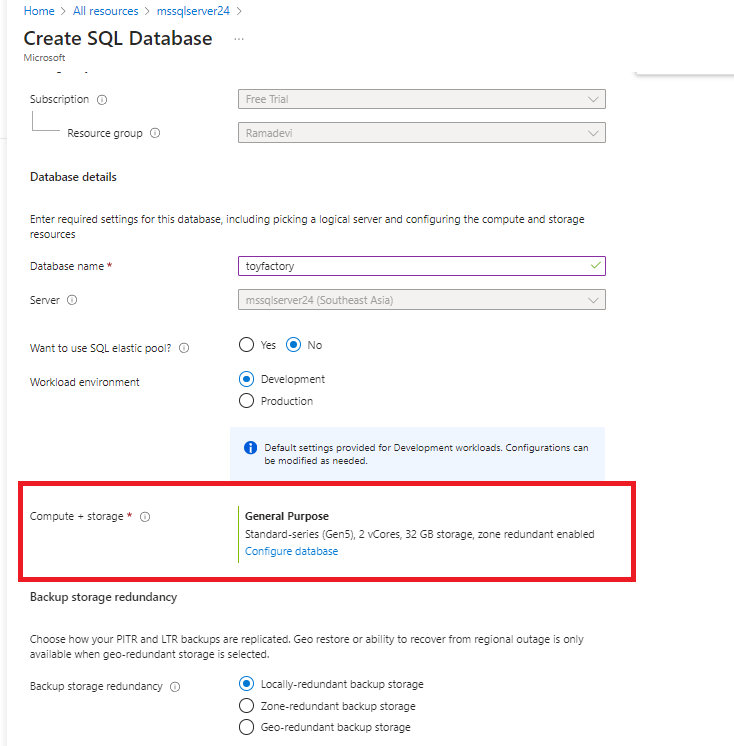
* **Automatic Backups:** Azure SQL Database automatically takes backups of your database and stores them in geo-redundant storage, ensuring that your backups are available even in the event of a regional outage.

#### **3. Zone Redundant Configuration**

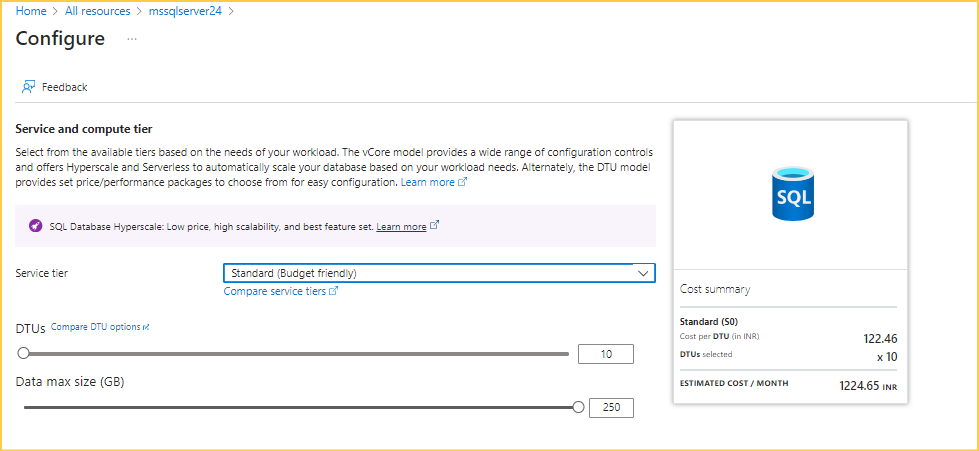
* Available in the Premium and Business Critical tiers, this configuration spreads replicas across different Availability Zones within the same Azure region to protect against data center failures.
* Zone redundant databases are only supported in the Premium service tier for databases within 1 TB in size

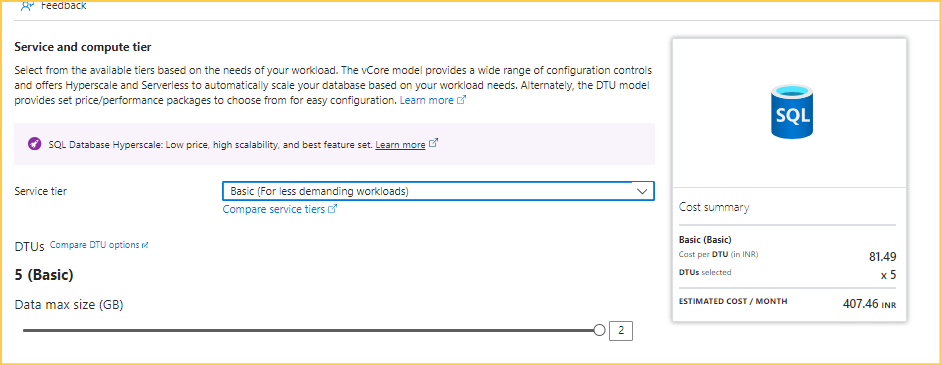
**Configuring zone redundant:**

* When creating the Azure SQL database, click on **Compute + Storage** and select a service tier where zone redundancy is available.

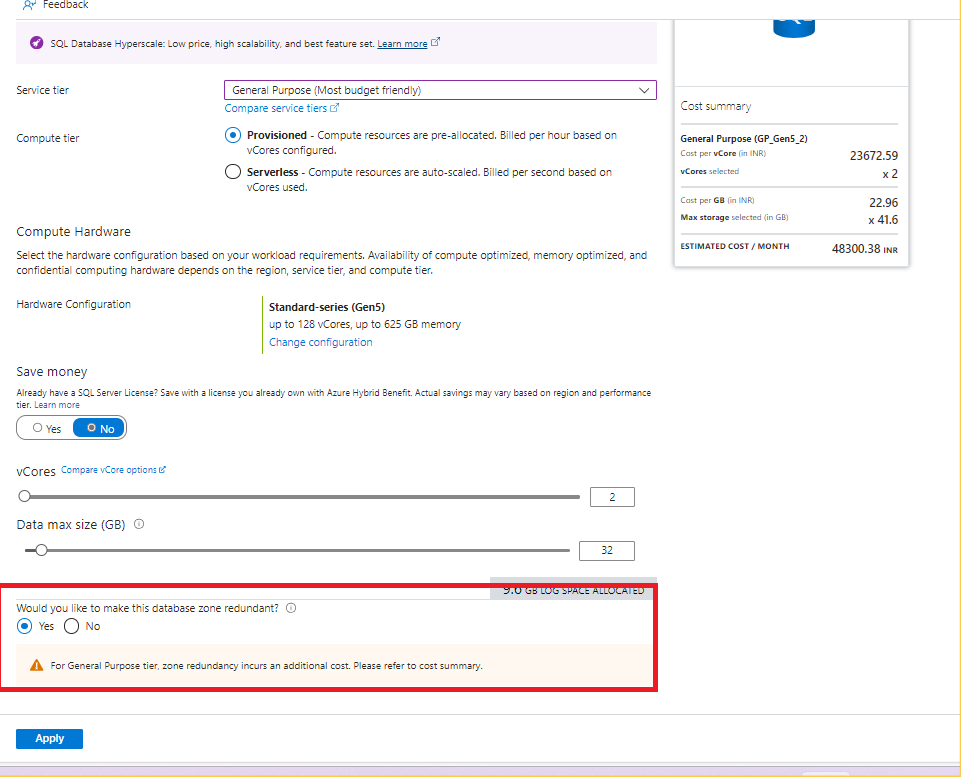


* For standard, basic DTU service tier zone redundant is not available





* Here we have service tier and compute tier
* In compute tier we have provisioned and serverless. Cost may vary based on selection



Disaster recovery

Microsoft provides three types of recovery solutions

* Automatic database backups
* Active geo replication
* Fail over groups

### **Automatic Database Backups**

Azure SQL Database automatically performs backups of your databases to protect your data from loss or corruption. These backups include:

* **Full Backups:** Taken weekly.
* **Differential Backups:** Taken every few hours.
* **Transaction Log Backups:** Taken every 5 to 10 minutes.

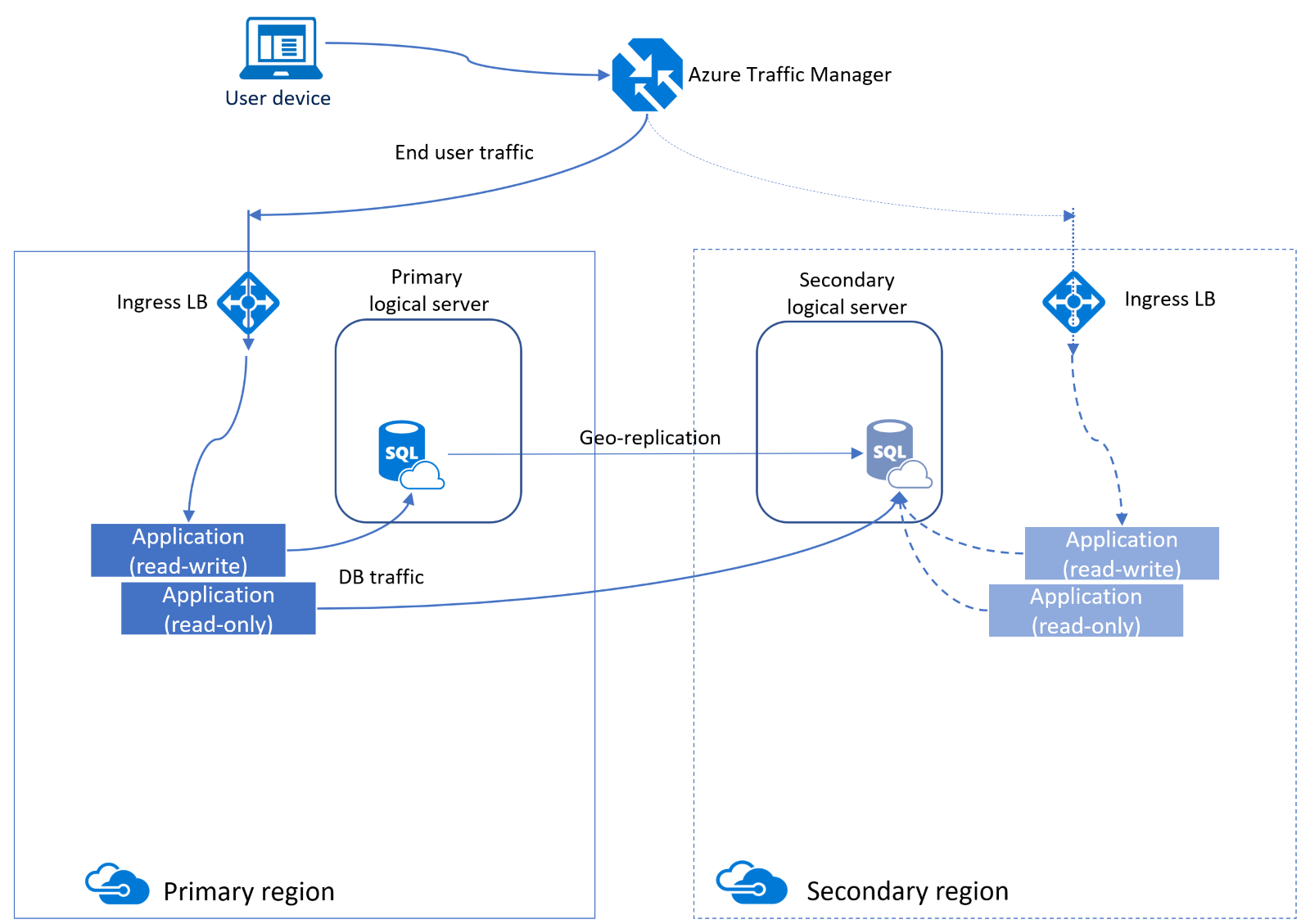
These backups are stored in geo-redundant storage, ensuring they are available even in a regional outage. You can use these backups to restore your database to any point in time within the retention period, which can be configured based on your requirements.

### **Active Geo-Replication**

Active geo-replication is a feature that allows you to create readable replicas of your Azure SQL Database in different regions. This provides several benefits:

* **Disaster Recovery:** In case of a regional outage, you can failover to a secondary replica in another region.
* **Read Scale-out:** You can offload read-only workloads to secondary replicas, thus reducing the load on the primary database.

Each secondary replica is created as a fully functional database that can be used for read operations. You can manually failover to any of the secondary replicas in the event of a primary database failure.

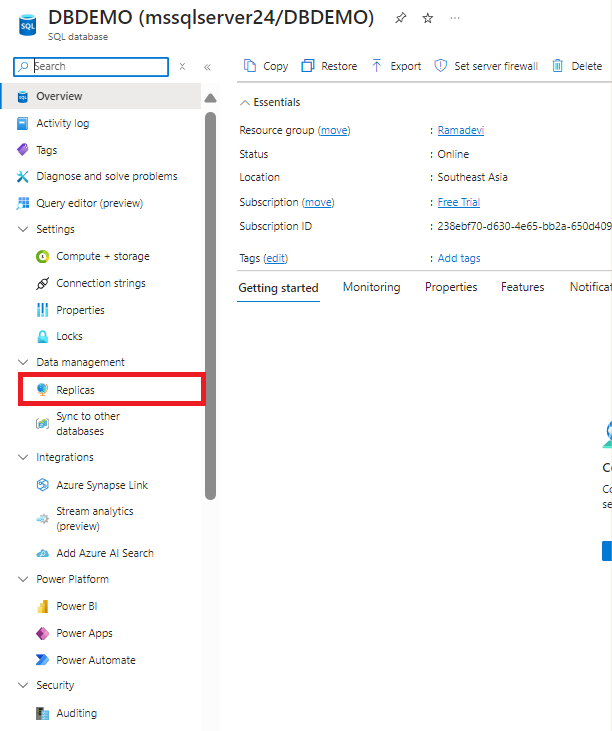


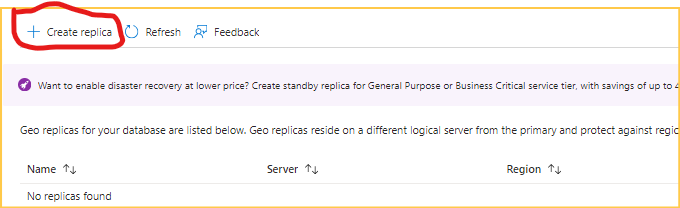
* Replicate the data from one server to another server
* We can configure for the only one database at a time
* Always on technology in SQL server
* It is Quick DR solutions in case of any regional disaster or large-scale outage
* Readable secondary single database or set of a database in elastic pool
* You can create up to 4 secondary replicas in same and different regions
* It is support asynchronously replicate committed transactions on the primary to a secondary using snapshot isolation
* We can use secondary readable databases for all the read only transactions like reports and database migrations
* It allows you to create a secondary readable database copy in a same or different region
* It leverages an always on high availability group technology which allow you to send data or replicate the data between primary and secondary servers
* Quick DR solution in case of any primary region outage
* We create up to 4 secondary databases in a same or different region
* Secondary database is read only
* Both servers are in same compute size and same service Tiere for better performance
* Data first committed on primary and send to the secondary server asynchronously

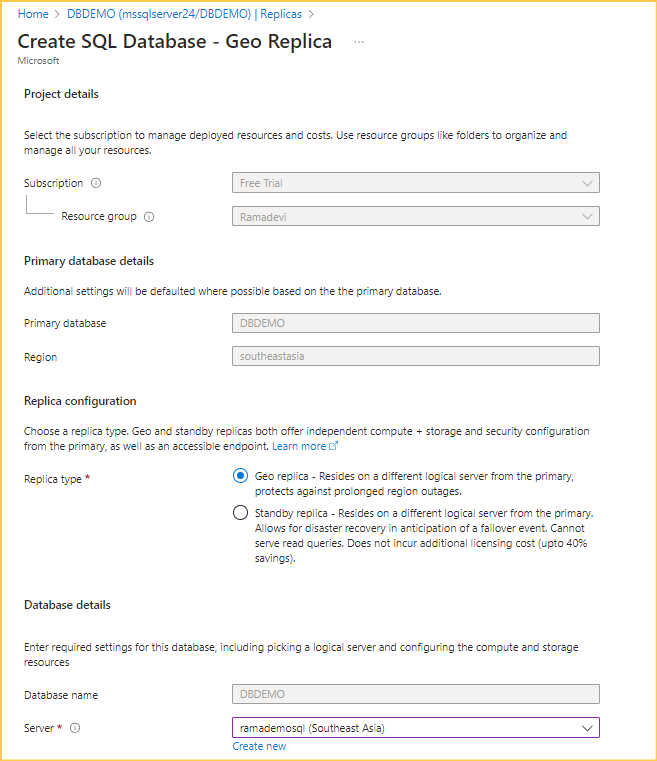
NOTE

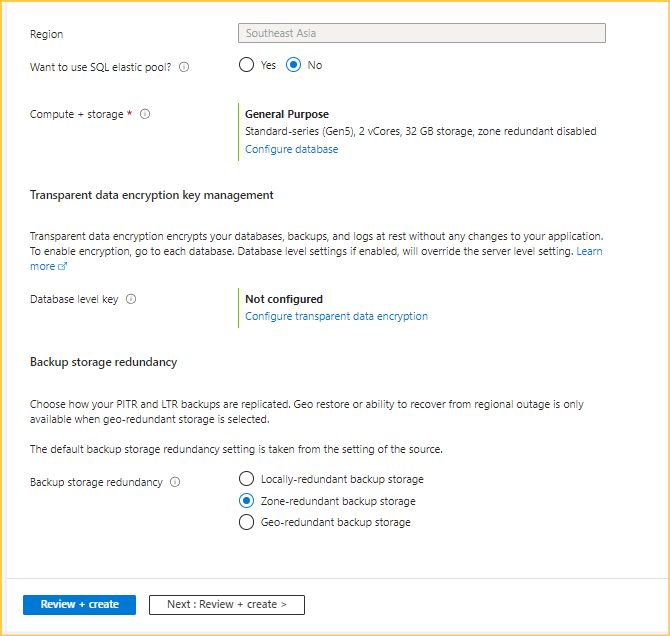
* Both primary and secondary databases are required to have the same service tier
* Same compute size

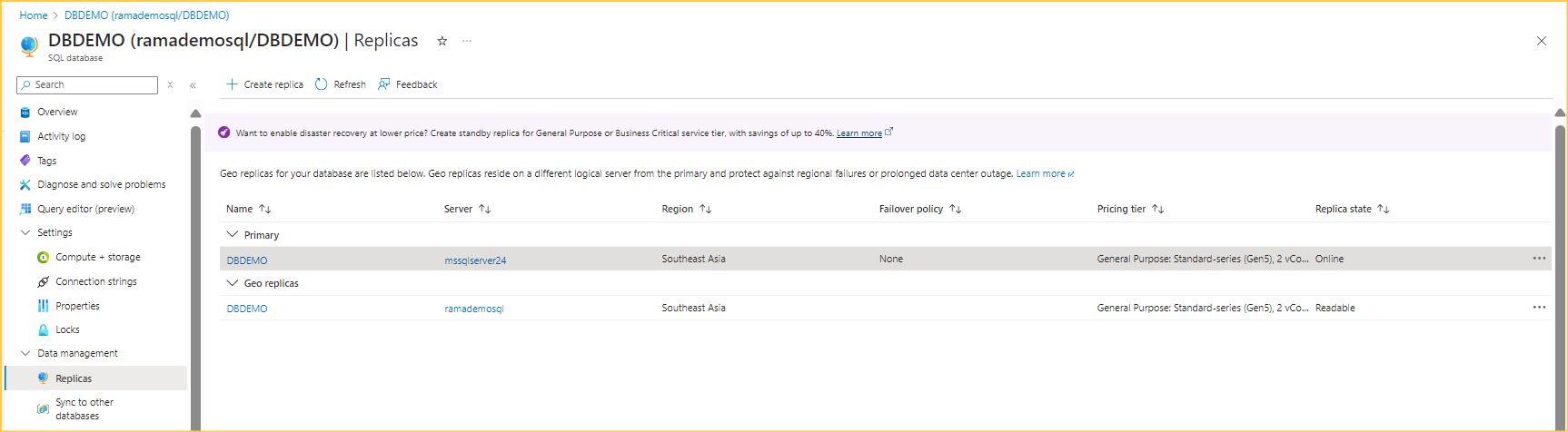
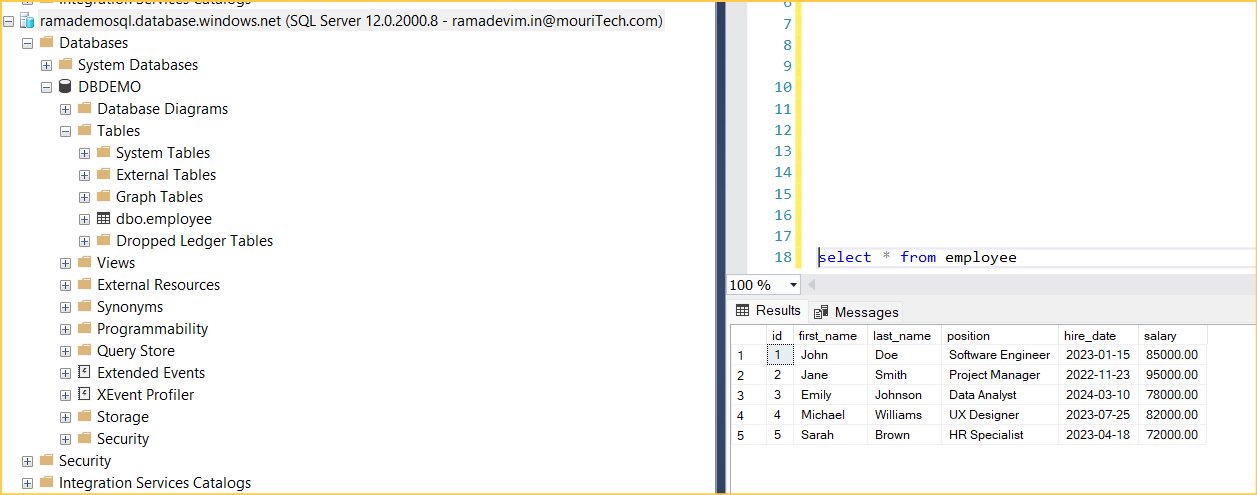
**Active Geo-replication configuration**

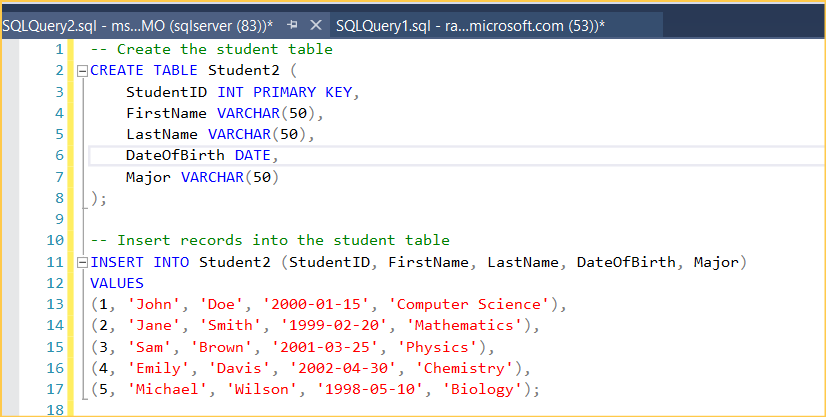
* Open the database in Azure portal
* In menu under data management click on replicas 
* Click on create replica



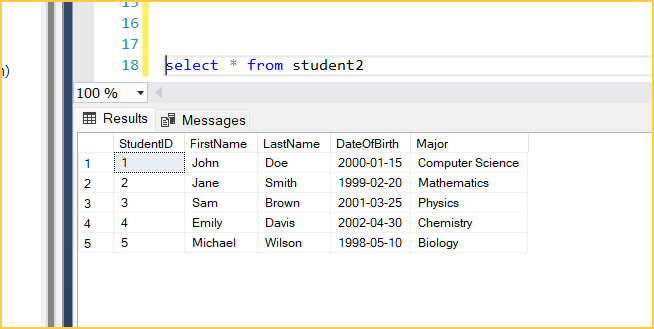




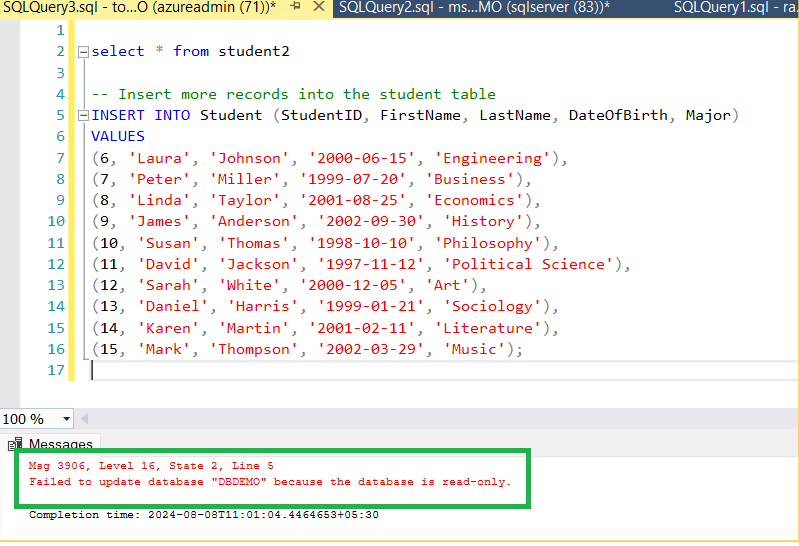
* Click on review and create
* Data replication has started from primary server to secondary server
* 
* Now Connect to the server in SSMS and verify the data
* 
* Insert some records in primary server and verify in the secondary server or create new object
* Created student2 table in primary server



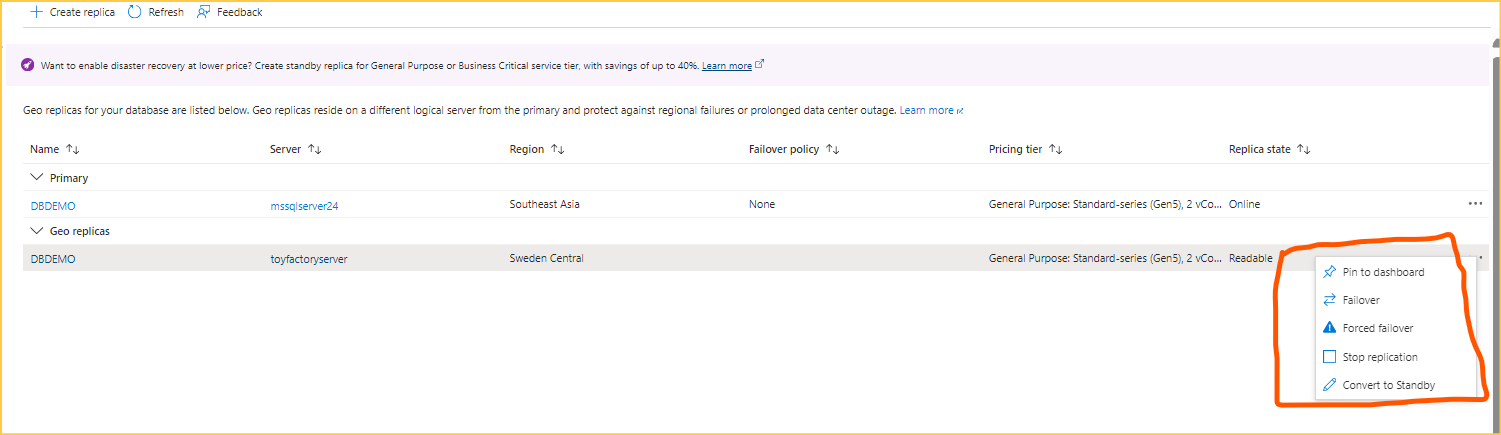
* Check the secondary server for data replicated or not



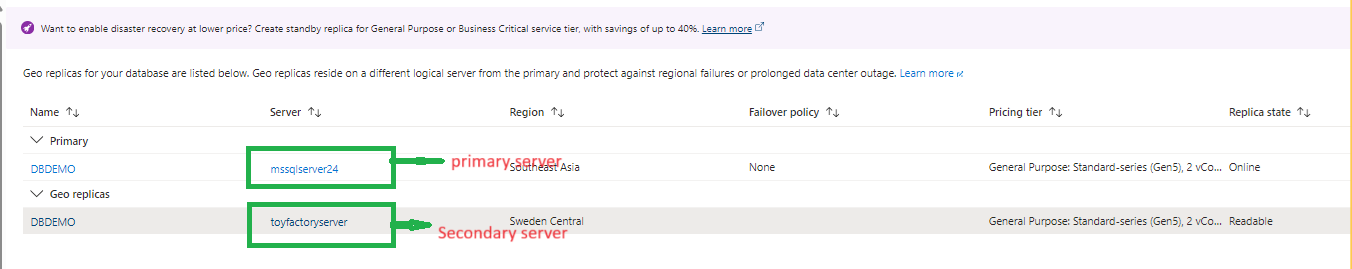
* Now the data is replicated successfully
* In secondary we can't insert and modify the data it is only for read-only purpose
* If we try to insert the data, it will throw an error



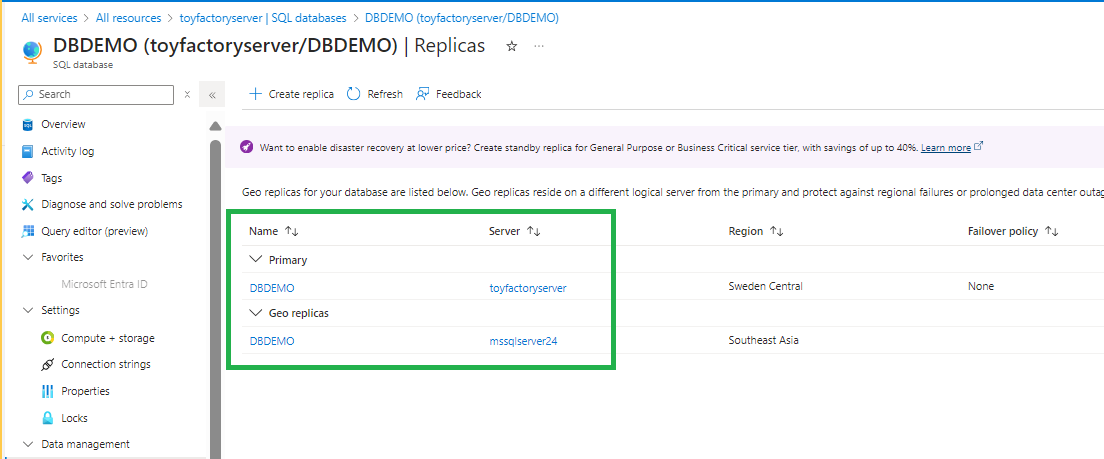
* Now we can do the manual fail over and make primary as secondary and secondary server as primary sever



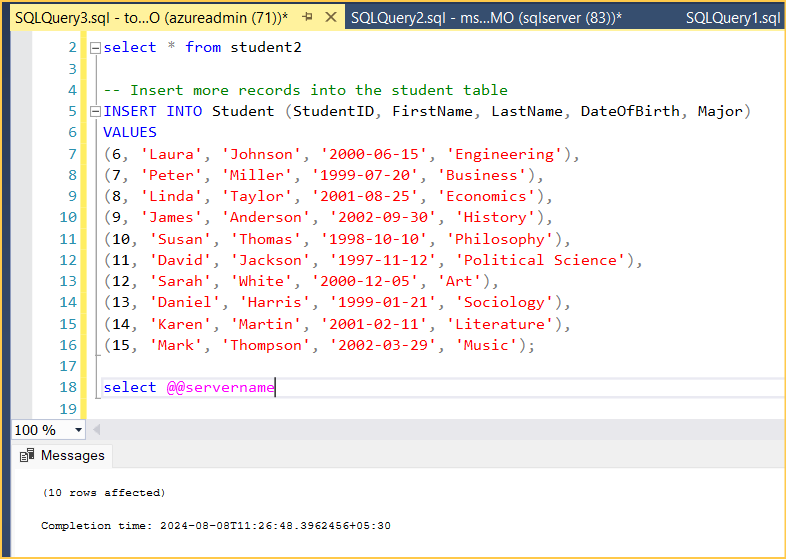
* Before Fail over primary server and secondary server



* After Fail over



* Now we can add the data in toy factory because now it become the primary server so we can insert the data



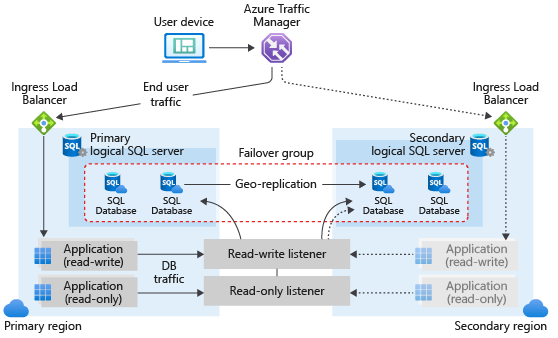
### **Failover Groups**

Failover groups provide an automatic failover mechanism for multiple databases within a group. This feature extends the capabilities of active geo-replication by:

* **Automatic Failover:** In the event of an outage, the failover group automatically switches to the secondary server without manual intervention.
* **Group Management:** Multiple databases can be managed together, ensuring consistency across all databases in the group during a failover.
* **Read-Write and Read-Only Endpoints:** The failover group provides a read-write listener endpoint that directs traffic to the primary database and a read-only listener endpoint for secondary databases.

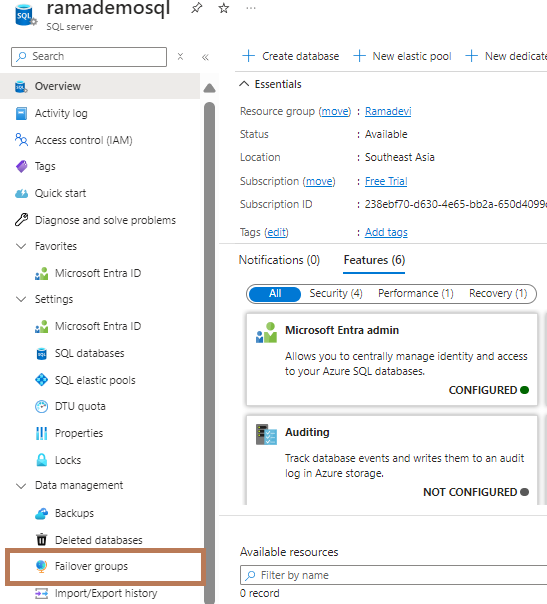
Failover groups are ideal for business continuity and disaster recovery scenarios, ensuring minimal downtime and data loss.

* Here we can configure at server level it includes number of databases
* Here we need to add the databases and delete the databases, and we can do the manual fail over also

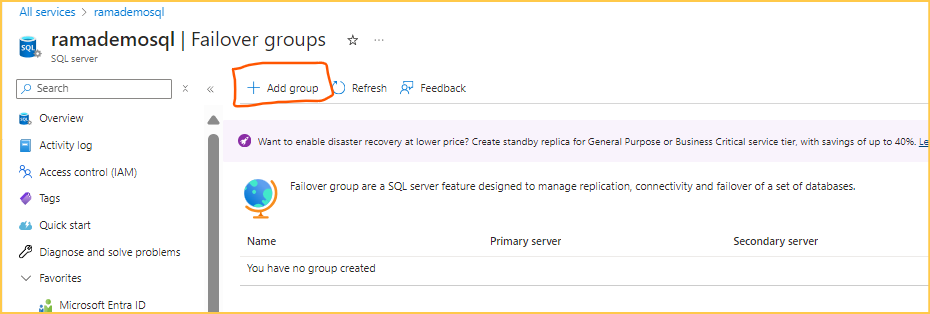


**Configuring the auto Fail over group**

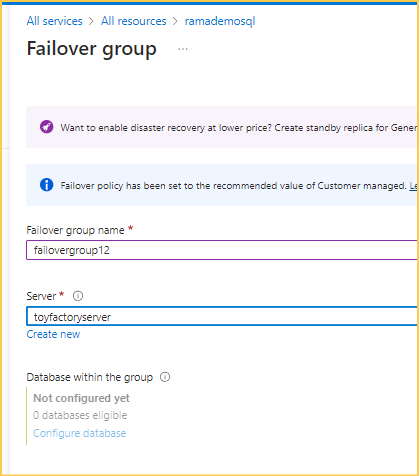
* Connect to the Azure portal
* Select the server



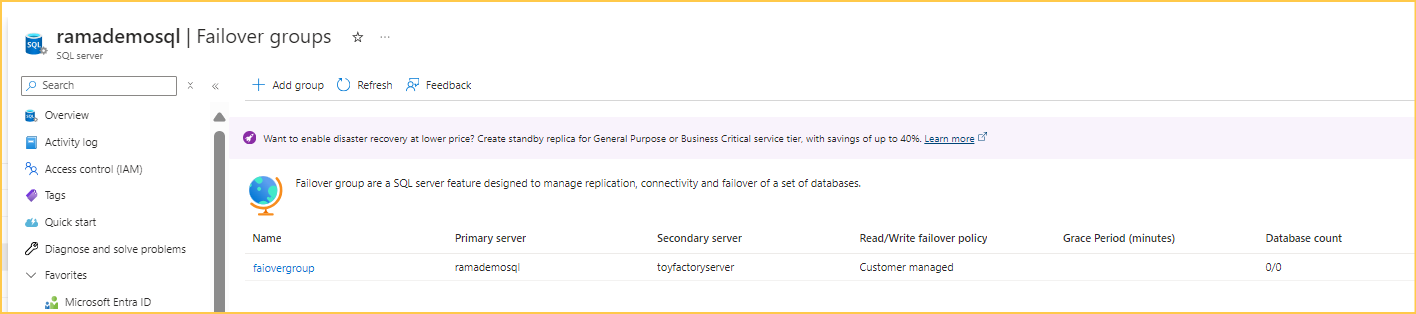
* Click on fail over groups--->add group



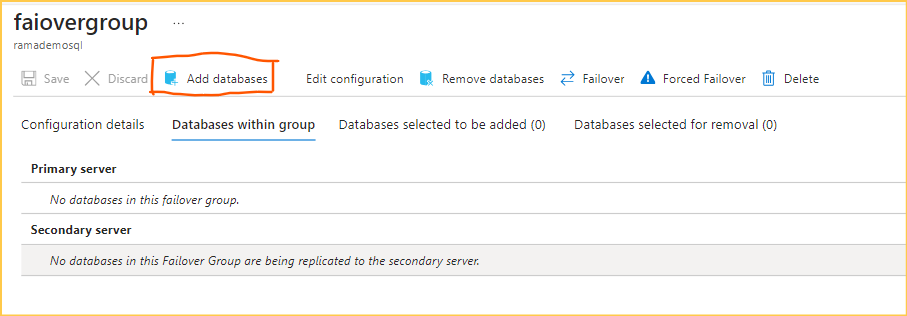
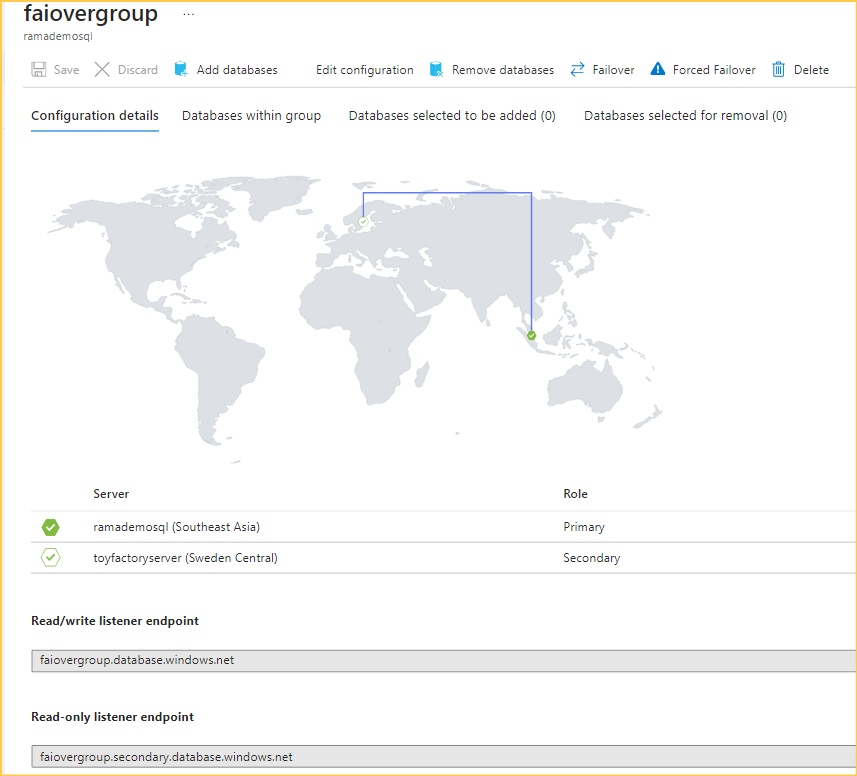
* Here adds the secondary server to the failover group



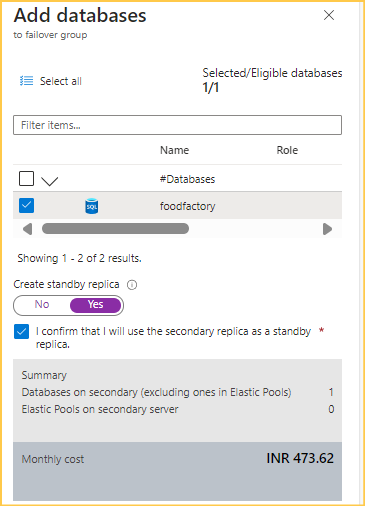
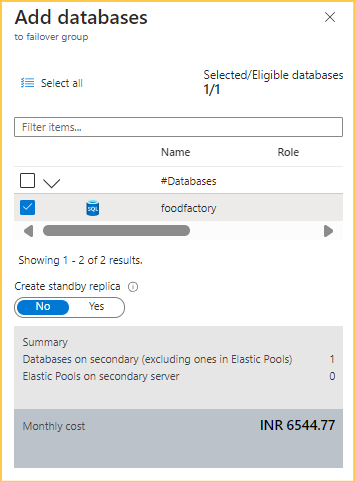
* Failover group is created with zero databases in the group



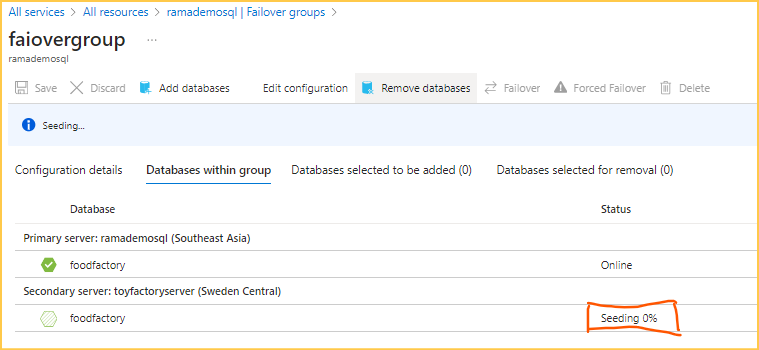
* Now, we can add databases to the failover group, not just single databases but also databases within an elastic pool.



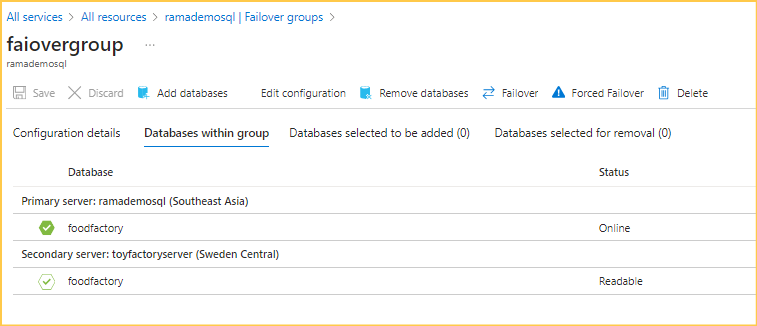
* Here we can add the secondary server as stand by or not
* The secondary server in the failover group must be in a different region than the primary server, and contain an elastic pool with the same name as the primary server.



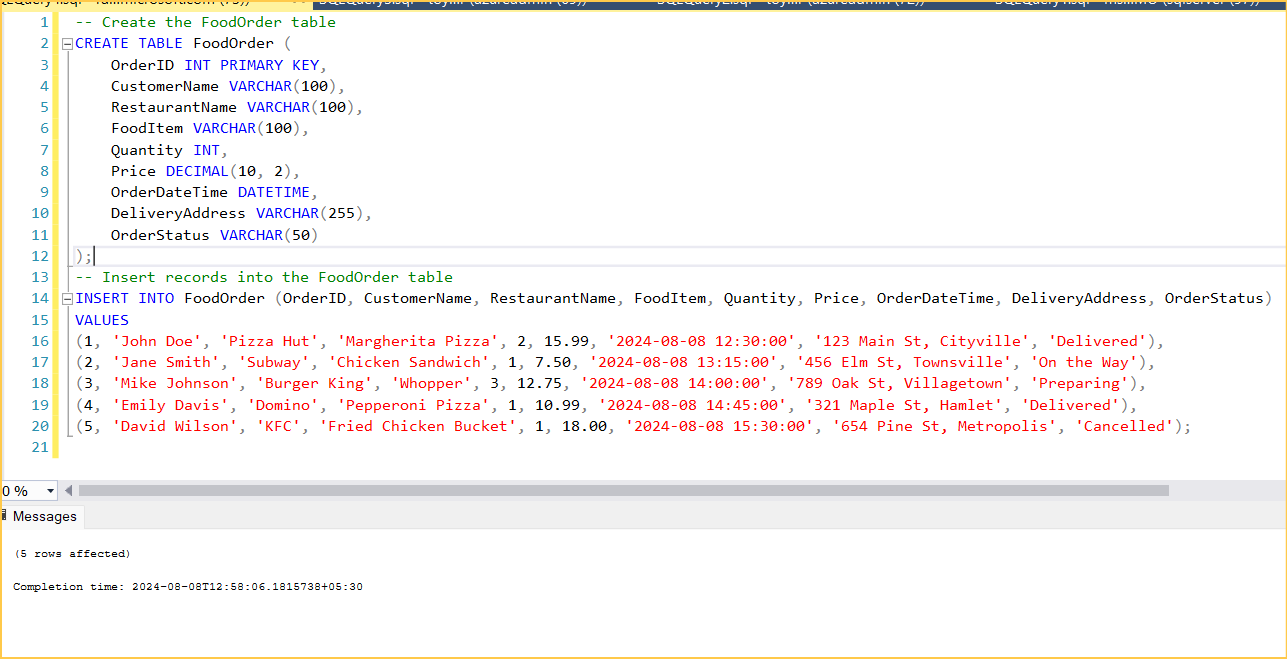
* Now the database is added to primary server in the failover group and is replicating to the second server



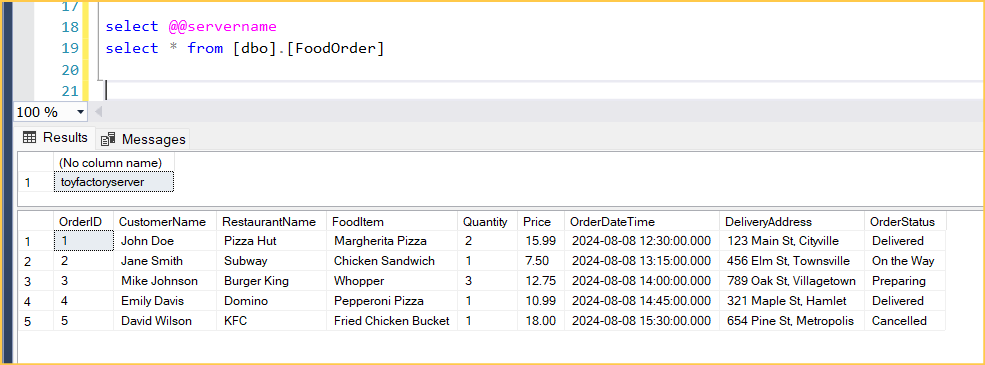
* Now the database replicated and becomes the readable server



* Here we can new databases to the primary server and delete the existing databases and we can do the failover forced failover
* Created table in primary server and inserted some records



* Now verify in the secondary server toyfactory server



Configuring the geo replication using PowerShell/Azure CLI

Configuring the failover group using PowerShell/Azure CLI

Configuring the manual failover in geo replication using PowerShell/Azure CLI

Configuring the failover and forced failover in failover group using PowerShell/Azure CLI