

Data & AI Tech Immersion Workshop – Product Review Guide and Lab Instructions

Day 1, Experience 3 - Unlocking new capabilities with friction-free migrations to Azure SQL Database Managed Instance

- [Data & AI Tech Immersion Workshop – Product Review Guide and Lab Instructions](#)
 - [Day 1, Experience 3 - Unlocking new capabilities with friction-free migrations to Azure SQL Database Managed Instance](#)
 - [Technology overview](#)
 - [Migrate your SQL Server databases without changing your apps](#)
 - [Accelerate your database migration](#)
 - [Maximize ROI by migrating to the cloud](#)
 - [Scenario overview](#)
 - [Task 1: Perform database assessments for migration](#)
 - [Task 2: Migrate the database to SQL MI](#)
 - [Task 3: Update the web application to use the new SQL MI database](#)
 - [Task 4: Enable Dynamic Data Masking](#)
 - [Task 5: Add clustered columnstore index](#)
 - [Task 6: Use online secondary for read-only queries](#)
 - [Task 7: Review Advanced Data Security Vulnerability Assessment](#)
 - [Task 8: SQL Data Discovery and Classification](#)
 - [Wrap-up](#)
 - [Additional resources and more information](#)

Technology overview

Migrate your SQL Server databases without changing your apps

[Azure SQL Database Managed Instance](#) (SQL MI) is a new deployment option of Azure SQL Database which enables the migration of existing on-premises SQL Server databases to the cloud with minimal or no application and database changes. With SQL MI, you get the broadest SQL Server engine compatibility and native virtual network (VNET) support. This option gives you the best of SQL Server, plus the operational and cost benefits of an intelligent, fully managed service. SQL MI is ideal for migrating a large number of existing SQL Server databases from on-premises or virtual machines to SQL Database.

What is SQL Database Managed Instance?

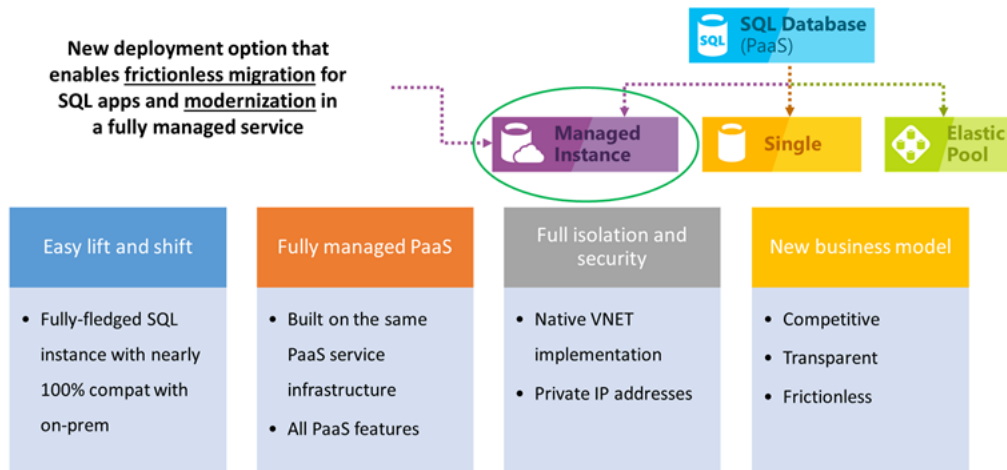


Diagram outlining the key features of managed instances.

Accelerate your database migration

Reduce the complexity of your cloud migration by using a single comprehensive service instead of multiple tools. [Azure Database Migration Service](#) is designed as a seamless, end-to-end solution for moving on-premises SQL Server databases to the cloud. Use the [Database Migration Guide](#) for recommendations, step-by-step guidance, and expert tips on your specific database migration.

Maximize ROI by migrating to the cloud

Reduce the burden of data-tier management and save time and costs by migrating workloads to the cloud. [Azure Hybrid Benefit](#) for SQL Server provides a cost-effective path for migrating hundreds or thousands of SQL Server databases with minimal effort. Use your SQL Server licenses with Software Assurance to pay a reduced rate when migrating to the cloud. Save up to 55 percent with Azure Hybrid Benefit, and up to 80 percent with [reserved capacity](#). Learn how [customers have increased productivity](#), by up to 40 percent by migrating to Azure SQL Database.

Scenario overview

ContosoAuto runs their operations and finance database, ContosoAutoDb, on an on-premises SQL Server 2008 R2 database. This system is vital to the company's daily activities and as SQL Server 2008 R2 is approaching end of support, they are looking at options for migrating this database into Azure. They have read about some of the

advanced security and performance tuning options that are available only in Azure and would prefer to migrate the database into a platform-as-a-service (PaaS) offering, if possible.

ContosoAuto is using the Service Broker feature of SQL Server within the ContosoAutoDb database. Service Broker is a feature of SQL Server used for sending and receiving guaranteed, asynchronous messages by using extensions to the Transact-SQL Data Manipulation Language (DML). This functionality is being used for several critical business processes, and they cannot afford to lose this capability when migrating their operations database to the cloud. They have also stated that, at this time, they do not have the resources to rearchitect the solution to use an alternative message broker.

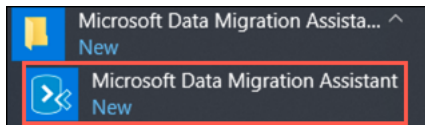
In this experience, you will use the Microsoft Data Migration Assistant (DMA) to perform assessments of feature parity and compatibility against both Azure SQL Database and Azure SQL Database Managed Instance, with the goal of migrating the ContosoAutoDb database into an Azure PaaS offering with minimal or no changes. After completing the assessments, you will perform the database migration and then update ContosoAuto's operations web application to use the new database. Once that is complete, you will review and enable some of the database features that are only available in Azure.

Task 1: Perform database assessments for migration

In this task, you will use the Microsoft [Data Migration Assistant](#) (DMA) to perform assessments on the ContosoAutoDb database. You will create two assessments, one for a migration to Azure SQL Database, and then a second for SQL MI. These assessments will provide reports about any feature parity and compatibility issues between the on-premises database and the Azure managed SQL database service options.

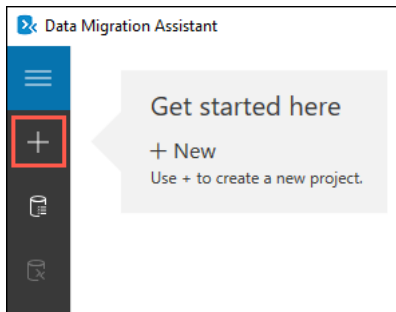
DMA helps you upgrade to a modern data platform by detecting compatibility issues that can impact database functionality in your new version of SQL Server or Azure SQL Database. DMA recommends performance and reliability improvements for your target environment and allows you to move your schema, data, and uncontained objects from your source server to your target server.

1. Launch the Microsoft Data Migration Assistant from the Windows Start menu within your lab environment.



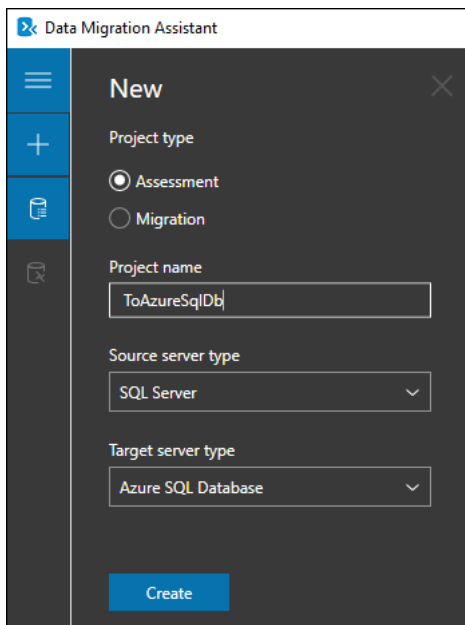
The Microsoft Data Migration Assistant is highlighted in the Windows start menu.

2. In the DMA dialog, select + from the left-hand menu to create a new project.



The new project icon is highlighted in DMA.

3. In the New project pane, set the following:
 - **Project type:** Select Assessment.
 - **Project name:** Enter ToAzureSqlDb.
 - **Source server type:** Select SQL Server.
 - **Target server type:** Select Azure SQL Database.



New project settings for doing an assessment of a migration from SQL Server to Azure SQL Database.

4. Select **Create**.

5. On the **Options** screen, ensure **Check database compatibility** and **Check feature parity** are both checked, and then select **Next**.

1 Options 2 Select sources 3 Review results

Select report type

- ☒ **Check database compatibility**
Discover migration blocking issues and deprecated features by analyzing databases you choose in your source server to be migrated to SQL Database.
- ☒ **Check feature parity**
Discover unsupported or partially-supported features and functions that your applications may rely on. Get guidance around these areas that may need some re-engineering.
- ☐ **Benefit from new features (coming soon...)**
Discover new SQL Database features that are applicable to the databases in your source once migrated to SQL database platform.

Check database compatibility and check feature parity are checked on the Options screen.

6. On the **Sources** screen, enter the following into the **Connect to a server** dialog that appears on the right-hand side:

- **Server name:** Enter the DNS name of the shared sqlServer2008R2 VM, `sqlserver2008r2.westus.cloudapp.azure.com`.
- **Authentication type:** Select **SQL Server Authentication**.
- **Username:** Enter **WorkshopUser**
- **Password:** Enter **Password.1!!**
- **Encrypt connection:** Check this box.
- **Trust server certificate:** Check this box.

Connect to a server X

Connect to a server and select sources

Server name
sqlserver2008r2.westus.cloudapp.azure.com

Authentication type
SQL Server Authentication

SQL Authentication credentials

Username
WorkshopUser

Password
.....

Connection properties

- ☒ Encrypt connection
- ☒ Trust server certificate

SQL Server permissions

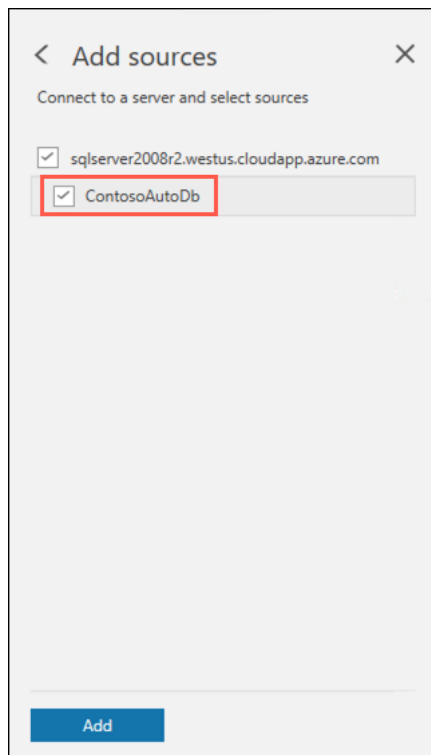
To run the selected advisor(s), credentials used to connect to a source SQL Server instance must be a member of the sysadmin server role.

Connect

In the Connect to a server dialog, the values specified above are entered into the appropriate fields.

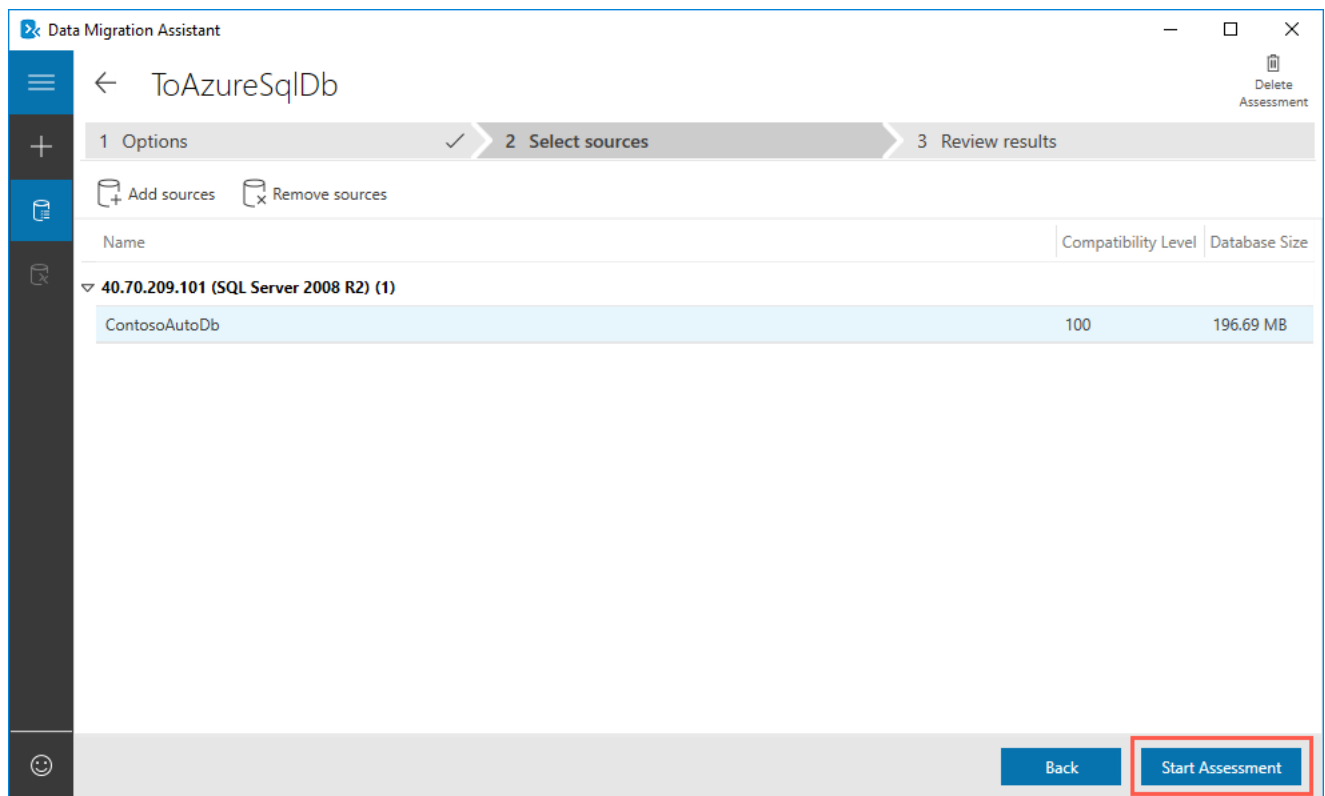
7. Select **Connect**.

8. On the **Add sources** dialog that appears next, check the box for **ContosoAutoDb** and select **Add**.



The ContosoAutoDb box is checked on the Add sources dialog.

9. Select **Start Assessment**.



Start assessment

10. Review the assessment of ability to migrate to Azure SQL Database.

Target Platform
Azure SQL Database
52.167.121.117 / SQL Server 2008 R2

Feature parity (5)

Recommendation Databases

Service Broker feature is not supported in Azure SQL Database

▼ **Unsupported features (4)**

Cross-database references not suppor...	1
Service Broker feature is not supporte...	1
Azure SQL Database doesn't support...	N/A
SQL Server Reporting Services is not s...	N/A

▼ **Partially-supported features (1)**

Full-text search partially supported in...	1
--	---

Details

Impact
SQL Server Service Broker provides native support for messaging and queuing applications in the SQL Server Database Engine.

Recommendation
Service Broker feature is not supported in Azure SQL Database. You need to disable Service Broker using the following command before migrating this database to Azure:
`ALTER DATABASE [database_name] SET DISABLE_BROKER;`

Databases

Type	Name
Database	ContosoAutoDb

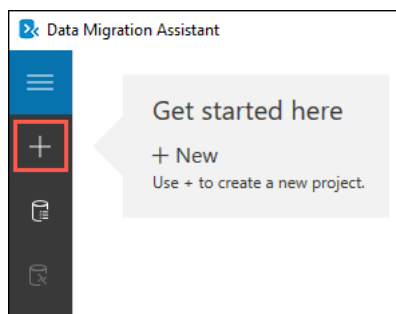
Database details

Type: Database
Name: ContosoAutoDb
This database has Service Broker enabled.

For a target platform of Azure SQL Database, feature parity shows two features which are not supported in Azure SQL Database. The Service broker feature is selected on the left and on the right Service Broker feature is not supported in Azure SQL Database is highlighted.

The DMA assessment for a migrating the ContosoAutoDb database to a target platform of Azure SQL Database shows two features in use which are not supported in Azure SQL Database. These features, cross-database references and Service broker, will prevent ContosoAuto from being able to migrate to the Azure SQL Database PaaS offering.

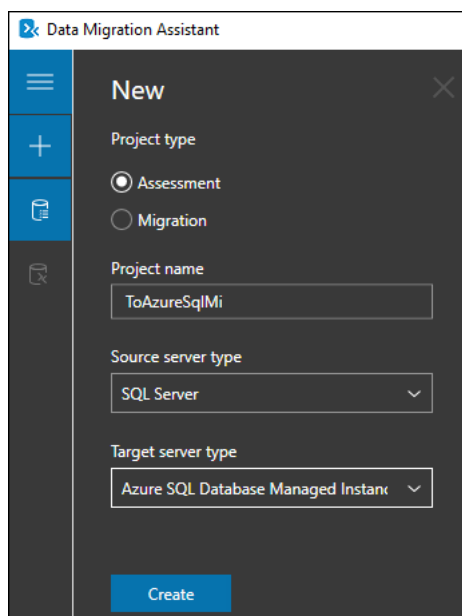
11. With one PaaS offering ruled out due to feature parity, you will now perform a second assessment, this time for a migration to Azure SQL Database Managed Instance (SQL MI). To get started, select + on the left-hand menu in DMA to create another new project.



The new project icon is highlighted in DMA.

12. In the New project pane, set the following:

- **Project type:** Select Assessment.
- **Project name:** Enter ToAzureSqlMi.
- **Source server type:** Select SQL Server.
- **Target server type:** Select Azure SQL Database Managed Instance.



New project settings for doing an assessment of a migration from SQL Server to Azure SQL Database Managed Instance.

13. Select **Create**.

14. On the **Options** screen, ensure **Check database compatibility** and **Check feature parity** are both checked, and then select **Next**.

1 Options 2 Select sources 3 Review results

Select report type

- ☒ **Check database compatibility**
Discover migration blocking issues and deprecated features by analyzing databases you choose in your source server to be migrated to SQL Database.
- ☒ **Check feature parity**
Discover unsupported or partially-supported features and functions that your applications may rely on. Get guidance around these areas that may need some re-engineering.
- ☐ **Benefit from new features (coming soon...)**
Discover new SQL Database features that are applicable to the databases in your source once migrated to SQL database platform.

Check database compatibility and check feature parity are checked on the Options screen.

15. On the **Sources** screen, enter the following into the **Connect to a server** dialog that appears on the right-hand side:

- **Server name:** Enter the DNS name of the shared sqlServer2008R2 VM, `sqlserver2008r2.westus.cloudapp.azure.com`.
- **Authentication type:** Select **SQL Server Authentication**.
- **Username:** Enter **WorkshopUser**
- **Password:** Enter **Password.1!!**
- **Encrypt connection:** Check this box.
- **Trust server certificate:** Check this box.

Connect to a server

Connect to a server and select sources

Server name
sqlserver2008r2.westus.cloudapp.azure.com

Authentication type
SQL Server Authentication

SQL Authentication credentials

Username
WorkshopUser

Password
Password.1!!

Connection properties

- ☒ Encrypt connection
- ☒ Trust server certificate

SQL Server permissions

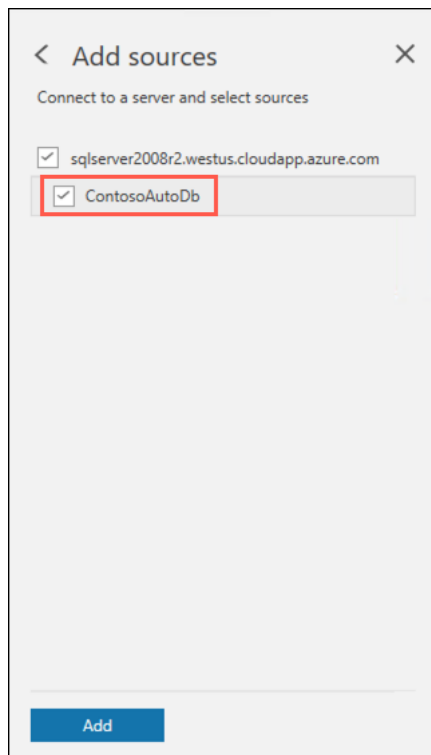
To run the selected advisor(s), credentials used to connect to a source SQL Server instance must be a member of the sysadmin server role.

Connect

In the Connect to a server dialog, the values specified above are entered into the appropriate fields.

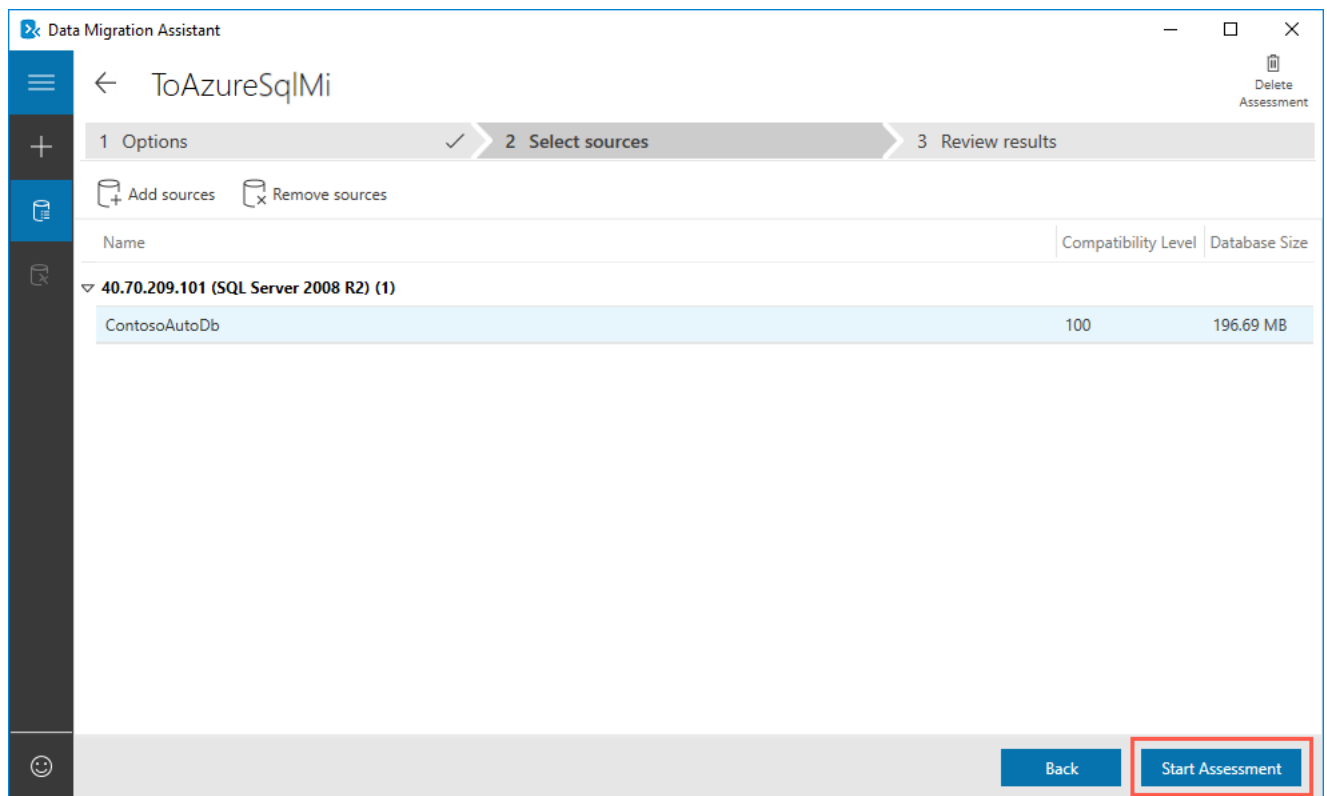
16. Select **Connect**.

17. On the **Add sources** dialog that appears next, check the box for **ContosoAutoDb** and select **Add**.



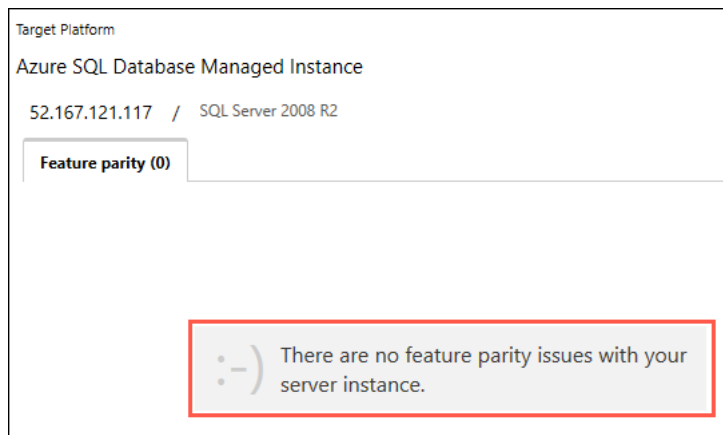
The ContosoAutoDb box is checked on the Add sources dialog.

18. Select **Start Assessment**.



Start assessment

19. Review the assessment of ability to migrate to Azure SQL Database Managed Instance.



For a target platform of Azure SQL Database Managed Instance, there are no feature parity issues found.

The assessment report for a migrating the ContosoAutoDb database to a target platform of Azure SQL Database Managed Instance shows no feature parity. The database, including the cross-database references and Service broker features, can be migrated as is, providing the opportunity for ContosoAuto to have a fully managed PaaS database instance running in Azure. Previously, their options for migrating a database using features, such as Service Broker, incompatible with Azure SQL Database, were to deploy the database to a virtual machine running in Azure (IaaS) or modify their database and applications to not use the unsupported features. The introduction of Azure SQL MI, however, provides the ability to migrate databases into a managed Azure SQL database with near 100% compatibility, including the features that prevented them from using Azure SQL Database.

Task 2: Migrate the database to SQL MI

In this task, you will migrate the ContosoAutoDb database from the on-premises SQL 2008 R2 database to SQL MI, targeting the [Business Critical service tier](#).

The Business Critical service tier is designed for business applications with the highest performance and high-availability (HA) requirements.

To migrate the ContosoAutoDb database from SQL 2008 R2 to SQL MI you will use a backup of the database stored in an Azure Blob storage account. RESTORE of native backups (.bak files) taken from SQL Server on-premises or SQL Server on Virtual Machines, available on Azure Storage, is one of key capabilities of the managed instance deployment option that enables quick and easy offline database migration. The following diagram provides a high-level overview of the process:

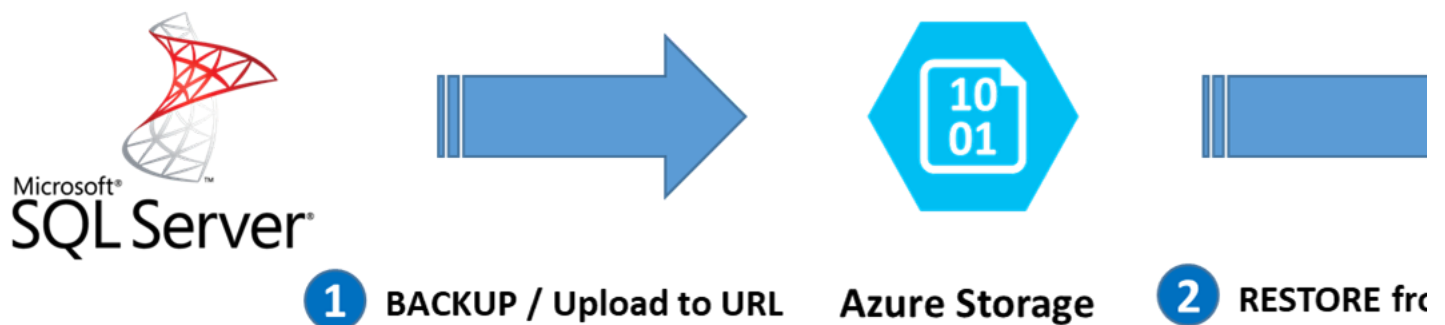
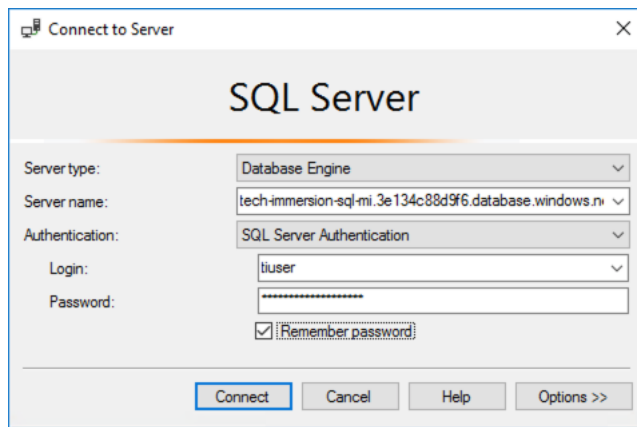


Diagram of the native RESTORE from URL capability.

1. Open **SQL Server Management Studio 17** (SSMS) from the Microsoft SQL Server Tools 17 folder in the Windows Start menu and connect to your SQL MI database. On the connection dialog enter the following:
 - **Server name:** Enter the name of the shared SQL MI server, `tech-immersion-sqlmi-shared.521f7783692d.database.windows.net`.
 - **Authentication:** Select **SQL Server Authentication**.
 - **Login:** Enter `tiuser`
 - **Password:** Enter `Password.1234567890`
 - Check the **Remember password** box.



Connection dialog for SSMS.

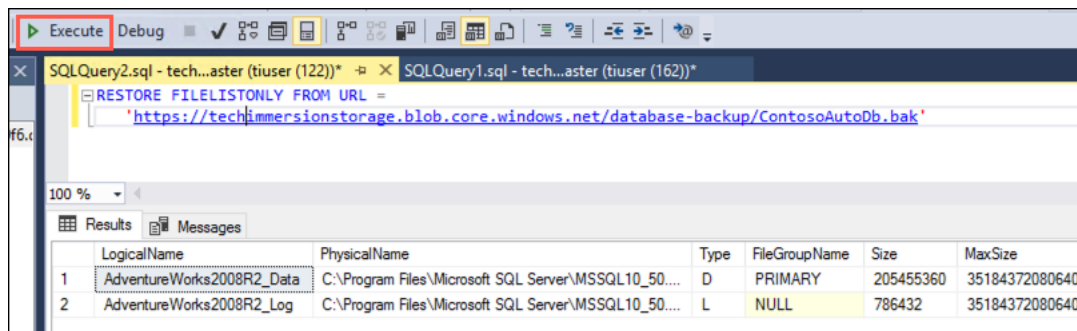
2. Select **Connect**.

3. To perform the RESTORE process, credentials for a pre-configured storage account and SAS token have already been added to the Managed Instance using the [create a credential](#) method. This process essentially creates a connection from your SQL MI database to the Blob storage account, allowing you to access files stored in the target container, database-backup. Because this is a shared SQL MI, only one credential is needed for all attendees. If you are curious, you would create it with a SQL statement similar to the below. **NOTE: You should NOT run the following SQL, it is just for illustration.**

```
CREATE CREDENTIAL [https://techimmersion.blob.core.windows.net/labfiles/data/3]
WITH IDENTITY = 'SHARED ACCESS SIGNATURE',
SECRET = 'sv=2018-03-28&ss=bfqt&srt=sco&sp=rwdlacup&se=2099-03-22T05:33:07Z&st=2019-03-21T21:33:07Z&spr=https&sig=xCq7hZfgdtM1UaN9%2FToz04GT5d5RsKae'
```

4. You can verify the credential's access to the Blob storage account by selecting **New Query** from the SSMS toolbar. Paste the following SQL script to get a backup file list from the storage account into the new query window and select **Execute** from the toolbar.

```
RESTORE FILELISTONLY FROM URL = 'https://techimmersion.blob.core.windows.net/labfiles/data/3/ContosoAutoDb.bak'
```



Script to list files in a backup file in Blob storage.

5. You are now ready to restore the ContosoAutoDb database in SQL MI. In this step, you will be creating a new database on the Managed Instance. Select **New Query** on the SSMS toolbar again, then paste the following SQL script into the new query window. **Replace the xxxxx value** with the unique identifier assigned to your for this workshop. The database name in the query should look something like ContosoAutoDb-01234.

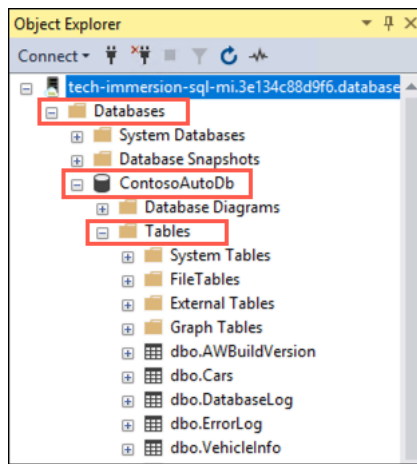
```
RESTORE DATABASE [ContosoAutoDb-XXXXX] FROM URL = 'https://techimmersion.blob.core.windows.net/labfiles/data/3/ContosoAutoDb.bak'
```

NOTE: You may notice multiple databases in on the Managed Instance. This is because the SQL MI is a shared resource for all workshop attendees, so make sure you use your assigned unique ID when restoring and accessing the database.

6. Select **Execute** on the SSMS toolbar.

7. The restore will take 1 - 2 minutes to complete. You will receive a "Commands completed successfully" message when it is done.

8. When the restore completes, expand **Databases** in the Object Explorer, and then expand **ContosoAutoDb-XXXXX** (where XXXXX is the unique identifier assigned to you for this workshop) and **Tables**. You will see that the tables are all listed, and the SQL Server 2008 R2 database has been successfully restored into SQL MI.



The Object Explorer is displayed with Databases, ContosoAutoDb, and Tables expanded.

NOTE: Your database name will differ from the above screen shot, in that it will contain the unique identifier assigned to you for this workshop, such as ContosoAutoDb-01234. The SQL Managed Instance is shared for all workshop participants, so you may also see databases for other participants.

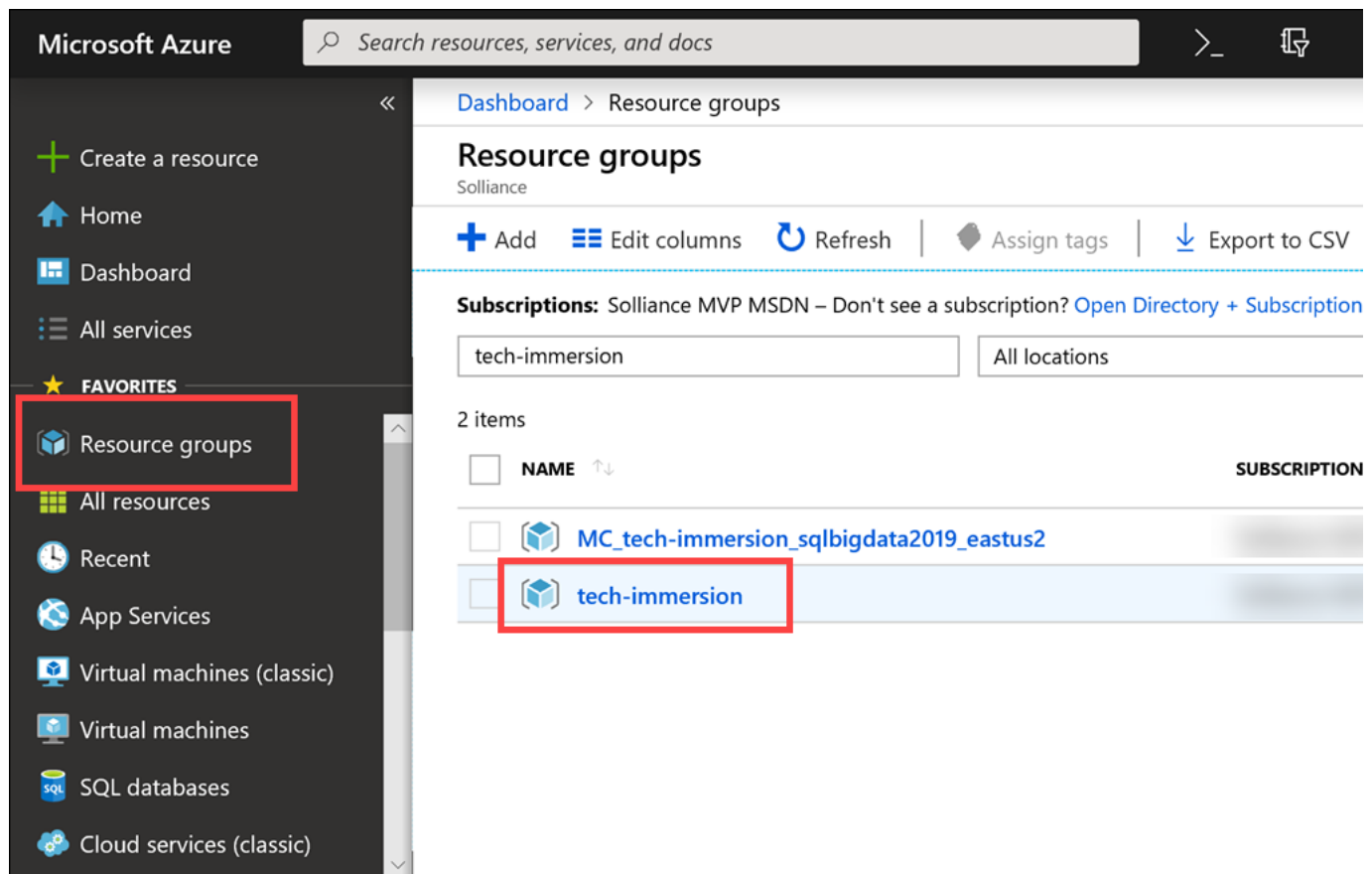
Task 3: Update the web application to use the new SQL MI database

With the ContosoAutoDb database now running on SQL MI in Azure, the next step is to make the required modifications to the ContosoAuto operations web application. The operations web app is currently running an [Azure App Service Environment](#), which was provisioned in the same virtual network as the SQL Managed Instance.

SQL Managed Instance has private IP address in its own VNet, so to connect an application you need to configure access to the VNet where Managed Instance is deployed. To learn more, read [Connect your application to Azure SQL Database Managed Instance](#).

In this task, you will make updates to the ContosoAuto operations web application to enable it to connect to and utilize the SQL MI database.

1. Using a web browser, navigate to the [Azure portal](#), select **Resource groups** from the left-hand menu, and then select the resource group named **tech-immersion-XXXXX** (where XXXXX is the unique identifier assigned to you for this workshop).



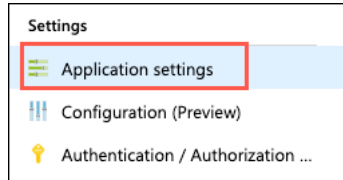
The tech-immersion resource group is selected.

2. Select the **Tech Immersion Web App Service** ending with your unique identifier (e.g., techimmersionwebapp01234) from the list of resources.

<input type="checkbox"/>	 tech-immersion-vnet	Virtual network
<input type="checkbox"/>	 tech-immersion-web	App Service plan
<input type="checkbox"/>	 tech-immersion-web	App Service
<input type="checkbox"/>	 VirtualClusterb52af2c3-7c09-4c67-8860-d597b67738ca	Virtual cluster
<input type="checkbox"/>	 vnet-tech-immersion-sql-mi	Virtual network

The App Service resource is selected from the list of resources in the tech-immersion resource group.

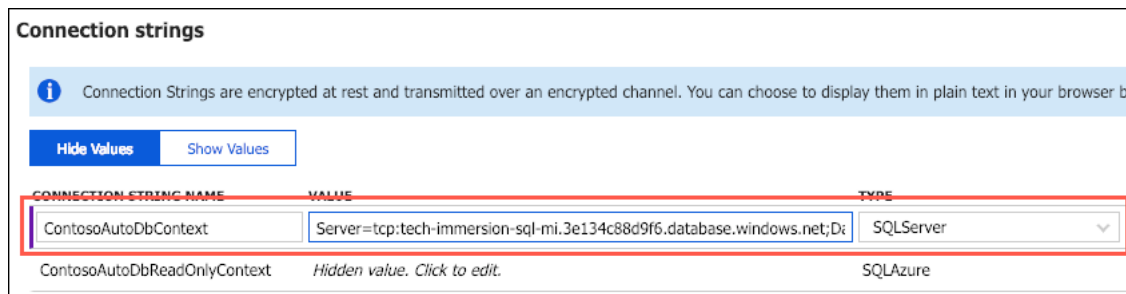
- On the App Service blade, select **Application settings** under Settings on the left-hand side.



The Application settings item is selected under Settings.


- On the Application settings blade, scroll down and locate the **Connection strings** section. Paste the connection string value below into the value for the ContosoAutoDbContext connection string. Also be sure to set the Type drop-down to SQLServer.

Server=tcp:tech-immersion-sqlmi-shared.521f7783692d.database.windows.net,1433;Persist Security Info=False;Database=ContosoAutoDb;User ID=tiuser;Passw



The copied SQL MI connection string is pasted into the value for the ContosoAutoDbContext connection string.

- Repeat the previous step, this time pasting the same connection string into the ContosoAutoDbReadOnlyContext connection string.



Read-only connection string.

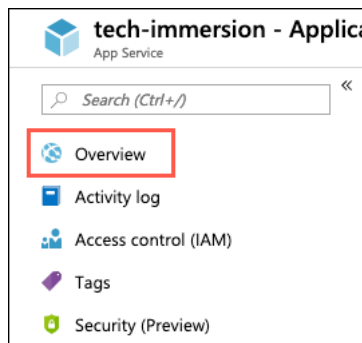
- Select **Save** at the top of the Application settings blade.



The save button on the Application settings blade is highlighted.

NOTE: The astute reader may have noticed in the above steps that the Web App continues to query a database called ContosoAutoDb and not the database that was just restored. This is intended only to shorten the application configuration steps. Rest assured that the changes you made to the database connection string would enable your application to reach any of the databases loaded on to the SQL Server Managed Instance.

- Select **Overview** to the left of the Application settings blade to return to the overview blade of your App Service.



Overview is highlighted on the left-hand menu for App Service

- On the overview blade, click the **URL** of your App service to launch the website. This will open the URL in a browser window.



The App service URL is highlighted.

- Verify that the web site and data is loaded correctly. The page should look similar to the following:

ContosoAuto					
Products					
Name	ProductNumber	Color	SafetyStockLevel	ReorderPoint	
Adjustable Race	AR-5381		1000	750	
Bearing Ball	BA-8327		1000	750	
BB Ball Bearing	BE-2349		800	600	
Headset Ball Bearings	BE-2908		800	600	
Blade	BL-2036		800	600	

Screenshot of the ContosoAuto Operations Web App.

That is it. You were able to successfully connect your application to the new SQL MI database by simply updating the application's connection string. No code changes or other updates are needed!

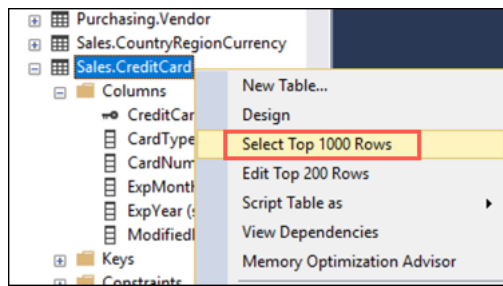
Task 4: Enable Dynamic Data Masking

[Dynamic Data Masking](#) (DDM) limits sensitive data exposure by masking it to non-privileged users. This feature helps prevent unauthorized access to sensitive data by enabling customers to designate how much of the sensitive data to reveal with minimal impact on the application layer. It's a policy-based security feature that hides the sensitive data in the result set of a query over designated database fields, while the data in the database is not changed.

For example, a service representative at a call center may identify callers by several digits of their credit card number, but those data items should not be fully exposed to the service representative. A masking rule can be defined that masks all but the last four digits of any credit card number in the result set of any query. As another example, an appropriate data mask can be defined to protect personally identifiable information (PII) data, so that a developer can query production environments for troubleshooting purposes without violating compliance regulations.

In this task, you will enable DDM on the CardNumber field in the CreditCard table in the ContosoAutoDb database, to prevent queries against that table from returning the full credit card number.

- Return to the SQL Server Management Studio (SSMS) window you opened previously.
- Expand **Tables** under the **ContosoAutoDb-XXXXXX** (where XXXXXX is the unique identifier assigned to you for this workshop) and locate the **Sales.CreditCard** table. Expand the table columns and observe that there is a column named **CardNumber**. Right-click the table, and choose **Select Top 1000 Rows** from the context menu.



The Select Top 1000 Rows item is highlighted in the context menu for the Sales.CreditCard table.

3. In the query window that opens, review the Results, including the CardNumber field. Notice it is displayed in plain text, making the data available to anyone with access to query the database.

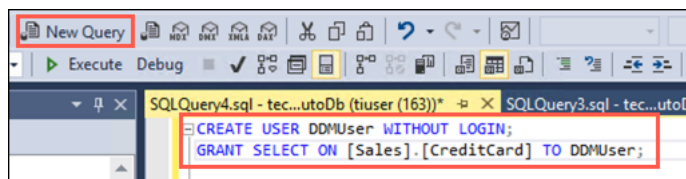
	CreditCardID	CardType	CardNumber	ExpMonth	ExpYear	ModifiedDate
1	1	SuperiorCard	33332664695310	11	2006	2007-08-30 00:00:00.000
2	2	Distinguish	55552127249722	8	2005	2008-01-06 00:00:00.000
3	3	ColonialVoice	77778344838353	7	2005	2008-02-15 00:00:00.000
4	4	ColonialVoice	77774915718248	7	2006	2007-06-21 00:00:00.000

Plain text credit card numbers are highlighted in the query results.

4. So we can test the mask being applied to the CardNumber field, you will first create a user in the database that will be used for testing the masked field. In SSMS, select **New Query** and paste the following SQL script into the new query window, replacing xxxxx with your unique ID (if the DDMUser already exists in your environment, the following script will show an error, but you are free to proceed to the next step):

```
USE [ContosoAutoDb-XXXXX];
GO

CREATE USER DDMUser WITHOUT LOGIN;
GRANT SELECT ON [Sales].[CreditCard] TO DDMUser;
```



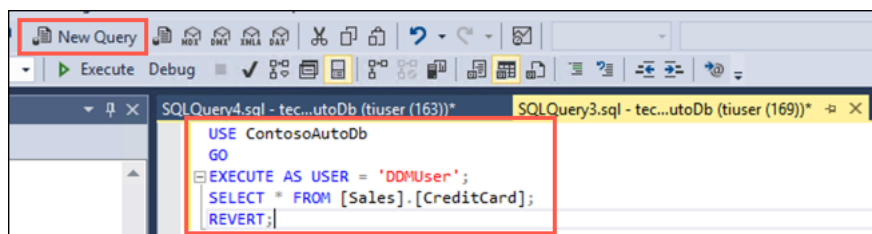
A Create User query is pasted into the new query window.

The SQL script above create a new user in the database named DDMUser, and grants that user SELECT rights on the Sales.CreditCard table.

5. Select **Execute** from the SSMS toolbar to run the query. You will get a message that the commands completed successfully in the Messages pane.
6. With the new user created, let's run a quick query to verify the results. Select **New Query** again, and paste the following into the new query window. Replace xxxxx in the USE statement to include the unique identifier of your database which will be ContosoAutoDb-xxxxx (e.g., ContosoAutoDb-01234).

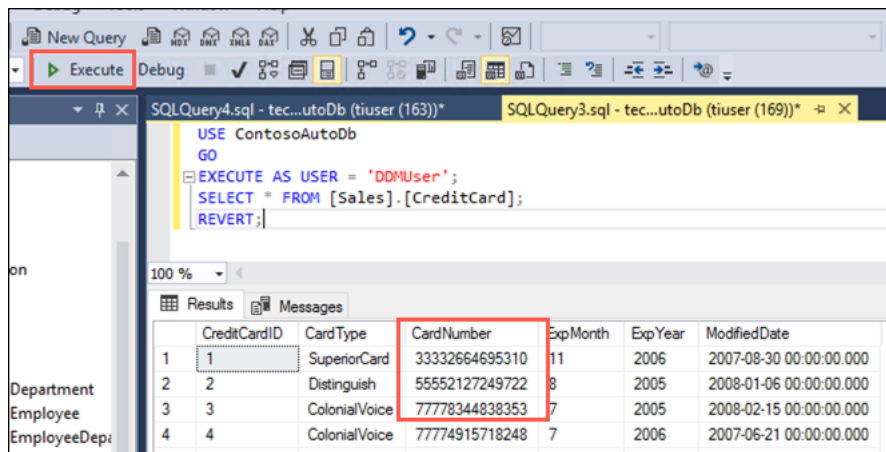
```
USE [ContosoAutoDb-XXXXX];
GO

EXECUTE AS USER = 'DDMUser';
SELECT * FROM [Sales].[CreditCard];
REVERT;
```



The SQL query above is pasted into the new query window in SSMS.

7. Select **Execute** from the toolbar, and examine the Results pane. Notice the credit card number, as above, is visible in clear text.

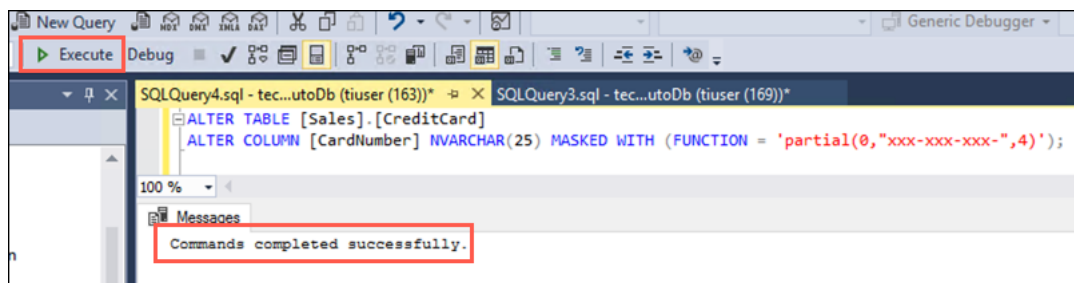


The credit card number is unmasked in the query results.

8. You will now apply DDM on the CardNumber field to prevent it from being viewed in query results. Select **New Query** from the SSMS toolbar and paste the following query into the query window to apply a mask to the CardNumber field, replacing XXXXX with your unique ID. Select **Execute** to run the query.

```
USE [ContosoAutoDb-XXXXX];
GO
```

```
ALTER TABLE [Sales].[CreditCard]
ALTER COLUMN [CardNumber] NVARCHAR(25) MASKED WITH (FUNCTION = 'partial(0,"xxx-xxx-xxx-",4)')
```

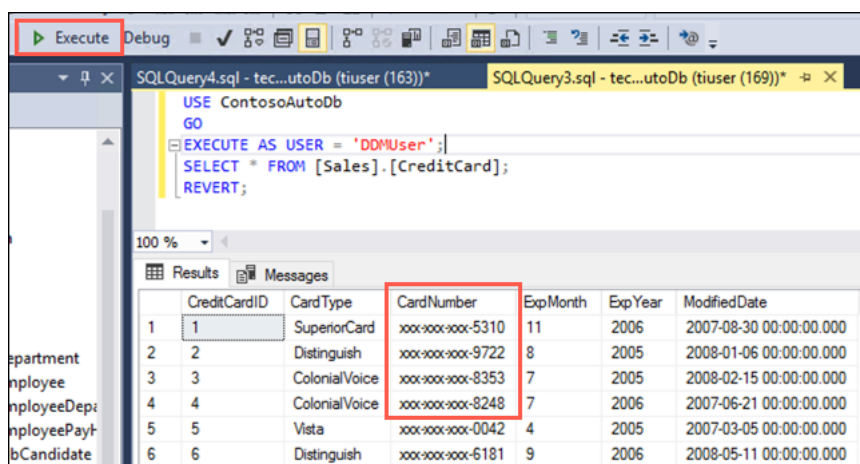


The SQL script above is pasted into the new query window. The Execute button is highlighted and a success message is displayed in the Messages pane.

9. Run the SELECT query you opened in step 7 above again, and observe the results, specifically inspect the output in the CardNumber field. For reference the query is below. You replaced XXXXX in the USE statement to include the unique identifier of your database which will be ContosoAutoDb-XXXXX (e.g., ContosoAutoDb-01234).

```
USE [ContosoAutoDb-XXXXX];
GO
```

```
EXECUTE AS USER = 'DDMUser';
SELECT * FROM [Sales].[CreditCard];
REVERT;
```



The credit card number is masked in the query results.

The CardNumber is now displayed using the mask applied to it, so only the last four digits of the card number are visible. Dynamic Data Masking is a powerful feature that enables you to prevent unauthorized users from viewing sensitive or restricted information. It's a policy-based security feature that hides the sensitive data in the result set of a query over designated database fields, while the data in the database is not changed.

Task 5: Add clustered columnstore index

ContosoAuto is looking to take advantage of some of the performance improvement features available in Azure SQL MI. In particular, they are interested in optimizing performance by using [In-Memory technologies](#).

In this task, you will create a new table based on the existing [Sales].[SalesOrderDetail] table and apply a [ColumnStore index](#).

Columnstore indexes are the standard for storing and querying large data warehousing fact tables. This index uses column-based data storage and query processing to achieve gains up to **10 times the query performance** in your data warehouse over traditional row-oriented storage. You can also achieve gains up to **10 times the data compression** over the uncompressed data size.

1. In SSMS, ensure you are connected to the Azure SQL Database Managed Instance.
2. Open a new query window by selecting **New Query** from the toolbar.



The New Query icon is highlighted on the SSMS toolbar.

3. Copy the script below, and paste it into the query window. Replace xxxxx in the USE statement to include the unique identifier of your database which will be ContosoAutoDb-xxxxx (e.g., ContosoAutoDb-01234).

```
USE [ContosoAutoDb-XXXXX];
GO

SELECT *
INTO [Sales].[ColumnStore_SalesOrderDetail]
FROM [Sales].[SalesOrderDetail]
GO
```

4. Select **Execute** on the toolbar to run the query, and create a new table named [Sales].[ColumnStore_SalesOrderDetail], populated with data from the [Sales].[SalesOrderDetail] table.



The Execute icon is highlighted on the SSMS toolbar.

5. Select **New Query** in the toolbar again, and paste the following query into the new query window. The query contains multiple parts; one to get the size of the ColumnStore_SalesOrderDetail table, a second to create a clustered ColumnStore index on the [Sales].[ColumnStore_SalesOrderDetail] table, and then the size query is repeated to get the size after adding the clustered ColumnStore index. Replace xxxxx in the USE statement to include the unique identifier of your database which will be ContosoAutoDb-xxxxx (e.g., ContosoAutoDb-01234).

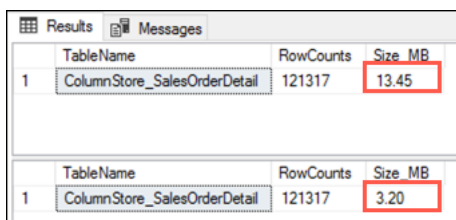
```
USE [ContosoAutoDb-XXXXX];
GO

-- Get the Size of the [Sales].[ColumnStore_SalesOrderDetail] table
SELECT
t.Name AS TableName,
p.rows AS RowCounts,
CAST(ROUND((SUM(a.total_pages) / 128.00), 2) AS NUMERIC(36, 2)) AS Size_MB
FROM sys.tables t
INNER JOIN sys.indexes i ON t.OBJECT_ID = i.object_id
INNER JOIN sys.partitions p ON i.object_id = p.OBJECT_ID AND i.index_id = p.index_id
INNER JOIN sys.allocation_units a ON p.partition_id = a.container_id
WHERE t.Name = 'ColumnStore_SalesOrderDetail'
GROUP BY t.Name, p.Rows
GO

-- Create a clustered columnstore index on the [Sales].[ColumnStore_SalesOrderDetail] table
CREATE CLUSTERED COLUMNSTORE INDEX [cci_SalesOrderDetail]
ON [Sales].[ColumnStore_SalesOrderDetail]
GO

-- Get the Size of the [Sales].[ColumnStore_SalesOrderDetail] table
SELECT
t.Name AS TableName,
p.rows AS RowCounts,
CAST(ROUND((SUM(a.total_pages) / 128.00), 2) AS NUMERIC(36, 2)) AS Size_MB
FROM sys.tables t
INNER JOIN sys.indexes i ON t.OBJECT_ID = i.object_id
INNER JOIN sys.partitions p ON i.object_id = p.OBJECT_ID AND i.index_id = p.index_id
INNER JOIN sys.allocation_units a ON p.partition_id = a.container_id
WHERE t.Name = 'ColumnStore_SalesOrderDetail'
GROUP BY t.Name, p.Rows
GO
```

6. Select **Execute** on the toolbar to run the query.
7. In the query results, observe the Size_MB value of the table before and after the creation of the clustered ColumnStore index. The first value is the size before the index was created, and the second value is the size after the ColumnStore index was created.



	TableName	RowCounts	Size_MB
1	ColumnStore_SalesOrderDetail	121317	13.45

	TableName	RowCounts	Size_MB
1	ColumnStore_SalesOrderDetail	121317	3.20

The SSMS results pane is displayed, with the size of the [Sales].[ColumnStore_SalesOrderDetail] table highlighted both before and after the creation of the clustered ColumnStore index.

8. Create another new query window by selecting **New Query** from the toolbar, and then select **Include Actual Execution Plan** by selecting its button in the toolbar.



The Include Actual Execution Plan icon is highlighted on the New Query the toolbar.

- Paste the queries below into the new query window, replace xxxxx with your unique ID, and select **Execute** on the toolbar:

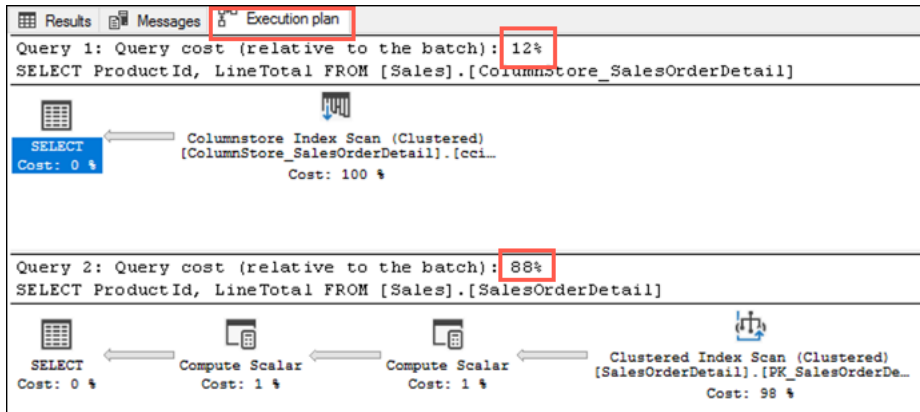
```
USE [ContosoAutoDb-XXXXX];
GO

SELECT ProductId, LineTotal
FROM [Sales].[ColumnStore_SalesOrderDetail]

SELECT ProductId, LineTotal
FROM [Sales].[SalesOrderDetail]
```

Running queries against both the SalesOrderDetail and ColumnStore_SalesOrderDetail will allow you to compare the query execution plans between tables with and without a columnstore index.

- In the Results pane, select the **Execution Plan** tab. Check the *Query cost (relative to the batch)* percentage value of the two queries and compare them.



The Execution Plan tab is highlighted in the Results pane, 12% is highlighted for Query 1, and 88% is highlighted for Query 2.

From the query cost, it is clear the query against the table with the columnstore index was more performant. Using a columnstore index, queries get an order of magnitude better performance boost with *BatchMode* processing, a unique value proposition in SQL Server. The basic idea of batch mode processing is to process multiple values, hence the term 'batch', together instead of one value at a time. Batch mode processing is perfectly suited for analytics where a large number of rows need to be processed, for example, to compute aggregates or apply filter predicates.

- Run the same queries again, but this time set statistics IO on in the query by adding the following to the top of the query window:

```
SET STATISTICS IO ON
GO
```

- Your final query should look like, with xxxxx replaced with your unique ID:

```
USE [ContosoAutoDb-XXXXX];
GO

SET STATISTICS IO ON
GO

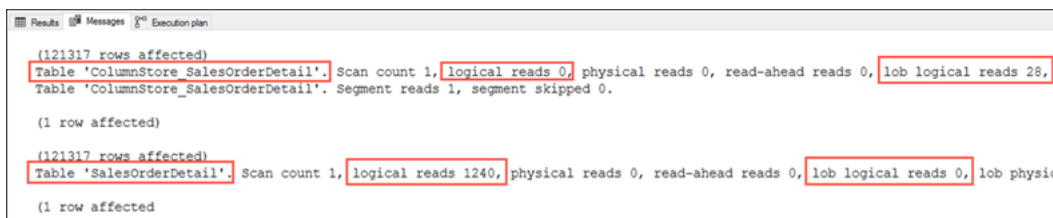
SELECT ProductId, LineTotal
FROM [Sales].[ColumnStore_SalesOrderDetail]

SELECT ProductId, LineTotal
FROM [Sales].[SalesOrderDetail]
```

- Select **Execute** from the toolbar to run the query.

Statistics IO reports on the amount of logical pages that are read in order to return the query results.

- Select the **Messages** tab of the Results pane, and compare two numbers, logical reads and lob logical reads. You should see a significant drop in total number of logical reads on the columns store table.



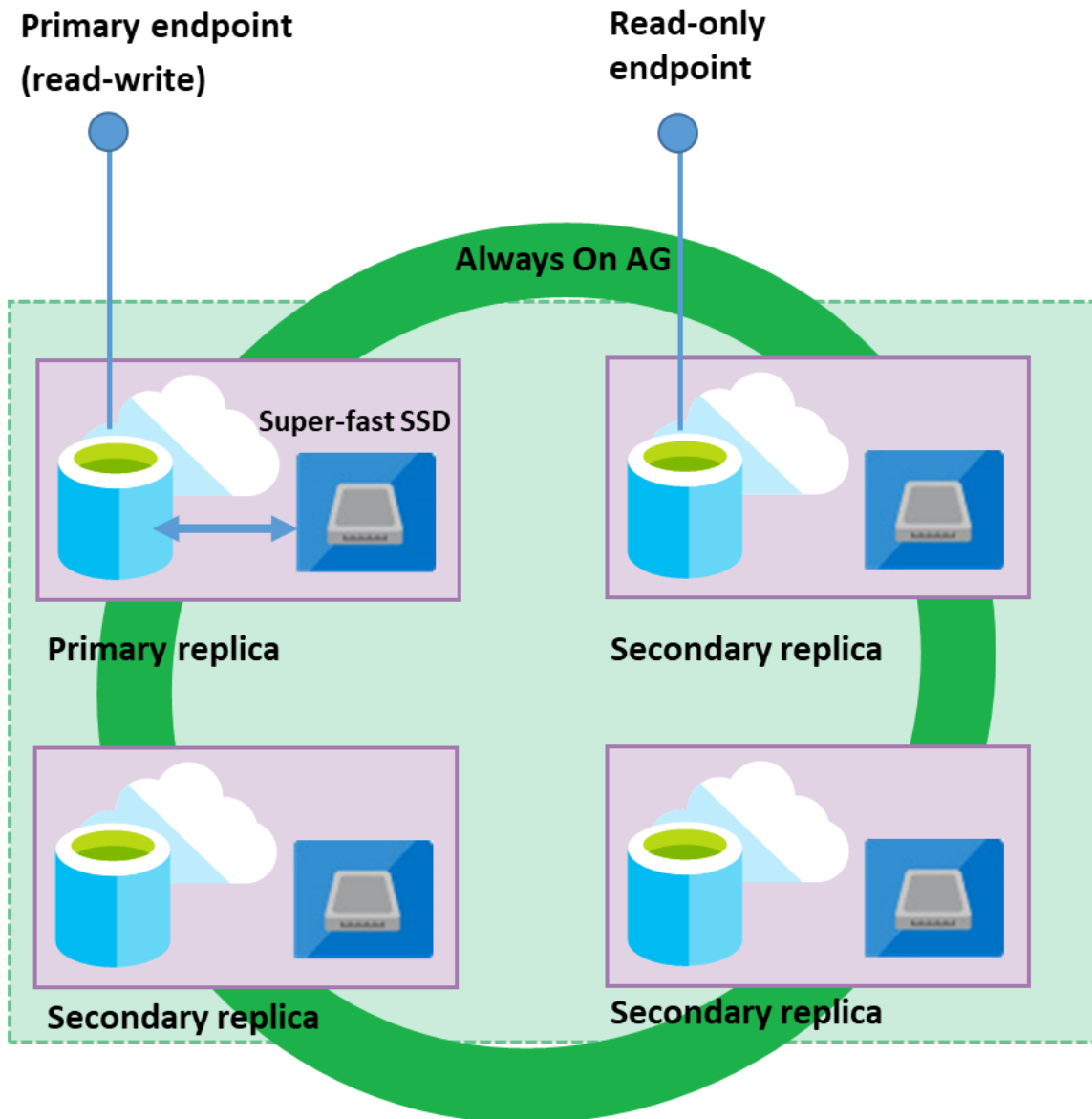
Various information is highlighted on the Messages tab of the Results pane.

Task 6: Use online secondary for read-only queries

In this task, you will look at how you can use the automatically created online secondary for reporting, without feeling the impacts of a heavy transactional load on the primary database. Each database in the SQL MI Business Critical tier is automatically provisioned with several AlwaysON replicas to support the availability SLA.

High availability in this architectural model is achieved by replication of compute (SQL Server Database Engine process) and storage (locally attached SSD) deployed in 4-node cluster, using technology similar to SQL Server [Always On Availability Groups](#). You can read more in the [SQL Database high availability](#) documentation.

[Read Scale-Out](#) allows you to load balance Azure SQL Database read-only workloads using the capacity of one read-only replica. This way the read-only workload will be isolated from the main read-write workload and will not affect its performance. To learn more, check out the [SQL Database Read Scale-Out documentation](#).



Business Critical service tier: collocated compute and storage

Business Critical service tier: collocated compute and storage.

The feature is intended for the applications that include logically separated read-only workloads, such as analytics, and therefore could gain performance benefits using this additional capacity at no extra cost.

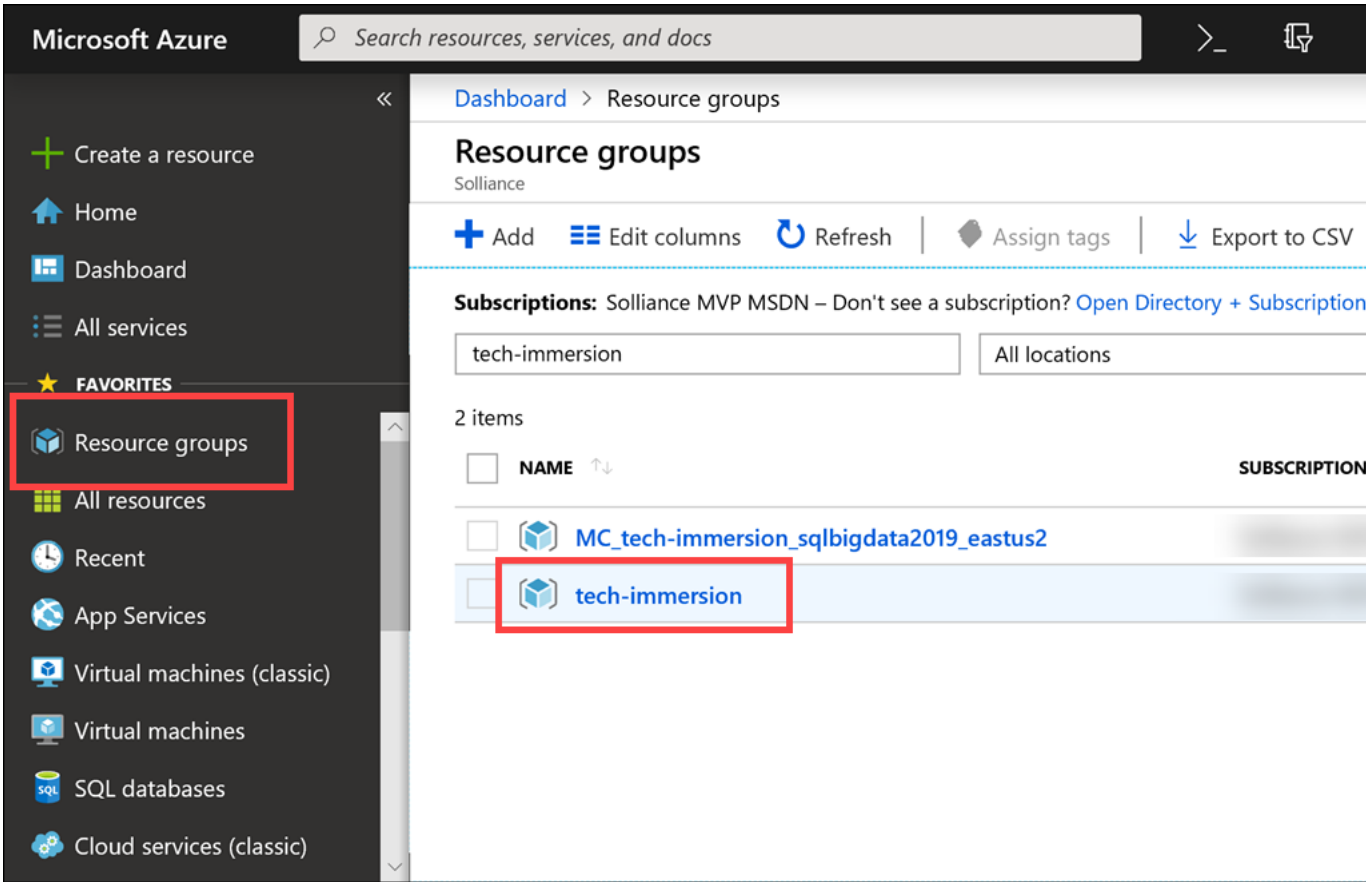
When you enable Read Scale-Out for a database, the `ApplicationIntent` option in the connection string provided by the client dictates whether the connection is routed to the write replica or to a read-only replica. Specifically, if the `ApplicationIntent` value is `ReadWrite` (the default value), the connection will be directed to the database's read-write replica. This is identical to existing behavior. If the `ApplicationIntent` value is `ReadOnly`, the connection is routed to a read-only replica.

For example, the following connection string connects the client to a read-only replica of the `tech-immersion-sql-mi` database:

```
Server=tcp:tech-immersion-sql-mi.3e134c88d9f6.database.windows.net;Database=ContosoAutoDb;User ID=tiuser;Password=Password.1234567890;Trusted_Connection=F;
```

Note the addition of `ApplicationIntent=ReadOnly;` to the end of the connection string.

1. Using a web browser, navigate to the [Azure portal](#), select **Resource groups** from the left-hand menu, and then select the resource group named **tech-immersion-XXXXXX** (where XXXXX is the unique ID assigned to you for this workshop).



The tech-immersion resource group is selected.

2. In the tech-immersion resource group, select the **techimmersionwebappXXXXXX** App Service from the list of resources (where XXXXX is the unique ID assigned to you for this workshop).

<input type="checkbox"/>	 tech-immersion-vnet	Virtual network
<input type="checkbox"/>	 tech-immersion-web	App Service plan
<input type="checkbox"/>	 tech-immersion-web	App Service
<input type="checkbox"/>	 VirtualClusterb52af2c3-7c09-4c67-8860-d597b67738ca	Virtual cluster
<input type="checkbox"/>	 vnet-tech-immersion-sql-mi	Virtual network

The App Service resource is selected from the list of resources in the tech-immersion resource group.

3. On the App Service overview blade, select the **URL** to open the web application in a browser window.

Resource group (change) : tech-immersion	URL : https://tech-immersion-app.tech-immersion.azurewebsites.net
Status : Running	App Service Plan : tech-immersion-app (Isolated: 1 Small)
Location : West US 2	App Service Environmen... : tech-immersion

The App service URL is highlighted.

4. In the ContosoAuto web app, select **Reports** from the menu.

Product Sales by Store Report

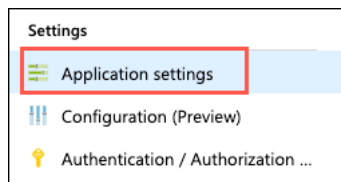
READ_WRITE

Store	Product	SalesTotal
A Bicycle Association	AWC Logo Cap	4463205.97
A Bike Store	AWC Logo Cap	2095434.93
A Cycle Shop	AWC Logo Cap	1939366.69
A Great Bicycle Company	AWC Logo Cap	6544020.83
A Typical Bike Shop	AWC Logo Cap	6544020.83
Acceptable Sales & Service	AWC Logo Cap	3866905.62

READ_WRITE is highlighted on the Reports page.

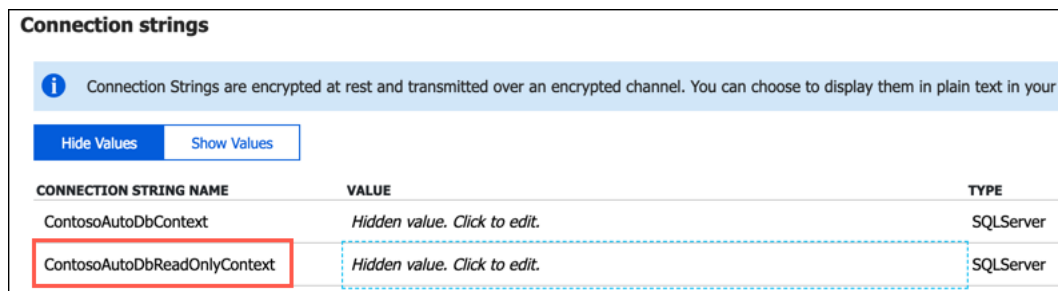
Note the READ_WRITE string on the page. This is the output from reading the Updateability property associated with the ApplicationIntent option on the target database. This can be retrieved using the SQL query `SELECT DATABASEPROPERTYEX(DB_NAME(), 'Updateability')`.

- Return to the App Service blade, and then select **Application settings** under Settings on the left-hand side.



The Application settings item is selected under Settings.

- On the Application settings blade, scroll down and locate the connection string named ContosoAutoDbReadOnlyContext within the **Connection strings** section.



The read-only connection string is highlighted.

- Select the **Value** for the ContosoAutoDbReadOnlyContext and paste the following parameter to end of the connection string.

```
ApplicationIntent=ReadOnly;
```

- Your ContosoAutoDbReadOnlyContext connection string should now look something like the following:

```
Server=tcp:tech-immersion-sqlmi-shared.521f7783692d.database.windows.net,1433;Persist Security Info=False;Database=ContosoAutoDb;User ID=tiuser;Passw
```

- Select **Save** at the top of the Application settings blade.



The save button on the Application settings blade is highlighted.

- Return to the ContosoAuto operations website you opened previously, and refresh the **Reports** page. The page should now look similar to the following:

Product Sales by Store Report

READ_ONLY

Store	Product	SalesTotal
A Bicycle Association	AWC Logo Cap	4463205.97
A Bike Store	AWC Logo Cap	2095434.93
A Cycle Shop	AWC Logo Cap	1939366.69
A Great Bicycle Company	AWC Logo Cap	6544020.83
A Typical Bike Shop	AWC Logo Cap	6544020.83
Acceptable Sales & Service	AWC Logo Cap	3866905.62

READ_ONLY is highlighted on the Reports page.

Notice the `updability` option is now displaying as `READ_ONLY`. With a simple addition to your database connection string, you are able to send read-only queries to the online secondary of your SQL MI database, allowing you to load-balance read-only workloads using the capacity of one read-only replica. The SQL MI Business Critical cluster has built-in Read Scale-Out capability that provides free-of charge built-in read-only node that can be used to run read-only queries that should not affect performance of your primary workload.

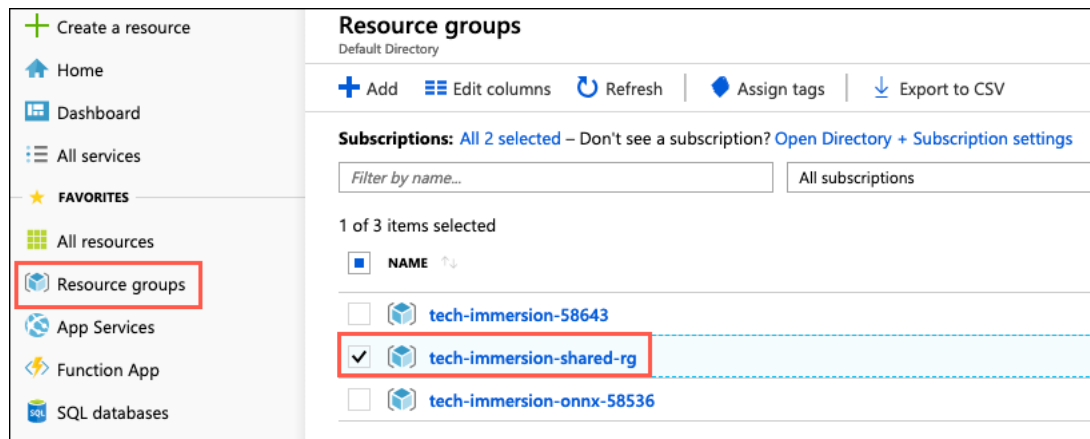
Task 7: Review Advanced Data Security Vulnerability Assessment

[SQL Database Advance Data Security](#) (ADS) provides advanced SQL security capabilities, including functionality for discovering and classifying sensitive data, surfacing and mitigating potential database vulnerabilities, and detecting anomalous activities that could indicate a threat to your database. ADS is enabled at the managed instance level by selecting **ON** on the **Advanced Data Security** blade for your managed instance. This turns ADS on for all databases on the managed instance. ADS uses an Azure Blob Storage account to save the associated outputs (e.g., assessment and vulnerability reports). In the interest of time for this workshop, the steps to enable ADS have already been performed on the shared SQL MI.

In this task, you will review an assessment report generated by [Advance Data Security](#) for the ContosoAutoDb database and take action to remediate one of the findings in your copy of the ContosoAutoDb database.

Advanced Data Security is enabled at the server level, and for this workshop it has already been enabled on the SQL MI, so you will focus on just your user-specific database.

- To review the Advanced Data Security assessment for your ContosoAutoDb-XXXX database, navigate to the **tech-immersion-shared-rg** resource group.



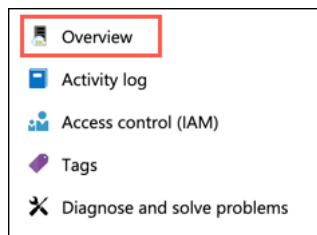
The tech-immersion-shared-rs is highlighted under Resource groups.

- In the shared resource group, select the **SQL Managed Instance** resource from the list.

<input type="checkbox"/>	tech-immersion-gateway	Virtual network gateway	West US
<input type="checkbox"/>	tech-immersion-gateway-ip	Public IP address	West US
<input checked="" type="checkbox"/>	tech-immersion-sqlmi-shared	SQL managed instance	West US
<input type="checkbox"/>	ContosoAutoDb (tech-immersion-sqlmi-shared/ContosoAuto...	Managed database	West US
<input type="checkbox"/>	ContosoAutoDb-58201 (tech-immersion-sqlmi-shared/Contos...	Managed database	West US

The SQL MI resource is highlighted in the list of resources in the shared resource group.

- On the SQL MI blade, select **Overview** from the left-hand menu.






The Overview menu item is highlighted.

- On the SQL MI Overview blade, scroll down and locate the list of databases on the Managed Instance, and then select your copy of the **ContosoAutoDb** database, which will be named ContosoAutoDb-XXXXX (e.g., ContosoAutoDb-0123), where XXXXX is the unique identifier assigned to you for this workshop.

3 Managed Instance databases

Search to filter databases...

NAME	STATUS
 ContosoAutoDb-58201	Online
 ContosoAutoDb-58643	Online
 ContosoAutoDb	Online

ContosoAutoDb is highlighted in the list of databases on the SQL MI.

You will see a list of all the databases on the SQL MI. Make sure you select the one which includes your assigned unique identifier from the list.

- On the **ContosoAutoDb-XXXXX** Managed database blade, select **Advanced Data Security** under Security in the left-hand menu and then select the **Vulnerability Assessment** tile.

ContosoAutoDb - Advanced Data Security

Managed database

Search (Ctrl+/)

Feedback

Overview

Activity log

Diagnose and solve problems

Settings

Locks

Automation script

Security

Advanced Data Security

Monitoring

Diagnostic settings

Support + troubleshooting

Resource health

New support request

Vulnerability Assessment

18 TOTAL

HIGH RISK FAILURES

MEDIUM RISK FAILURES

LOW RISK FAILURES

Failed Checks

SECURITY CHECK	RISK
Database communication using TDS should ...	High
Execute permissions to access the registry sh...	High
Minimal set of principals should be members...	High

...

Advanced Threat Protection

0 TOTAL

HIGH SEVERITY ALERTS

MEDIUM SEVERITY ALERTS

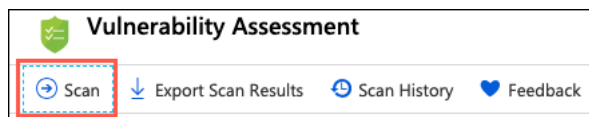
Security Alerts

DESCRIPTION	DATE
There are no alerts or recommendations to display.	

Advanced Data Security is selected in the left-hand menu, and the Vulnerability tile is highlighted.

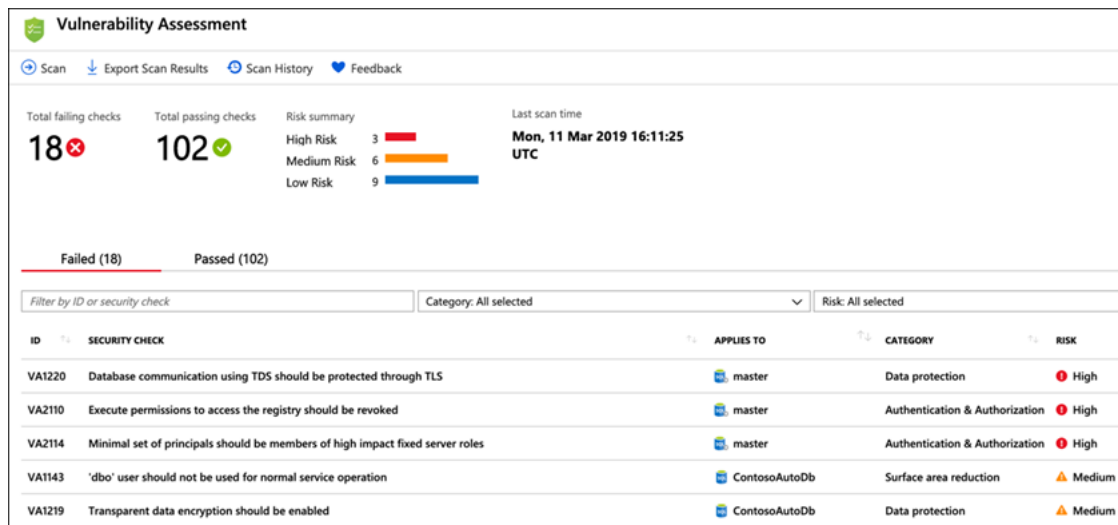
The [SQL Vulnerability Assessment service](#) is a service that provides visibility into your security state, and includes actionable steps to resolve security issues, and enhance your database security.

- On the Vulnerability Assessment blade, select **Scan** on the toolbar.



Vulnerability assessment scan button.

- When the scan completes, you will see a dashboard, displaying the number of failing checks, passing checks, and a breakdown of the risk summary by severity level.



The Vulnerability Assessment dashboard is displayed.

Scans are run on a schedule, so if you see a message that no vulnerabilities are found your database may not have been scanned yet. You will need to run a scan manually. To do this, select the **Scan** button on the toolbar, and follow any prompts to start a scan. This will take a minute or so to complete.

- In the scan results, take a few minutes to browse both the Failed and Passed checks, and review the types of checks that are performed. In the **Failed** the list, locate the security check for **Transparent data encryption**. This check has an ID of **VA1219**.

Failed (18) Passed (102)				
Filter by ID or security check Category: All selected Risk: All selected				
ID	SECURITY CHECK	APPLIES TO	CATEGORY	RISK
VA1220	Database communication using TDS should be protected through TLS	master	Data protection	High
VA2110	Execute permissions to access the registry should be revoked	master	Authentication & Authoriza...	High
VA2114	Minimal set of principals should be members of high impact fixed server roles	master	Authentication & Authoriza...	High
VA1143	'dbo' user should not be used for normal service operation	ContosoAutoDb	Surface area reduction	Medium
VA1219	Transparent data encryption should be enabled	ContosoAutoDb	Data protection	Medium
VA1276	Agent XPs feature should be disabled	master	Auditing & Logging	Medium

The VA1219 finding for Transparent data encryption is highlighted.

- Select the **VA1219** finding to view the detailed description.

VA1219 - Transparent data encryption should be enabled

☒ Approve as Baseline ☒ Clear Baseline

name	VA1219 - Transparent data encryption should be enabled
risk	Medium
status	FAIL
Applies To	ContosoAutoDb
description	Transparent data encryption (TDE) helps protect against the threat of malicious activity by performing real-time encryption and decryption of the database, associated backups, and transaction log files 'at rest', without requiring changes to the application. This rule checks that TDE is enabled on the database.
impact	Transparent Data Encryption (TDE) protects data 'at rest', meaning the data and log files are encrypted when stored on disk.
BENCHMARK REFERENCES	<ul style="list-style-type: none"> FedRAMP
RULE QUERY	<pre>SELECT CASE WHEN EXISTS (SELECT * FROM sys.databases</pre>
MICROSOFT RECOMMENDATION	True
BASELINE	Not set
ACTUAL RESULT	False
REMEDIATION	Enable TDE on the affected database. Please follow the instructions on https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/transparent-data-encryption

The details of the VA1219 - Transparent data encryption should be enabled finding are displayed with the description, impact, and remediation fields highlighted.

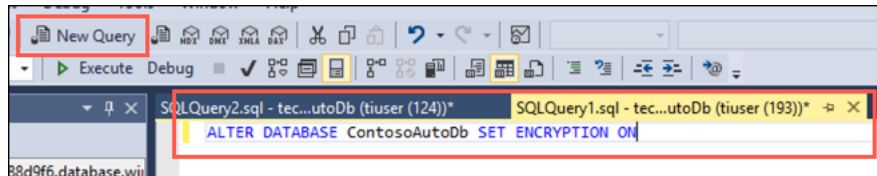
The details for each finding provide more insight into the reason for the finding. Of note are the fields describing the finding, the impact of the recommended settings, and details on remediation for the finding.

10. Let's now act on the recommendation remediation steps for the finding, and enable [Transparent Data Encryption](#) for the ContosoAutoDb database. To accomplish this, you will switch back to using SSMS for the next few steps.

Transparent data encryption (TDE) needs to be manually enabled for Azure SQL Managed Instance. TDE helps protect Azure SQL Database, Azure SQL Managed Instance, and Azure Data Warehouse against the threat of malicious activity. It performs real-time encryption and decryption of the database, associated backups, and transaction log files at rest without requiring changes to the application.

11. In SSMS, select **New Query** from the toolbar, paste the following SQL script into the new query window. Replace <XXXX> in the ALTER DATABASE statement to include the unique identifier of your database which will be ContosoAutoDb-XXXX (e.g., ContosoAutoDb-0123).

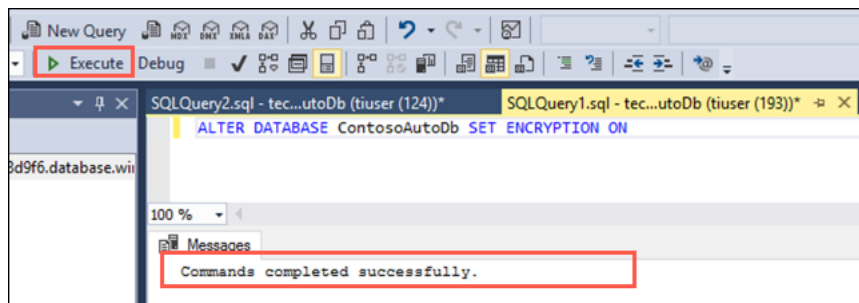
```
ALTER DATABASE [ContosoAutoDb-XXXX] SET ENCRYPTION ON
```



A new query window is displayed, with the script above pasted into it.

You turn transparent data encryption on and off on the database level. To enable transparent data encryption on a database in Azure SQL Managed Instance use must use T-SQL.

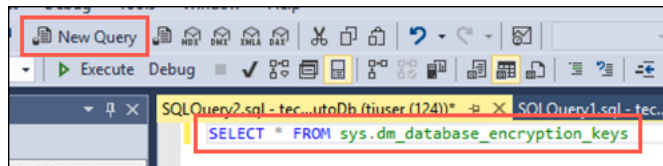
12. Select **Execute** from the SSMS toolbar. After a few seconds, you will see a message that the “Commands completed successfully.”



The Execute button is highlighted on the SSMS toolbar, and the Commands completed successfully message is highlighted in the output window.

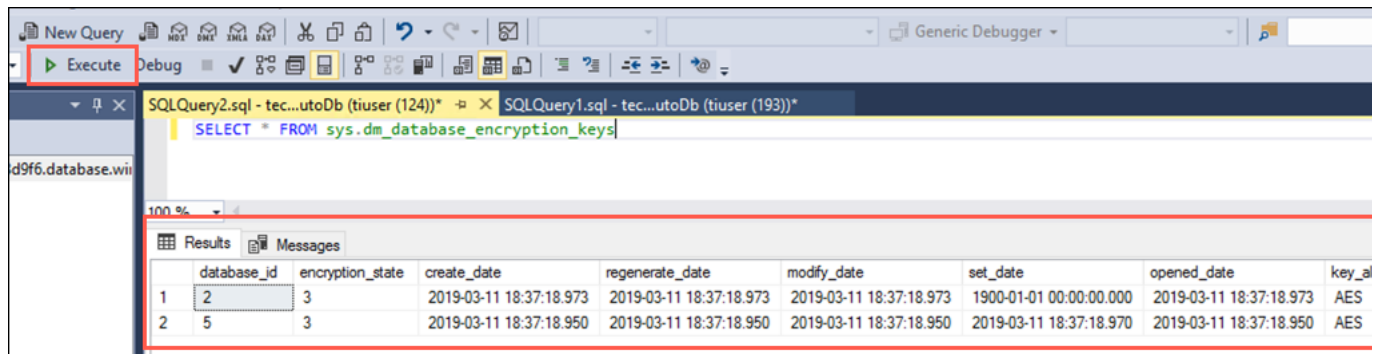
13. You can verify the encryption state and view information the associated encryption keys by using the [sys.dm_database_encryption_keys view](#). Select **New Query** on the SSMS toolbar again, and paste the following query into the new query window:

```
SELECT * FROM sys.dm_database_encryption_keys
```



The query above is pasted into a new query window in SSMS.

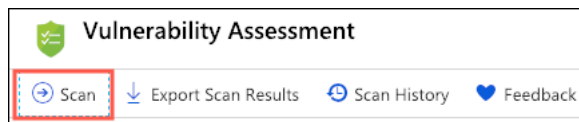
14. Select **Execute** from the SSMS toolbar. You will see two records in the Results window, which provide information about the encryption state and keys used for encryption.



The Execute button on the SSMS toolbar is highlighted, and in the Results pane the two records about the encryption state and keys for the ContosoAutoDb database are highlighted.

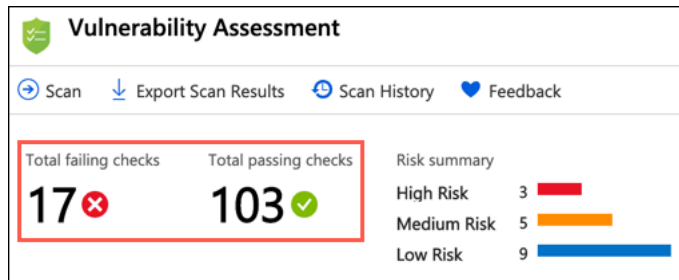
By default, service-managed transparent data encryption is used. A transparent data encryption certificate is automatically generated for the server that contains the database.

15. Return to the Azure portal and the Vulnerability Assessment blade for your copy of the ContosoAutoDb managed database (e.g, ContosoAutoDb-0123). On the toolbar, select **Scan** to start a new assessment of the database.



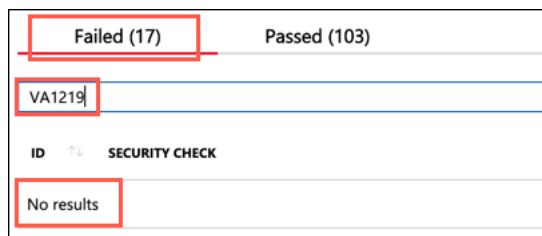
The Scan button on the SQL MI Vulnerability Assessment dialog is highlighted.

16. When the scan completes, notice that the numbers for failing and passing checks has changed. The number of failing checks has been reduced by 1 and the number of passing checks has increased by 1.



The total number of failing and passing checks is highlighted.

17. On the **Failed** tab, enter **VA1219** into the search filter box, and observe that the previous failure is no longer in the Failed list.



The Failed tab is highlighted and VA1219 is entered into the search filter. The list displays no results.

18. Now, select the **Passed** tab, and observe the **VA1219** check is listed with a status of PASS.

Failed (17)		Passed (103)	
VA1219		Category: All selected	Status: All selected
ID	SECURITY CHECK	APPLIES TO	CATEGORY
VA1219	Transparent data encryption should be enabled	ContosoAutoDb	Data protection

The Passed tab is highlighted and VA1219 is entered into the search filter. VA1219 with a status of PASS is highlighted in the results.

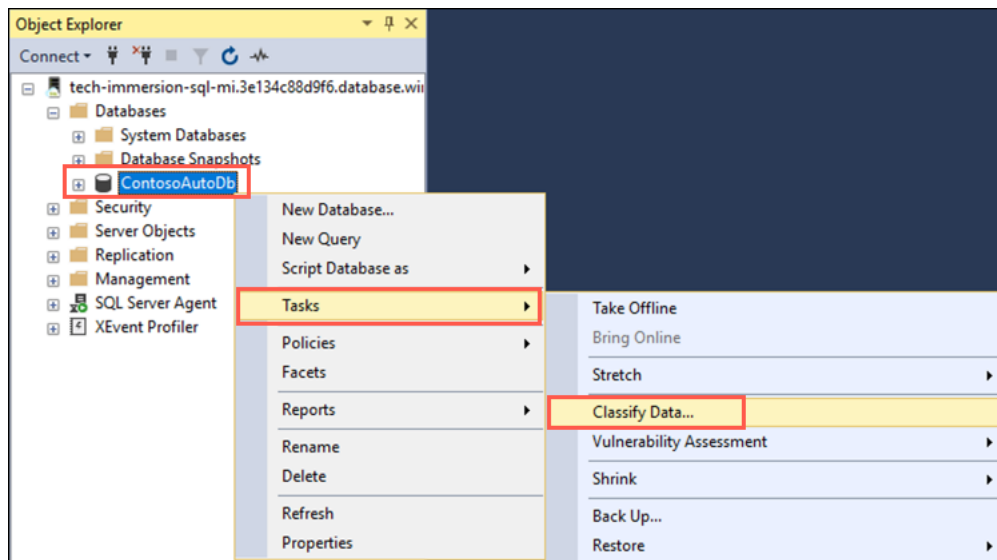
Using the SQL Vulnerability Assessment it is simple to identify and remediate potential database vulnerabilities, allowing you to proactively improve your database security.

Task 8: SQL Data Discovery and Classification

In this task, you will look at another **Advanced Data Security** feature available within the SQL MI database, [SQL Data Discovery and Classification](#). Data Discovery & Classification introduces a new tool built into SQL Server Management Studio (SSMS) for discovering, classifying, labeling & reporting the sensitive data in your databases. It introduces a set of advanced services, forming a new SQL Information Protection paradigm aimed at protecting the data in your database, not just the database. Discovering and classifying your most sensitive data (business, financial, healthcare, etc.) can play a pivotal role in your organizational information protection stature.

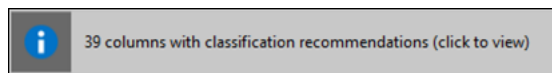
> This functionality is not currently available for SQL MI through the Azure portal, so you return to SSMS to use this capability.

1. In SSMS, right-click the ContosoAutoDb-XXXXX database (e.g., ContosoAutoDb-0123) in the Object Explorer (where XXXXX is the unique identifier assigned to you for this workshop), and then select **Tasks** and **Classify Data** in the context menus.



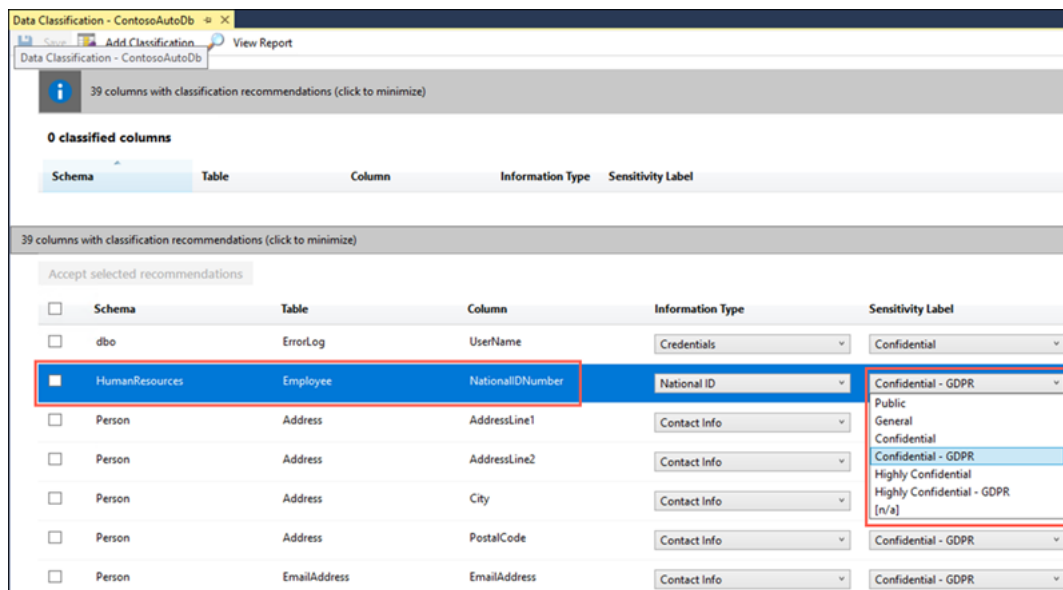
The Tasks > Classify Data context menu items are highlighted for the ContosoAutoDb database in SSMS.

2. In the Data Classification - ContosoAutoDb window, select the info link with the message *39 columns with classification recommendations (click to view)*.



The link to classification recommendations is displayed.

3. In the list of classification recommendations, select the recommendation for the **NationalIDNumber** field, and then expand the **Sensitivity Label** drop down list. You can see the list of built-in sensitivity classification, including those related to compliance requirements around GDPR.



The NationalIDNumber field is highlighted within the recommendations list, and the Sensitivity Label drop down is expanded and highlighted.

4. Select the check box at the top of the list to select all of the recommended classifications, and then select **Accept selected recommendations**.

Accept selected recommendations					
<input checked="" type="checkbox"/>	Schema	Table	Column	Information Type	Sensitivity Label
<input checked="" type="checkbox"/>	dbo	ErrorLog	UserName	Credentials	Confidential
<input checked="" type="checkbox"/>	Person	Password	PasswordHash	Credentials	Confidential
<input checked="" type="checkbox"/>	Person	Password	PasswordSalt	Credentials	Confidential
<input checked="" type="checkbox"/>	Person	PersonPhone	PhoneNumber	Contact Info	Confidential - GDPR
<input checked="" type="checkbox"/>	Person	PersonPhone	PhoneNumberTypeID	Contact Info	Confidential - GDPR
<input checked="" type="checkbox"/>	Person	PhoneNumberType	PhoneNumberTypeID	Contact Info	Confidential - GDPR
<input checked="" type="checkbox"/>	Production	ProductReview	EmailAddress	Contact Info	Confidential - GDPR
<input checked="" type="checkbox"/>	Purchasing	PurchaseOrderHeader	TaxAmt	Financial	Confidential
<input checked="" type="checkbox"/>	Purchasing	Vendor	AccountNumber	Financial	Confidential
<input checked="" type="checkbox"/>	Purchasing	Vendor	CreditRating	Credit Card	Confidential

All the recommended classifications are checked and the Accept selected recommendations button is highlighted.

5. Select **Save** on the toolbar of the Data Classification window.

Data Classification - ContosoAutoDb
Save
Add Classification
View Report

There are pending classification updates. Please save.

No classification recommendations.

39 classified columns

Schema	Table	Column	Information Type	Sensitivity Label
dbo	ErrorLog	UserName	Credentials	Confidential
Person	Password	PasswordHash	Credentials	Confidential
Person	Password	PasswordSalt	Credentials	Confidential
Person	PersonPhone	PhoneNumber	Contact Info	Confidential - GDPR

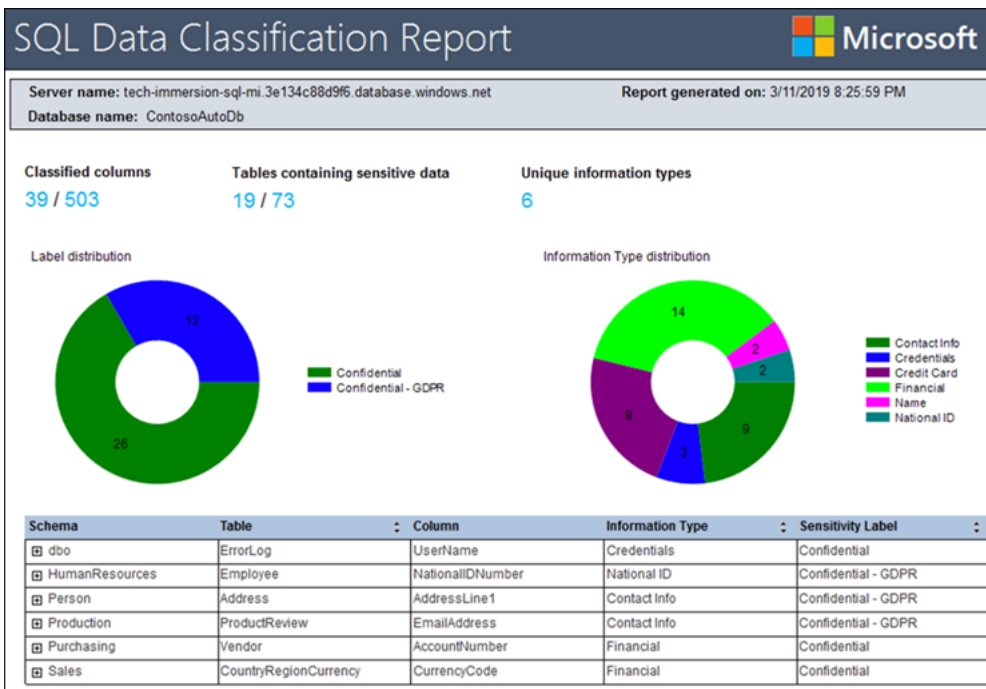
Save the updates to the classified columns list.

6. Select **View Report** on the Data Classification window to generate a report with a full summary of the database classification state.

Save
Add Classification
View Report

The View Report button is highlighted on the toolbar.

7. View the report.



The SQL Data Classification Report is displayed.

Wrap-up

In this experience you unlocked new capabilities for a SQL Server 2008 R2 database by performing a friction-free migration to Azure SQL Database Managed Instance. You learned how Azure SQL Database Managed Instance enables you to migrate on-premises databases quickly and easily into a fully-managed PaaS database running in Azure, with no application code changes. SQL MI provides a migration path for databases using features, such as Service broker, which previously prevented them from running in Azure SQL Database.

After you migrated the database into SQL MI, you explored some of advanced SQL features available only in Azure, including Advanced Data Security Vulnerability Assessments and Data Classification and Discovery. In addition, you enabled Dynamic Data Masking and created a ColumnStore index on a table in the database, demonstrating how SQL MI allows you to utilize features unavailable in SQL Server 2008 R2. You also examined how to connect to an online secondary replica of your database, which provides a free read-only copy of your database. This feature takes advantage of one of the high-availability features of the Azure SQL MI Business Critical service tier.

This experience was meant to provide a brief introduction to Azure SQL Database Managed Instance. There are many more features of SQL MI that you can now explore, including [Advanced Threat Detection](#) and [Transactional replication](#). Threat detection for Azure SQL Database Managed Instance detects anomalous activities indicating unusual and potentially harmful attempts to access or exploit databases. Transactional replication allows you to replicate data into an Azure SQL MI database from a remote SQL Server database or another instance database. You can also use it to push changes made in an instance database in SQL MI to a remote SQL Server database, to a single database in Azure SQL Database, or to a pooled database in an Azure SQL Database elastic pool.

Additional resources and more information

Use the links below as a starting point to continue learning about the capabilities and features available with Azure SQL Database Managed Instance.

- [Azure SQL Database](#)
 - [Service tiers](#)
- [What is Azure SQL Database Managed Instance?](#)
- [Database Migration Guide](#)
 - [Database Migration Assistant](#)
 - [Azure Database Migration Service](#)
- [Migrate SQL Server to an Azure SQL Database Managed Instance](#)
- [SQL Database Platform as a Service](#)
 - [Business continuity](#)
 - [High availability](#)
 - [Automated backups](#)
 - [Long-term back retention](#)
 - [Geo-replication](#)
 - [Scale resources](#)
- [How to use Azure SQL Database](#)
- [Azure updates for Azure SQL Database](#)
- [Azure SQL Database pricing](#)
- [Overview of Azure SQL Database security capabilities](#)
 - [Advanced data security](#)
 - [Data discovery and classification](#)
 - [SQL Vulnerability Assessment service](#)
 - [Threat detection](#)
- [SQL Database Read Scale-Out](#)
- [Connect an application to Azure SQL Database Managed Instance](#)