

Azure Discovery Days 2019

Data Analytics & Near Real Time Intelligence with Azure - Hands-On Lab Guide

Lab 2: Copy processed data from Azure Blob storage to Azure SQL DB

Summary

In this hands-on lab, you will:

1. Create an Azure SQL DB.
2. Create an Azure Data Factory.
3. Create an Azure Data Factory pipeline using the Copy Data wizard.
4. Create Power BI Reports using Power BI desktop and Azure SQL DB as data source.
5. Deploy Power BI reports to Power BI Online and create Dashboards.

About this Lab

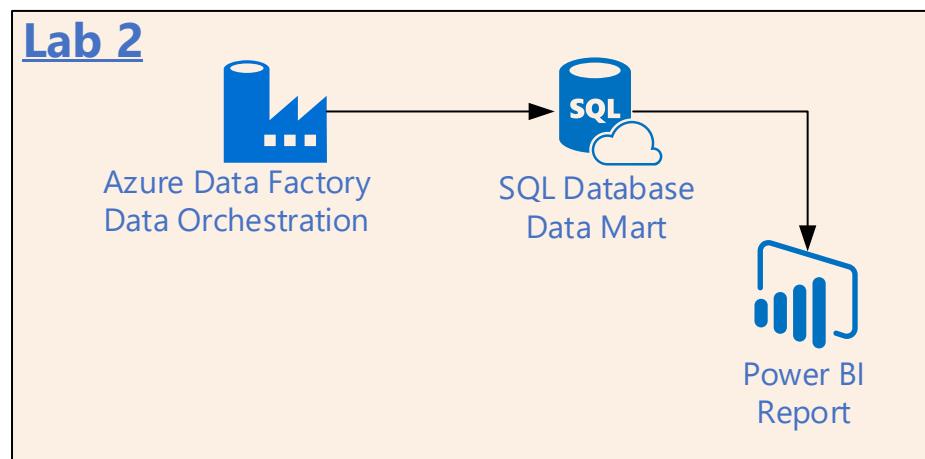
Copy data from Azure Blob storage (Parquet format) to Azure SQL Database using copy recursion option. And then create Power BI reports & dashboards using Azure SQL Database as a source.

References

<https://docs.microsoft.com/en-us/azure/data-factory/tutorial-copy-data-tool>

Architecture for this Lab

The tasks in this lab cover the following components of the overall architecture.



Task 1 – Create Azure SQL Database

Go to Azure portal and then select ‘Create a resource’ → Search for SQL database → Select SQL Database as shown:

The screenshot shows the Microsoft Azure portal interface. On the left, there's a dark sidebar with various navigation links. A red arrow points from the 'Create a resource' link at the top of the sidebar down towards the search bar. The main area is titled 'Everything' and contains a search bar with 'SQL Database' typed in. Another red arrow points from this search bar to the first result in the list. The results table has columns for NAME, PUBLISHER, and CATEGORY. The first result, 'SQL Database', is highlighted with a red box and has a red arrow pointing to it from below. Other visible results include 'Web App + SQL', 'SQL Database Reserved vCores', and several Microsoft products like 'SQL Server 2016 SP1 Enterprise on Windows Server 2016'.

NAME	PUBLISHER	CATEGORY
SQL Database	Microsoft	Databases
Web App + SQL	Microsoft	Web
SQL Database Reserved vCores	Microsoft	Databases
Striim for Real-Time Integration to SQL Database	Striim, Inc.	Compute
SQL Server 2016 SP1 Enterprise on Windows Server 2016	Microsoft	Compute
SQL Elastic database pool	Microsoft	Databases
SQL server (logical server)	Microsoft	Databases
SQL Beacon	WARDY IT Solutions	Compute
SQL Data Warehouse	Microsoft	Databases
Azure SQL Analytics (Preview)	Microsoft	Management Tools
ScaleArc for SQL Server	ScaleArc	Compute
Azure SQL Managed Instance	Microsoft	Databases
ScaleArc for SQL Server (pay-go)	ScaleArc	Compute
SQL Server Module	Microsoft	Databases

Please click create.

Home > New > Marketplace > Everything > SQL Database

Everything

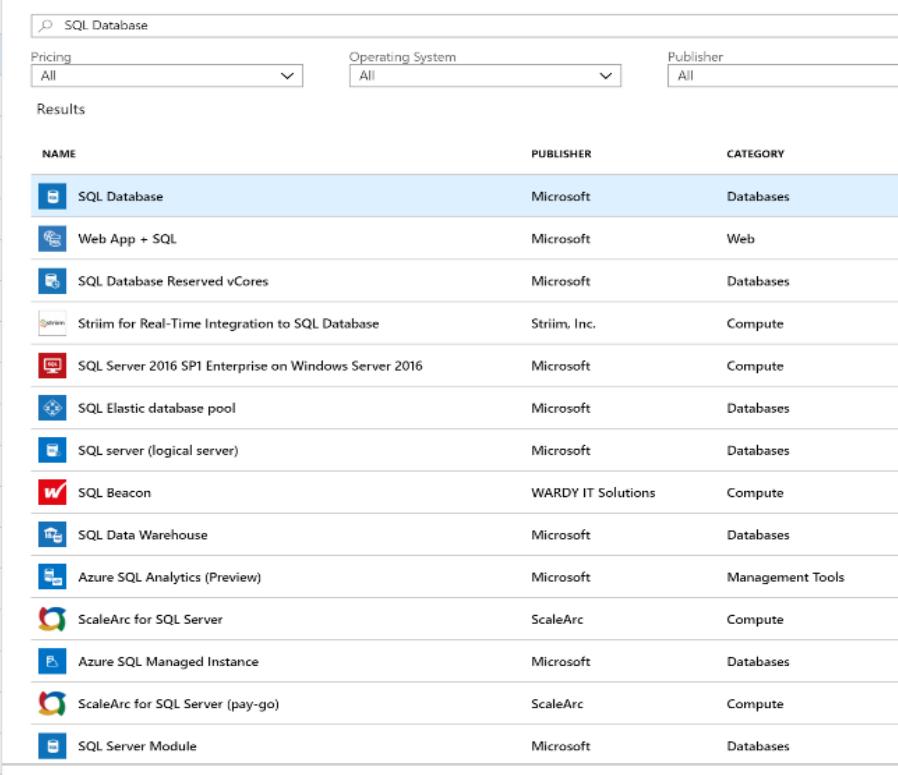
SQL Database

Microsoft

SQL Database is a cloud database service built for application developers that lets you scale on-the-fly without downtime and efficiently deliver your applications. Built-in advisors quickly learn your application's unique characteristics and dynamically adapt to maximize performance, reliability, and data protection.

Use this template to create a new database in the SQL Database service. You can create the database on a new logical server or on a logical server that already exists in your subscription.

[Save for later](#)



Results

NAME	PUBLISHER	CATEGORY
SQL Database	Microsoft	Databases
Web App + SQL	Microsoft	Web
SQL Database Reserved vCores	Microsoft	Databases
Striim for Real-Time Integration to SQL Database	Striim, Inc.	Compute
SQL Server 2016 SP1 Enterprise on Windows Server 2016	Microsoft	Compute
SQL Elastic database pool	Microsoft	Databases
SQL server (logical server)	Microsoft	Databases
SQL Beacon	WARDY IT Solutions	Compute
SQL Data Warehouse	Microsoft	Databases
Azure SQL Analytics (Preview)	Microsoft	Management Tools
ScaleArc for SQL Server	ScaleArc	Compute
Azure SQL Managed Instance	Microsoft	Databases
ScaleArc for SQL Server (pay-go)	ScaleArc	Compute
SQL Server Module	Microsoft	Databases

Related to your search ▾

-  Web App Microsoft
-  Azure Cosmos DB Microsoft

PUBLISHER Microsoft

USEFUL LINKS

- Documentation
- Service Overview
- Solutions you can deliver
- Pricing Details

[Create](#)



Enter Database name, select Azure subscription and select your Resource Group.

Select source as Blank Database. Then click on Server to create a new logical server.

Enter details as shown and please take note of admin user name and password you entered.

Preview Microsoft Azure Report a bug Search resources, services, and docs

Home > New > Marketplace > Everything > SQL Database > SQL Database > Server > New server

SQL Database

- * Database name: azdiscday2019
- * Subscription: Neelam's Microsoft Azure Internal Consum...
- * Resource group: DiscoveryDay-
- Create new
- * Select source: Blank database

Server

- + Create a new server
- nsocsaegbackup North Central US NSOS...
- nsosales South Central US NSOS...

New server

- * Server name: azdiscday2019 .database.windows.net
- * Server admin login: neelam
- * Password:
- * Confirm password:
- * Location: East US
- Allow Azure services to access server

Advanced Threat Protection: Start FREE Trial Not now

FREE trial period of 30 days, and then 991.44375 INR/server/month.

Learn more

Configure required settings

Want to use SQL elastic pool? Yes Not now

Pricing tier: Configure required settings

* Collation: SQL_Latin1_General_CI_AS

Create Automation options

Select

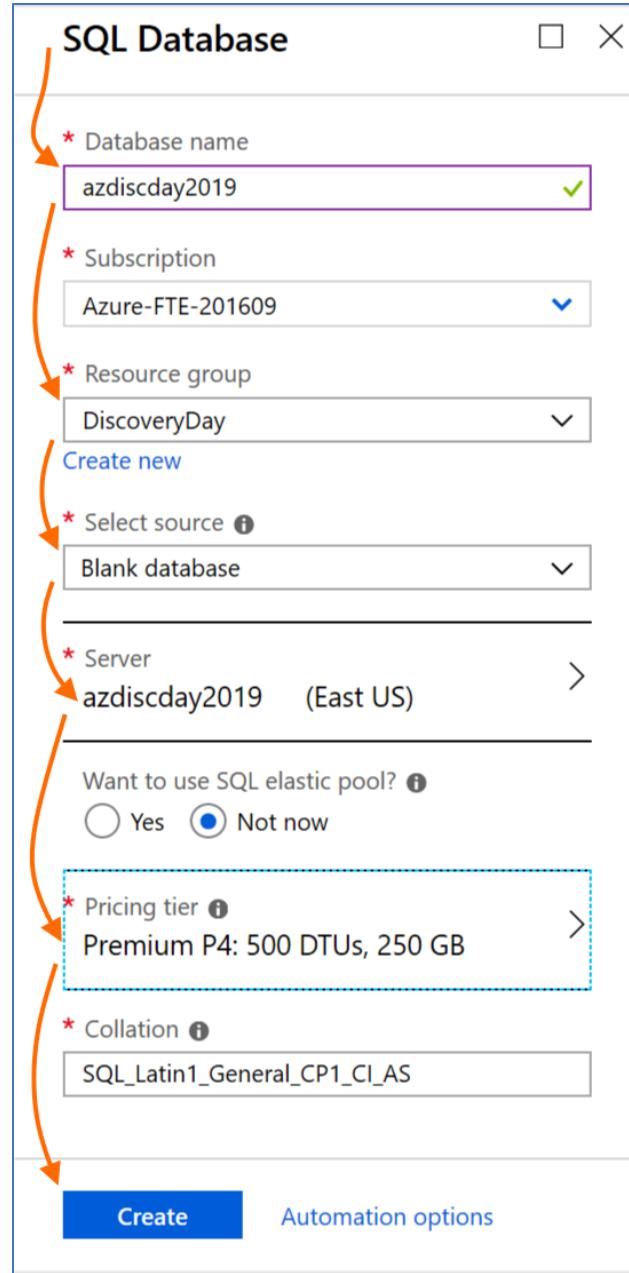
The screenshot shows the Microsoft Azure portal interface for creating a new SQL Database. On the left, the navigation menu includes options like Home, Dashboard, All services, Resource groups, App Services, SQL databases, Virtual machines (classic), Images, Cloud services (classic), Subscriptions, SQL servers, Data factories, Storage accounts, Virtual networks (classic), Azure Active Directory, Monitor, Security Center, Cost Management + Billing, Help + support, Advisor, and DevTest Labs. The main area shows the 'SQL Database' creation wizard. The 'Server' tab is open, displaying existing servers: nsocsaegbackup (North Central US) and nsosales (South Central US). The 'New server' tab is also open, requiring configuration of server name (azdiscday2019), server admin login (neelam), password, confirm password, location (East US), and enabling Azure service access. Two orange arrows point from the 'Configure required settings' sections in the 'SQL Database' and 'Server' tabs to the 'Select' button in the 'New server' tab, highlighting that these configurations are mandatory before proceeding.

Click on the pricing tier. While “S0” is the default, change the tier to “Premium” and slide the “DTUs” slider to **500 (P4)**. You can leave “Max data size” as is, or lower it to 250GB. Then click “Apply”.

Do not accept the initial default of “S0”. A higher performance level, such as P4 shown here, is critical to completing high-volume, high-IO data ingestion jobs such as this in a reasonable amount of time (like part of a lab).

The screenshot shows the Azure portal's configuration interface for a new SQL database. On the left, the 'SQL Database' configuration pane is visible, containing fields for Database name (azdiscday2019), Subscription (Azure-FTE-201609), Resource group (DiscoveryDay), Select source (Blank database), Server (azdiscday2019sql (East US)), and Pricing tier (Premium P4: 500 DTUs, 500 GB). A large orange arrow points from the 'Pricing tier' section towards the configuration area. In the center, the 'Configure' pane displays three pricing tiers: Basic (Starting at 4.99 USD / month), Standard (Starting at 15.00 USD / month), and Premium (Starting at 465.00 USD / month). The Premium tier is selected. The DTUs slider is set to 500 (P4). The Max data size slider is set to 250 GB. The Read scale-out setting is disabled. At the bottom, there are 'Create' and 'Automation options' buttons, and a prominent blue 'Apply' button with an orange arrow pointing to it.

Please review the details and then “Create”. Remember – please use a higher performance tier, such as P4.



Go to your Resource Group and you will see two new components created under your resource group: Azure SQL Database and its logical SQL Server.

The screenshot shows the Azure Resource Group Overview page for a group named "DiscoveryDay-". The left sidebar lists various settings like Overview, Activity log, Access control (IAM), Tags, Events, Quickstart, Resource costs, Deployments, Policies, and Properties. The main area displays subscription information (Neelam's Microsoft Azure Internal Consu...), deployment status (2 Succeeded), and tags. Below these are filtering options (Filter by name..., 158 types, All locations, No grouping) and a table of resources. The table has columns for NAME, TYPE, and LOCATION. It shows two items: "azdiscday2019" (SQL server, East US) and "azdiscday2019 (azdiscday2019/azdiscday2019)" (SQL database, East US). Two red arrows point to the names of these resources.

NAME	TYPE	LOCATION	⋮
azdiscday2019	SQL server	East US	⋮
azdiscday2019 (azdiscday2019/azdiscday2019)	SQL database	East US	⋮

By default, all other Azure services like Azure Data Factory and Power BI etc can connect to Azure SQL Server.

Go to Server and then click “Show firewall settings” as shown –

The screenshot shows the Azure portal interface for a SQL server named 'azdiscday2019'. On the left, there's a sidebar with various navigation options like Overview, Activity log, Access control (IAM), Tags, and Diagnose and solve problems. Below that is a 'Settings' section with links for Quick start, Failover groups, Manage Backups, Active Directory admin, SQL databases, SQL elastic pools, Deleted databases, Import/Export history, DTU quota, and Properties.

The main content area displays the server details:

- Resource group (change) : DiscoveryDay-
- Status : Available
- Location : East US
- Subscription (change) : Neelam's Microsoft Azure Internal Consumption
- Subscription ID : 698e7133-d0be-44f4-bc25-76e2296b0fb0
- Tags (change) : Click here to add tags
- Server admin : neelam
- Firewalls and virtual net... : Show firewall settings (highlighted with an orange arrow)
- Active Directory admin : negupt@microsoft.com

Below the details, there are tabs for Notifications (0) and Features (6). Under Features, there are six cards:

- Active Directory admin**: Allows you to centrally manage identity and access to your Azure SQL databases. Status: CONFIGURED.
- Advanced Data Security**: Data Discovery & Classification, Vulnerability Assessment and Threat Detection. Status: NOT CONFIGURED.
- Automatic tuning**: Monitors and tunes your database automatically to optimize performance. Status: NOT CONFIGURED.
- Auditing**: Track database events and writes them to an audit log in Azure storage. Status: NOT CONFIGURED.
- Failover groups**: Automatically manages replication, connectivity and failover for a set of databases. Status: NOT CONFIGURED.
- Transparent data encryption**: Encryption at rest for your databases, backups, and logs. Status: SERVICE MANAGED KEY.

Allow access to Azure Services option is ON by default.

azdiscday2019 - Firewalls and virtual networks

Save Discard Add client IP

Connections from the IPs specified below provides access to all the databases in azdiscday2019.

Allow access to Azure services **ON** **OFF**

Client IP address

RULE NAME	START IP	END IP

Connections from the VNET/Subnet specified below provides access to all databases in azdiscday2019.

Virtual networks + Add existing virtual network + Create new virtual network

RULE NAME	VIRTUAL NETWORK	SUBNET	ADDRESS RANGE
No vnet rules for this server.			

If you want to connect from your laptop or VM using tools like SQL Server Management Studio or Power BI Desktop, then you will need to whitelist all required IP addresses.

The screenshot shows the Azure portal interface for managing a resource group named 'azdiscday2019'. The left sidebar contains navigation links for Home, Resource groups, DiscoveryDay-, and azdiscday2019. The main content area displays the resource group details:

- Resource group (change) : DiscoveryDay-**
- Status** : Available
- Location** : East US
- Subscription (change)** : Neelam's Microsoft Azure Internal Consumption
- Subscription ID** : 698e7133-d0be-44f4-bc25-76e2296b0fb0
- Tags (change)** : Click here to add tags

Under the **Settings** section, there are several tabs: Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, Quick start, Failover groups, Manage Backups, Active Directory admin, SQL databases, SQL elastic pools, Deleted databases, Import/Export history, DTU quota, and Properties. The **Firewalls and virtual net...** tab is currently selected, showing the following configuration:

- Server admin** : neelam
- Firewalls and virtual net...** : Show firewall settings (highlighted with an orange arrow)
- Active Directory admin** : Not configured

Below this, there are six feature cards:

- Active Directory admin**: Allows you to centrally manage identity and access to your Azure SQL databases. Status: NOT CONFIGURED.
- Advanced Threat Protection**: Data Discovery & Classification, Vulnerability Assessment and Threat Detection. Status: NOT CONFIGURED.
- Automatic tuning**: Monitors and tunes your database automatically to optimize performance. Status: NOT CONFIGURED.
- Auditing**: Track database events and writes them to an audit log in Azure storage. Status: NOT CONFIGURED.
- Failover groups**: Automatically manages replication, connectivity and failover for a set of databases. Status: NOT CONFIGURED.
- Transparent data encryption**: Encryption at rest for your databases, backups, and logs. Status: NOT CONFIGURED.

Click on Add Client IP and then click save. It will automatically whitelist your IP to use Azure SQL DB. It may take up to 5 minutes for this change to take effect. (Make sure you do this part from the machine – i.e. your laptop or the lab VM – from where you will also run SSMS or Power BI.)

Home > Resource groups > DiscoveryDay- > azdiscday2019 - Firewalls and virtual networks

azdiscday2019 - Firewalls and virtual networks

SQL server

Search (Ctrl+ /)

Save Discard Add client IP

Connections from the IPs specified below provides access to all the databases in azdiscday2019.

Allow access to Azure services

ON OFF

Client IP address 167.220.148.7

RULE NAME	START IP	END IP	...
ClientIPAddress_2019-1-19	167.220.148.7	167.220.148.7	...

Connections from the VNET/Subnet specified below provides access to all databases in azdiscday2019.

Virtual networks + Add existing virtual network + Create new virtual network

RULE NAME	VIRTUAL NETWORK	SUBNET	ADDRESS RANGE	ENDPOINT STATUS	RESOURCE GRO
No vnet rules for this server.					

Now we will connect to Azure SQL Database to execute DDL. There are two options to connect to Azure SQL Database –

1. Using SQL query editor on the Azure portal
2. Using SQL Server Management Studio

Option 1- Go to Azure SQL Database query editor and execute sql statements to create schema for the tables and view.

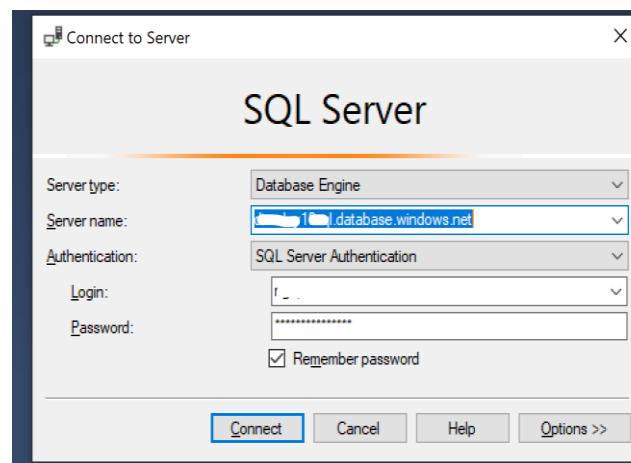
The screenshot shows the Azure portal's Query editor (preview) interface. On the left, there's a sidebar with various database management options like Overview, Activity log, Tags, Diagnose and solve problems, Quick start, and Query editor (preview). The Query editor (preview) option is highlighted with an orange arrow. The main area shows a query window titled 'Query 1' containing the following SQL code:

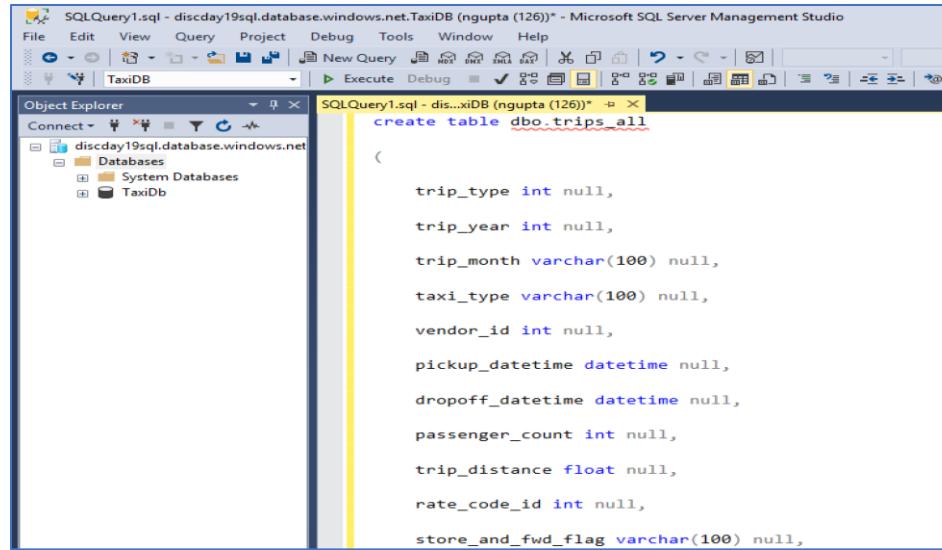
```
1 Create table tblTrip_All
2 (
3     trip_type Int,
4     trip_year Int,
5     trip_month Varchar(100),
6     taxi_type Varchar(100),
7     vendor_id Int,
8     pickup_datetime Date,
9     dropoff_datetime Date,
10    passenger_count Int,
11    trip_distance Float,
12    rate_code_id Int,
13    store_and_fwd_flag Varchar(100),
14    pickup_location_id Int,
15    dropoff_location_id Int ,
```

Below the code, there are 'Results' and 'Messages' tabs, and a search bar at the bottom.

Option -2 - Go to SQL Server Management Studio client tool and connect to Azure SQL Database as shown –

Enter Azure SQL Server Name, Login and Password.



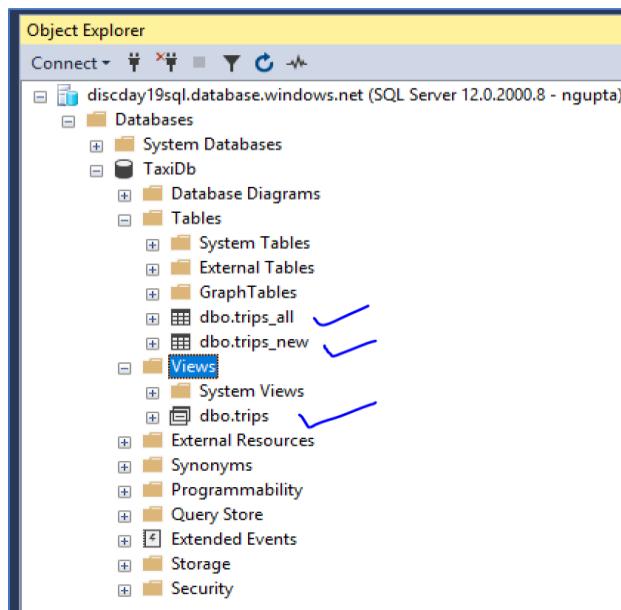


The screenshot shows the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows a connection to 'discday19sql.database.windows.net' with the database 'TaxiDb' selected. The central pane displays a SQL query window with the following code:

```
SQLQuery1.sql - discday19sql.database.windows.net.TaxiDB (ngupta (126)) - Microsoft SQL Server Management Studio
File Edit View Query Project Debug Tools Window Help
New Query Execute Debug
Object Explorer
Connect
discday19sql.database.windows.net
Databases
System Databases
TaxiDb
SQLQuery1.sql - discday19sql.database.windows.net.TaxiDB (ngupta (126))
create table dbo.trips_all
(
    trip_type int null,
    trip_year int null,
    trip_month varchar(100) null,
    taxi_type varchar(100) null,
    vendor_id int null,
    pickup_datetime datetime null,
    dropoff_datetime datetime null,
    passenger_count int null,
    trip_distance float null,
    rate_code_id int null,
    store_and_fwd_flag varchar(100) null,
```

The script to create tables and a view for the taxi data is at: <https://raw.githubusercontent.com/plzm/azure-discoveryday2019-mdw/master/labs/lab2/lab2.sql>

When the script gets executed successfully, right-click the “Databases” node and select “Refresh”. You will see three objects under your database-



Task 2 – Create Azure Data Factory Project

In the Azure portal, click “+ Create a resource” and search for “Azure Data Factory”. (You can also click “+ Add” in your Resource Group.)

The screenshot shows the Azure Marketplace search results for "Azure Data Factory". The search bar at the top contains the text "Azure Data Factory". The results table below has three columns: NAME, PUBLISHER, and CATEGORY. The first result, "Data Factory" by Microsoft under the Analytics category, is highlighted with an orange border. A red arrow points from the "Create a resource" button in the left sidebar to the search bar. Another red arrow points from the "Containers" category in the sidebar to the "Data Factory" result in the search results.

NAME	PUBLISHER	CATEGORY
Azure Data Factory Analytics (Preview)	Microsoft	Management Tools
OutSystems on Microsoft Azure	OutSystems	Compute
Data Factory	Microsoft	Analytics
Data Science Virtual Machine - Windows 2012	Microsoft	Compute
Data Science Virtual Machine - Windows 2016	Microsoft	Compute
DC/OS on Azure	Mesosphere	Compute
Diagramics Visual Server	Diagramics	Compute
CloudBlaze	Rawcubes Inc	Compute
SoftNAS Cloud Platinum - 10TB	SoftNAS	Compute
SoftNAS Cloud Platinum - 50TB	SoftNAS	Compute
SoftNAS Cloud Platinum - 20TB	SoftNAS	Compute
SoftNAS Cloud Platinum - 1TB	SoftNAS	Compute
FreeStor Storage Server	Falconstor Software	Compute
Cisco CSR1000V- Sec Pkg. Max Performance- XE 16.10	Cisco Systems, Inc.	Compute

Data Factory

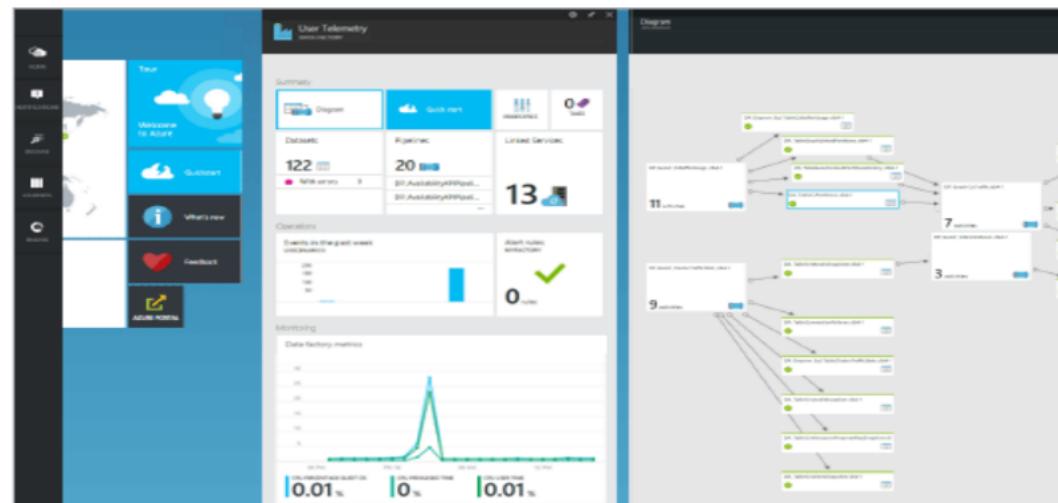
Microsoft



Microsoft Azure Data Factory is a cloud-based data integration service that automates the movement and transformation of data. You can quickly create, deploy, schedule, and monitor highly-available, fault tolerant data flow pipelines. Move and transform data of all shapes and sizes, and deliver the results to a range of destination storage services. Monitor all of your data pipelines and service health at a glance with a rich visual experience. Easily consume the data produced with BI, analytics tools, and other applications to drive key business insights and decisions.

- Compose data storage, movement and processing services into data flow pipelines
- Enhanced HDInsight integration including HCAT and on-demand cluster management
- Schedule data pipelines with fine-tuned control
- New data connectors for on-premises and cloud data sources
- Integration with Azure Machine Learning and Azure Batch
- Globally deployed data movement as a service
- Create, edit and deploy data pipelines with a Visual Studio plug-in

Save for later



PUBLISHER

Microsoft

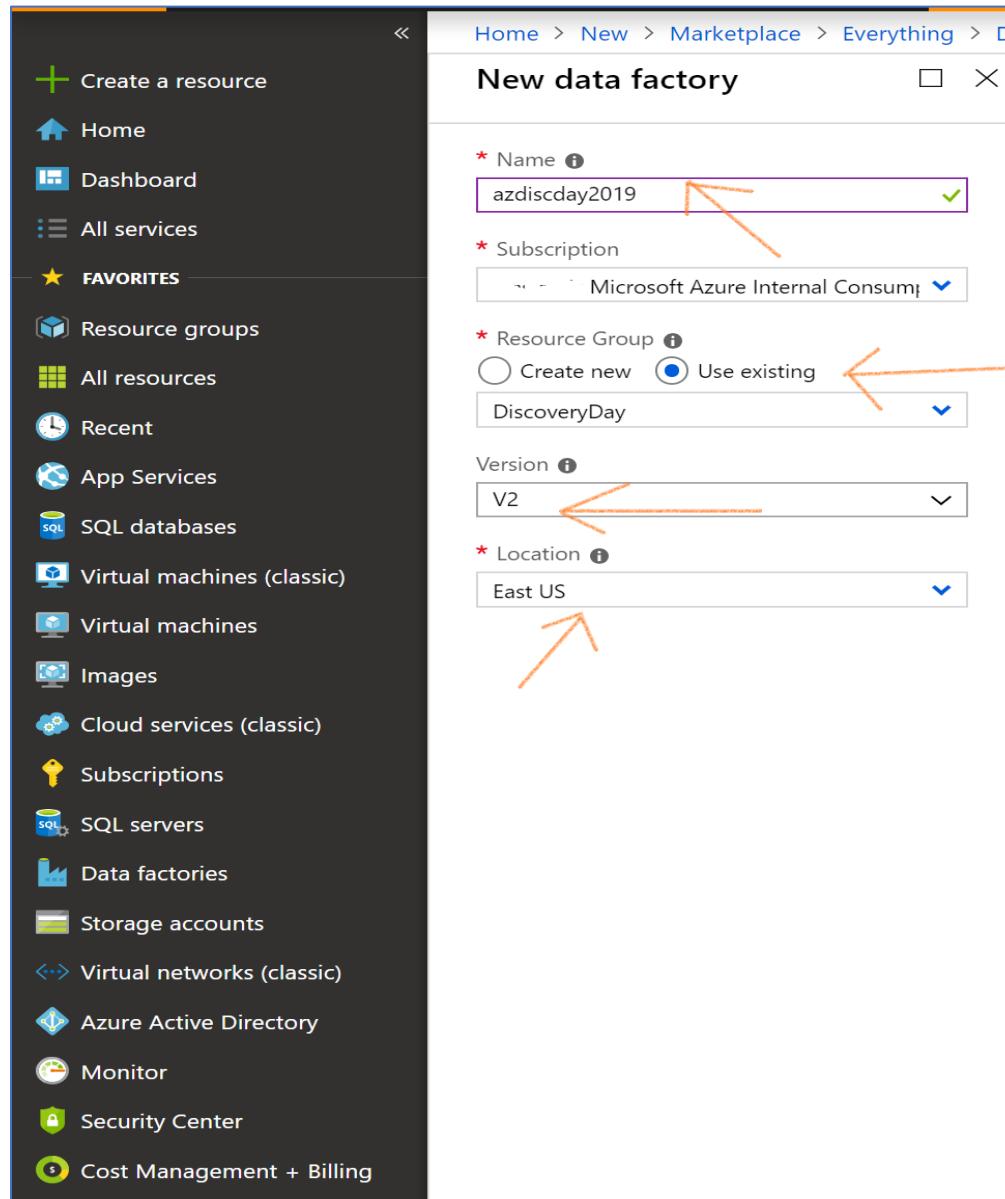
USEFUL LINKS

[Documentation](#)
[Service overview](#)
[Pricing details](#)

[Create](#)



Please enter Data Factory Name, Select Azure subscription and then select your Resource Group. Set “Version” to “V2”, and select the same Azure region you have been using so far. Then click “Create” and wait for the success notification.



The screenshot shows the Azure Notifications page. At the top, there are several icons: a left arrow, a refresh, a bell, a gear, a question mark, a smiley face, and the user's email address (@microsoft.com) with the Microsoft logo. Below the header, the word "Notifications" is displayed in bold. A blue button at the top left says "More events in the activity log" with a right-pointing arrow. On the right side of the same row, there is a "Dismiss all" link. The main content area contains a single notification card. The card has a green circular icon with a white checkmark followed by the text "Deployment succeeded". Below this, a detailed message states: "Deployment 'Microsoft.DataFactory-azdiscday2019' to resource group 'DiscoveryDay-' was successful." Two blue buttons are at the bottom of the card: "Go to resource" and "Pin to dashboard". To the right of the "Pin to dashboard" button is the text "a few seconds ago". A red arrow points from the text "a few seconds ago" up towards the checkmark icon.

More events in the activity log →

Dismiss all

Deployment succeeded

Deployment 'Microsoft.DataFactory-azdiscday2019' to resource group 'DiscoveryDay-' was successful.

Go to resource Pin to dashboard

a few seconds ago

When you click “Go to resource”, it will take you to the Data Factory project as shown – on this screen, click Author & Monitor.

The screenshot shows the Azure portal interface for a Data Factory project named "azdiscday2019".

Left sidebar (Navigation):

- Home > azdiscday2019
- azdiscday2019 Data factory (V2)
- Search (Ctrl+ /)
- Overview** (selected)
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems
- Settings
- Locks
- General
- Properties
- Getting Started
- Quick start
- Monitoring
- Alerts
- Metrics
- Diagnostic settings
- Support + troubleshooting
- Resource health
- New support request

Main Content Area:

Delete

Resource group (change) : DiscoveryDay- Type : Data factory (V2)
Status : Succeeded Getting started : Quick start
Location : East US
Subscription (change) : Neelam's Microsoft Azure Internal Consumption
Subscription ID : 698e7133-d0be-44f4-bc25-76e2296b0fb0

Documentation

Author & Monitor (highlighted with an orange arrow)

Monitoring

PipelineRuns

	6 PM	Jan 19	6 AM	12 PM
Succeeded pipeline r... azdiscday2019	0	0		
Failed pipeline runs... azdiscday2019			0	

ActivityRuns

	6 PM	Jan 19	6 AM	12 PM
Succeeded activity r... azdiscday2019	0	0		
Failed activity runs... azdiscday2019			0	

There are two ways to create and edit Azure Data Factory pipelines. There's a “Copy Data” wizard, which is very effective to build simple data copying pipelines, and this is what you will use for this lab. The other options (“Create pipeline” and “Create pipeline from template”) are more powerful and flexible, and you would use those to build orchestrations more complex than the kind of file copying you'll do in this lab.

To get started, click “Copy Data”.

The screenshot shows the Microsoft Azure Data Factory interface. At the top, the navigation bar includes 'Microsoft Azure' and 'Data Factory' with a dropdown menu, and a status bar with signal strength and battery icons. A purple header bar contains the text 'Help us improve. [Click here](#) to tell us how we are doing.' Below this, the main title 'Azure Data Factory' is displayed above the heading 'Let's get started'. Five circular icons represent different actions: 'Create pipeline' (blue cylinder), 'Create pipeline from template' (yellow and green flowchart), 'Copy Data' (two blue cylinders with yellow stars), 'Configure SSIS Integration' (green cylinder with magnifying glass), and 'Set up Code Repository' (red square with gear and code). An orange arrow originates from the text 'To get started, click “Copy Data”.' and points to the 'Copy Data' icon. This icon is also enclosed in a dashed blue rectangular box. In the bottom left corner, there is a 'Videos' section with a blue button labeled 'View'.

Enter Task Name and then select “Run once now” for this lab. Then click “Next”.

(There is also an option to create a recurring run schedule, but for this lab we’ll run the pipeline just once. You can still manually re-run the pipeline if needed.

Copy Data

Properties

Task name *

Task description

Task cadence or Task schedule

Run once now Run regularly on schedule

Previous **Next**

The screenshot shows the 'Copy Data' pipeline creation interface. On the left, a vertical sidebar lists six steps: 1 Properties (highlighted), 2 Source, 3 Destination, 4 Settings, 5 Summary, and 6 Deployment. The main panel is titled 'Properties' and contains fields for 'Task name' (set to 'CopyParquetToSQL') and 'Task description'. Below these is a section for 'Task cadence or Task schedule' with two radio buttons: 'Run once now' (selected) and 'Run regularly on schedule'. At the bottom are 'Previous' and 'Next' buttons. Red arrows highlight the 'Properties' step in the sidebar and the 'Run once now' radio button in the main panel.

In this lab, we are going to copy data from Azure blob storage (the Parquet data you created at the end of lab 1) to Azure SQL DB.

First, let's create a connection to our source data. Click “+ Create new connection”.

The screenshot shows the 'Copy Data' interface with the 'Properties' step selected. A red arrow points from the 'Source' section of the left sidebar to the 'Create new connection' button on the right. The 'Source data store' screen is displayed, featuring tabs for All, Azure, Database, File, Generic Protocol, NoSQL, and Services and apps. The 'All' tab is selected. Below the tabs are two input fields: 'All' and 'Filter by name', followed by the '+ Create new connection' button.

Properties
One time copy

Source

Connection

Dataset

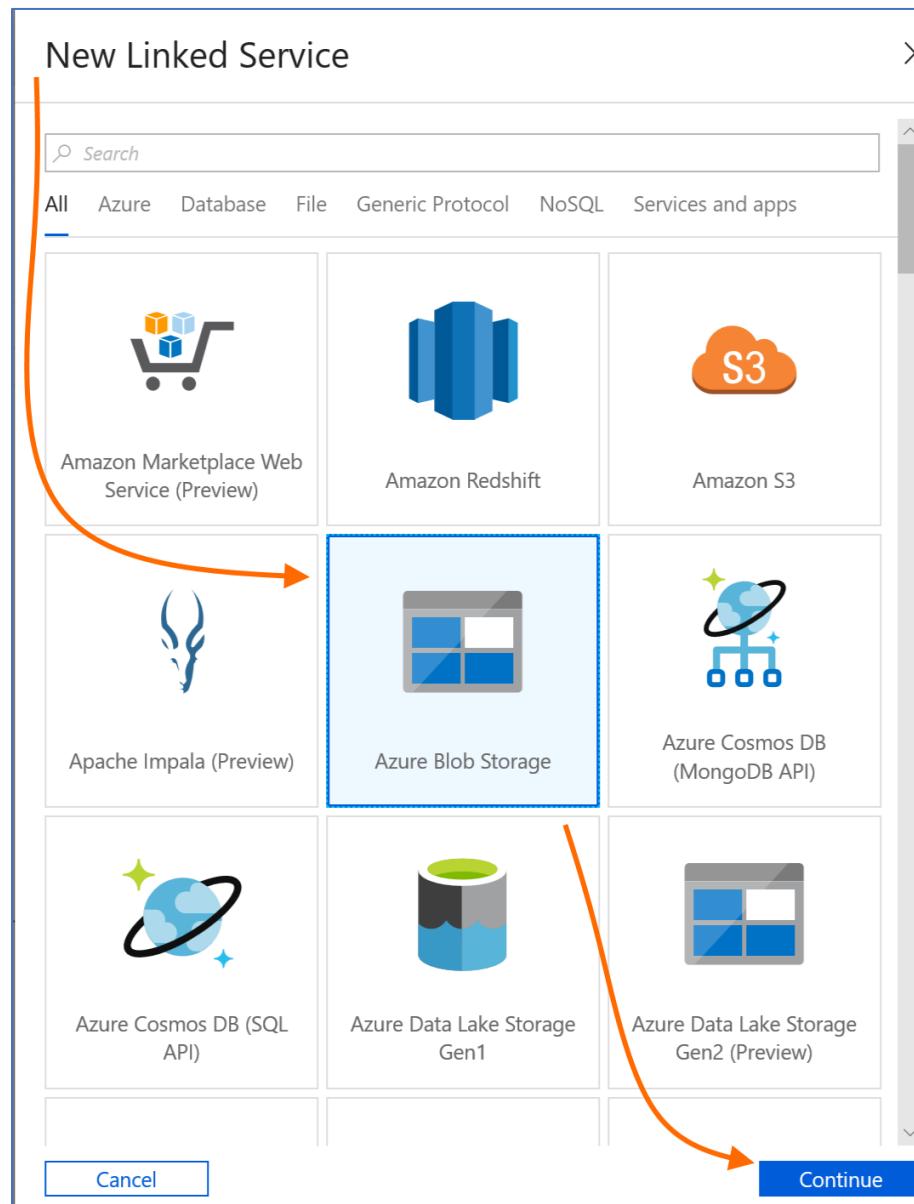
Source data store

Specify the source data store for the copy task. You can use an existing data store connection or specify a new data store.

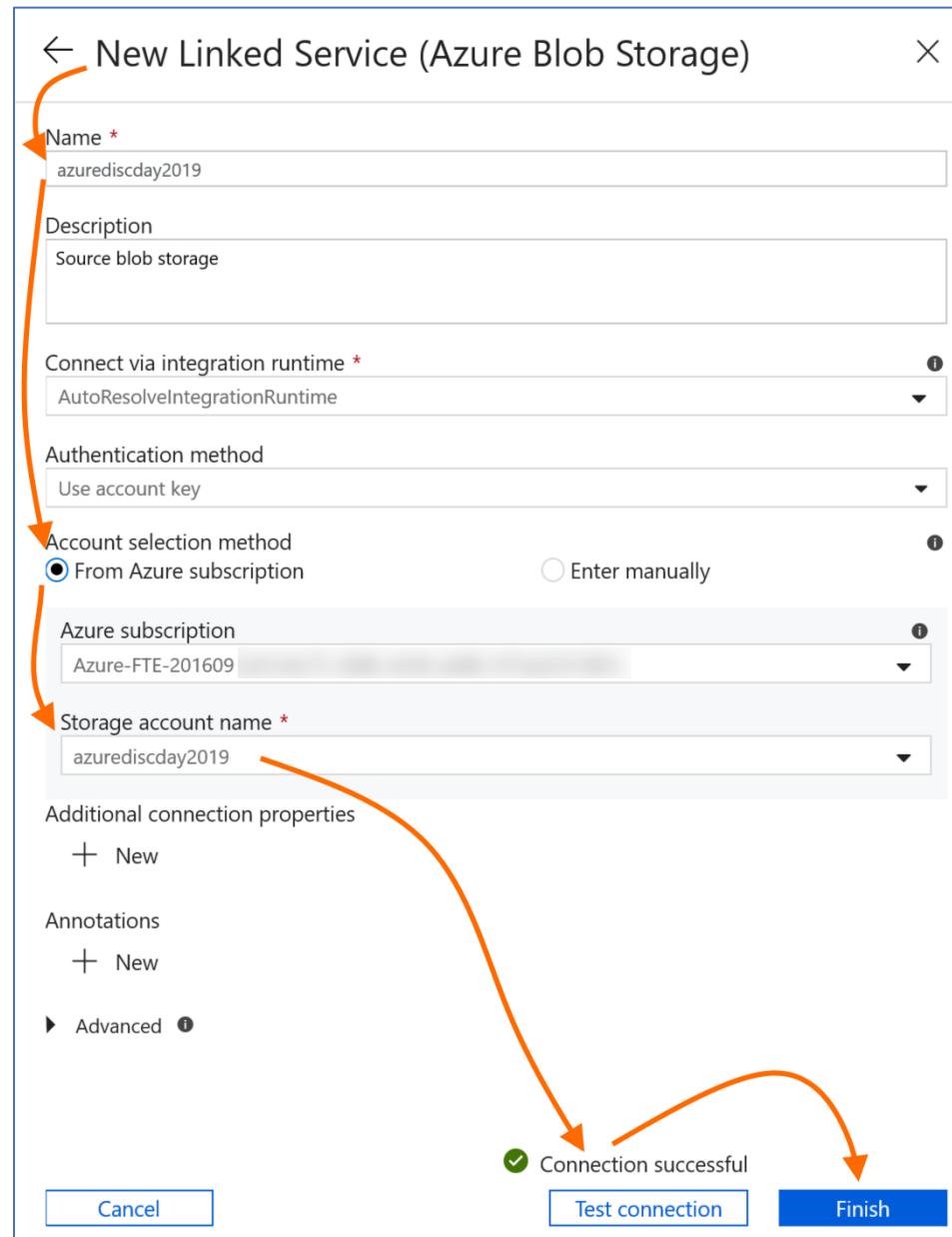
All Azure Database File Generic Protocol NoSQL Services and apps

All Filter by name + Create new connection

Next, select “Azure Blob Storage”, then click “Continue”.



Name the data source (be specific, so you'll know what it is in a list of data sources – here, we're using the storage account name). Select your Azure subscription and storage account. Optionally, click “Test connection” and ensure the connection is successful. Then click “Finish”.



Now that we have our Azure blob storage source, we need to select the specific files. Click “Next” to continue in the ADF Copy wizard.

1 Properties One time copy

2 Source Connection Dataset

3 Destination Connection Dataset

4 Settings

5 Summary

6 Deployment

Source data store

Specify the source data store for the copy task. You can use an existing connection or create a new one.

All Azure Database File Generic Protocol

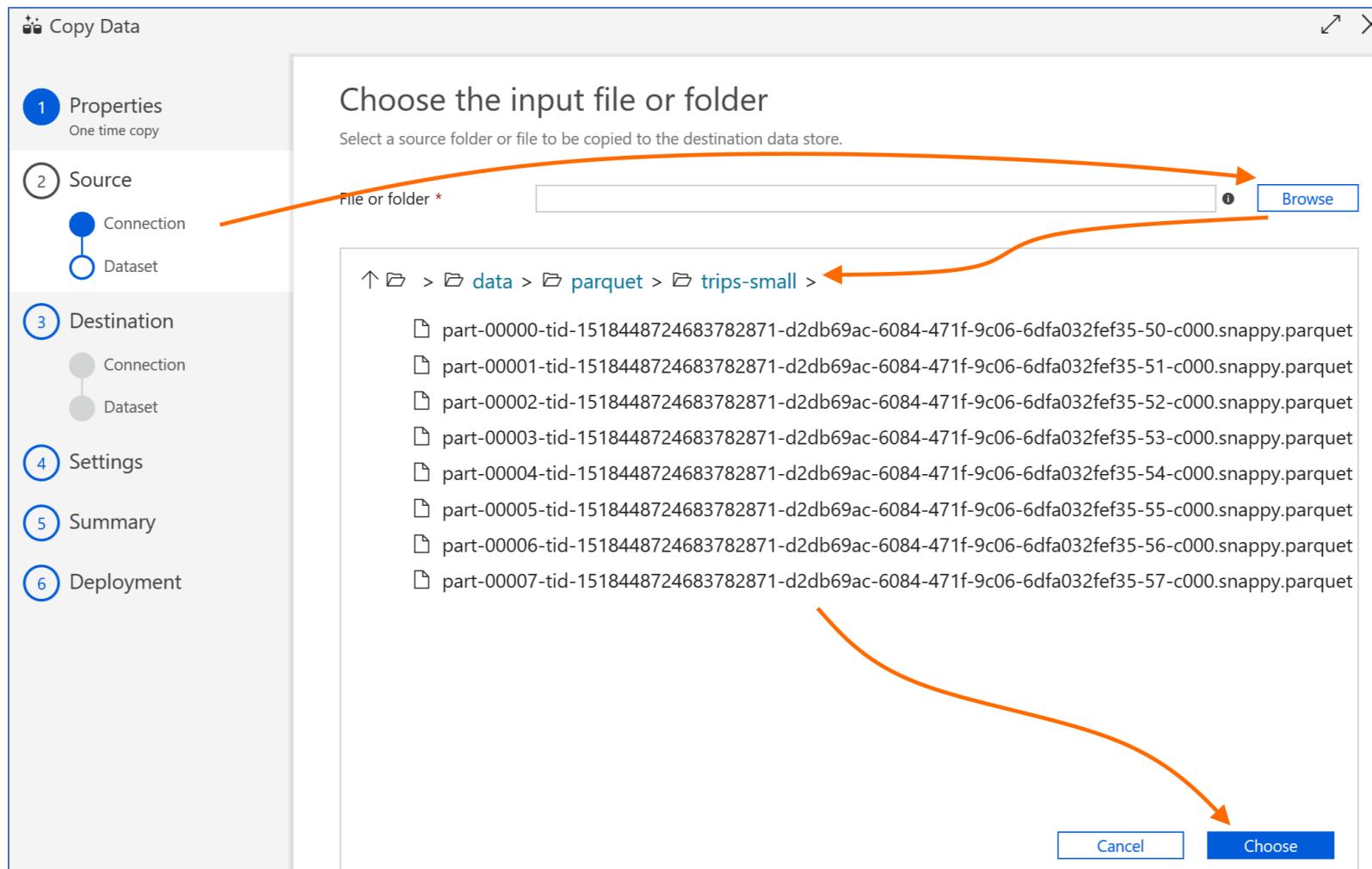
All Filter by name

 azurediscday2019



Previous Next

Click “Browse”, then use the folder list to navigate into the folder where your Parquet files were written at the end of lab 1. Then click “Choose”.



Check “Copy file recursively” (since our Parquet data is spread across multiple files and we want to read all files in the selected folder), then click “Next”.

Copy Data

1 Properties One time copy

2 Source Connection
Dataset

3 Destination Connection
Dataset

4 Settings

5 Summary

6 Deployment

Choose the input file or folder

Select a source folder or file to be copied to the destination data store.

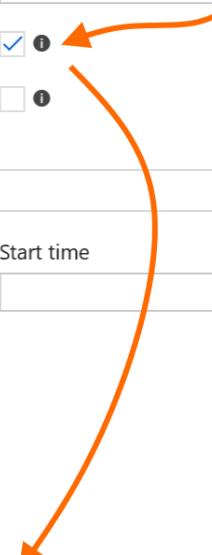
File or folder *

Copy file recursively

Binary Copy

Compression Type

Filter by last modified Start time End time




Previous

Set “File format” to “Parquet format”. The Preview tab automatically shows a preview of the data for a quick visual check. Optionally, you can also click the “Schema” tab to validate the file schema compared to the files you worked with in lab 1. Then, click “Next”.

Copy Data

1 Properties One time copy

2 Source Connection Dataset

3 Destination Connection Dataset

4 Settings

5 Summary

6 Deployment

File format settings

File format

Parquet format

Preview Schema

trip_type	trip_year	trip_month	taxis_type	vendor_id	pickup_datetime	dropoff_datetime	passenger_count	trip_distance
0	2017	01	yellow	1	2017-01-09T11:13:28	2017-01-09T11:25:45	1	3.3
0	2017	01	yellow	2	2017-01-01T00:00:05	2017-01-01T00:15:36	1	8.47

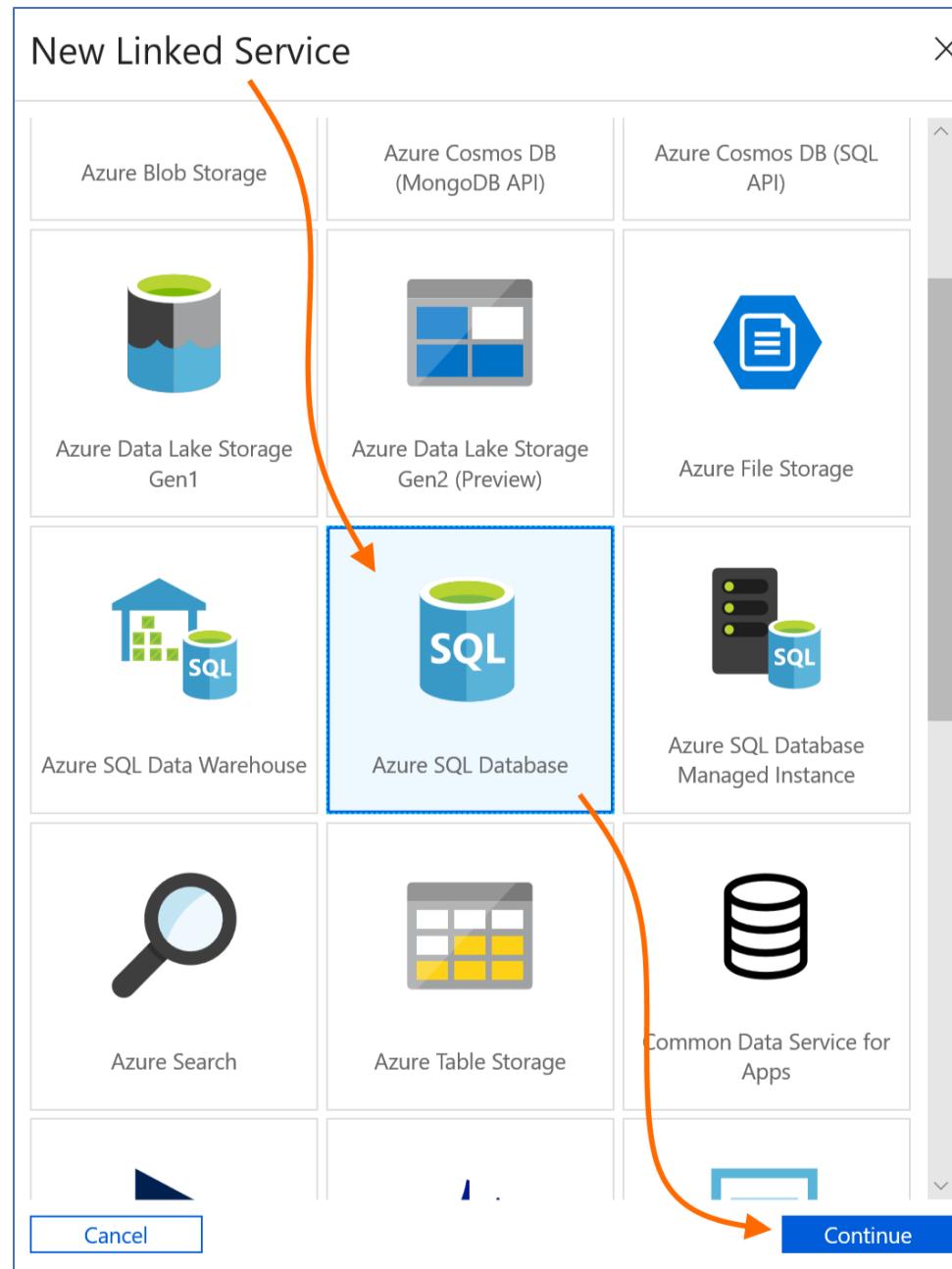
Previous Next

The screenshot shows the 'Copy Data' interface in Azure Data Factory. On the left, a vertical navigation pane lists six steps: 1. Properties (selected), 2. Source, 3. Destination, 4. Settings, 5. Summary, and 6. Deployment. Step 1 has a sub-section 'One time copy'. Step 2 has icons for 'Connection' and 'Dataset'. Step 3 has icons for 'Connection' and 'Dataset'. Step 4 is labeled 'Settings'. Step 5 is labeled 'Summary'. Step 6 is labeled 'Deployment'. A modal window titled 'File format settings' is open over the navigation pane. It contains a 'File format' dropdown set to 'Parquet format'. Below it are two tabs: 'Preview' (which is selected) and 'Schema'. The 'Preview' tab displays a table with nine columns: trip_type, trip_year, trip_month, taxis_type, vendor_id, pickup_datetime, dropoff_datetime, passenger_count, and trip_distance. Two rows of data are shown: one for trip_type 0 (yellow taxi, vendor_id 1) and one for trip_type 0 (yellow taxi, vendor_id 2). At the bottom of the preview table are 'Previous' and 'Next' buttons, with 'Next' being highlighted. A large orange arrow points from the 'Properties' step in the sidebar to the 'File format' dropdown. Another large orange arrow points from the 'Next' button in the preview table to the 'Next' button in the bottom navigation bar.

Next, you need to configure a destination, which is the Azure SQL Database you created earlier in this lab. In the “Destination” stage, click “+ Create new connection”, similarly to how you created a source data connection previously.

The screenshot shows the 'Copy Data' wizard interface. On the left, a vertical navigation pane lists steps: 1 Properties (One time copy), 2 Source (Azure Blob Storage), 3 Destination (Connection and Dataset). Step 3 is currently selected. The main area is titled 'Destination data store' with the sub-instruction: 'Specify the destination data store for the copy task. You can use an existing data store connection or specify a new data store.' Below this are tabs: All (selected), Azure, Database, File, Generic Protocol, NoSQL, and Services and apps. A search bar includes 'All' and 'Filter by name'. A list shows a single item: 'azurediscday2019'. To the right of the list is a '+ Create new connection' button with an orange arrow pointing towards it from the 'Destination' step in the sidebar.

Select “Azure SQL Database”, then click “Continue”.



Name the data source. Select your Azure subscription, then the SQL server and Azure SQL DB database name. Enter the credentials (username and password) you specified when creating the database earlier in this lab. Optionally, click “Test connection” and verify that it is successful, then click “Finish”.

← New Linked Service (Azure SQL Database) X

Name * (arrow pointing to this field)

Description

Connect via integration runtime * (arrow pointing to this field)

Account selection method From Azure subscription Enter manually (arrow pointing to this radio button)

Azure subscription (arrow pointing to this field)

Server name * (arrow pointing to this field)

Database name * (arrow pointing to this field)

Authentication type * (arrow pointing to this field)

User name * (arrow pointing to this field)

Password (arrow pointing to this field)

Azure Key Vault

Cancel (arrow pointing from here to the Test connection button) Test connection (checkmark icon) Connection successful (arrow pointing to this button) Finish

You have now created a source (Parquet files from lab 1) and a destination data set (your Azure SQL database). Click “Next” to move to schema mapping.

Copy Data

1 Properties One time copy

2 Source Azure Blob Storage

- Connection
- Dataset

3 Destination

- Connection
- Dataset

4 Settings

5 Summary

6 Deployment

Destination data store

Specify the destination data store for the copy task. You can use an existing data store connection or specify a new one.

All Azure Database File Generic Protocol NoSQL Services and apps

All Filter by name + Create

azurediscday2019 TaxiDb

Previous Next



Select the destination table in your Azure SQL DB, then click “Next”.

Copy Data

1 Properties One time copy

2 Source Azure Blob Storage

Connection
Dataset

3 Destination

Connection
Dataset

4 Settings

5 Summary

6 Deployment

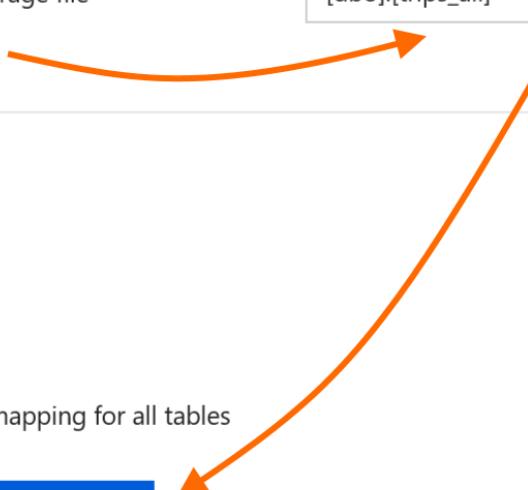
Table mapping

For each table you have selected to copy in the source data store, select a corresponding table in the destination database to run at the destination.

Source	Destination
Azure Blob Storage file	→ [dbo].[trips_all]

Skip column mapping for all tables

Previous Next



Review the column mappings – no changes should be needed here – then click “Next”.

Copy Data

1 Properties
One time copy

2 Source
Azure Blob Storage
Connection
Dataset

3 Destination
Connection
Dataset

4 Settings

5 Summary

6 Deployment

Column mapping

Choose how source and destination columns are mapped

Table mappings (1)

<input checked="" type="checkbox"/> Source Azure Blob Storage file Destination [dbo].[trips_all]	<input checked="" type="checkbox"/> Azure Blob Storage file [dbo].[trips_all]
---	--

Column mappings

<input checked="" type="checkbox"/> trip_type (Int32)	→ trip_type (Int32)
<input checked="" type="checkbox"/> trip_year (String)	→ trip_year (Int32)
<input checked="" type="checkbox"/> trip_month (String)	→ trip_month (String)
<input checked="" type="checkbox"/> taxi_type (String)	→ taxi_type (String)
<input checked="" type="checkbox"/> vendor_id (Int32)	→ vendor_id (Int32)
<input checked="" type="checkbox"/> pickup_datetime (DateTime)	→ pickup_datetime (DateTime)
<input checked="" type="checkbox"/> ehail_fee (Double)	→ ehail_fee (Double)
<input checked="" type="checkbox"/> total_amount (Double)	→ total_amount (Double)
<input checked="" type="checkbox"/> fare_amount (Double)	→ fare_amount (Double)

Azure SQL Database sink properties

Pre-copy script

Write batch size
10000

Next

The screenshot shows the 'Column mapping' step of a 'Copy Data' wizard. On the left, a sidebar lists steps 1 through 6. Step 2, 'Source', is selected and shows 'Azure Blob Storage' with icons for 'Connection' and 'Dataset'. Step 3, 'Destination', is also shown with its icons. Steps 4 through 6 are listed below. The main area has a title 'Column mapping' with the sub-instruction 'Choose how source and destination columns are mapped'. It shows 'Table mappings (1)' with a single row mapping from 'Source' (Azure Blob Storage file) to 'Destination' ([dbo].[trips_all]). Below this is a large 'Column mappings' table with 10 rows, each mapping a source column to a destination column. The source columns are: trip_type (Int32), trip_year (String), trip_month (String), taxi_type (String), vendor_id (Int32), pickup_datetime (DateTime), ehail_fee (Double), total_amount (Double), and fare_amount (Double). The destination columns are: trip_type (Int32), trip_year (Int32), trip_month (String), taxi_type (String), vendor_id (Int32), pickup_datetime (DateTime), ehail_fee (Double), total_amount (Double), and fare_amount (Double). At the bottom, there are 'Azure SQL Database sink properties' with fields for 'Pre-copy script' (empty) and 'Write batch size' (set to 10000). Two orange arrows highlight the 'Source' selection in the sidebar and the 'Column mappings' table.

Next, on the tabs for “4 – Settings” and “5 – Summary”, accept the defaults and click “Next” until the “Deployment” tab is reached and the copy activity runs. You can monitor its progress by clicking “Monitor”, or close the copy wizard by clicking “Finish”.

Copy Data

1 Properties One time copy

2 Source Azure Blob Storage

Connection

Dataset

3 Destination Azure SQL Database

Connection

Dataset

4 Settings

5 Summary

6 Deployment

The screenshot shows the "Copy Data" wizard in the Azure portal. The left sidebar lists steps 1 through 6. Step 6, "Deployment", is currently selected, indicated by a blue border. The main area displays a flow diagram from "Azure Blob Storage" to "Azure SQL Database". Below the diagram, the text "Deployment complete" is displayed, followed by a list of successful tasks: "Creating Datasets" (green checkmark), "Creating Pipelines" (green checkmark), and "Running Pipelines" (green checkmark). A message states: "Datasets and pipelines have been created. You can now monitor and edit the copy pipelines or click finish to close the copy wizard." At the bottom, there are three buttons: "Edit Pipeline", "Monitor" (highlighted with a blue arrow), and "Finish".

Azure Blob Storage → Azure SQL Database

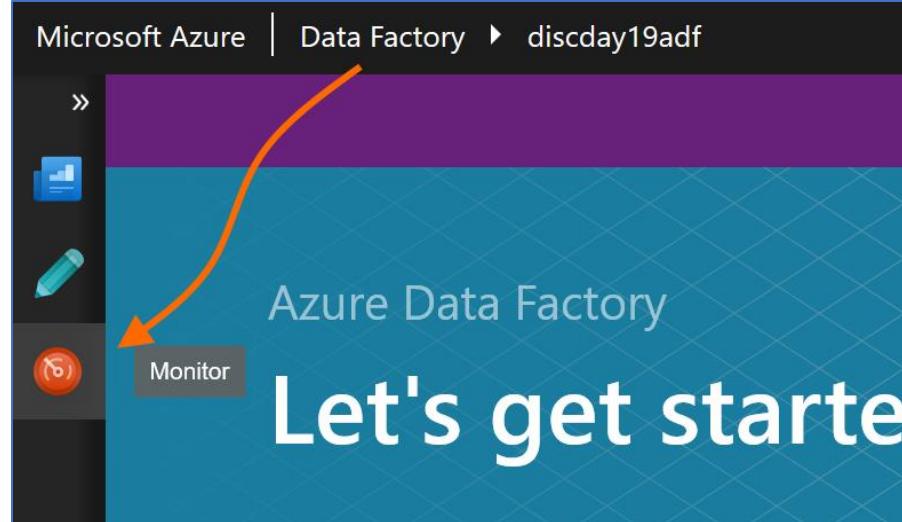
Deployment complete

- ▶ Creating Datasets ✓
- ▶ Creating Pipelines ✓
- ▶ Running Pipelines ✓

Datasets and pipelines have been created. You can now monitor and edit the copy pipelines or click finish to close the copy wizard.

Edit Pipeline Monitor Finish

You can monitor pipeline progress anytime by clicking “Monitor”.



In the Monitor view, you can drill into a pipeline’s run history (current and past) by clicking the “View Activity Runs” icon next to its name.

A screenshot of the Microsoft Azure Data Factory Monitor view. The top navigation bar shows 'Microsoft Azure | Data Factory > discday19adf'. The sidebar includes 'Dashboards', 'Pipeline Runs' (selected), 'Trigger Runs', 'Integration Runtimes', 'Alerts & Metrics', 'Run', 'Cancel', and 'Refresh'. Below the navigation is a search bar with 'Last 24 Hours' (01/27/2019 8:22 PM - 01/28/2019 8:22 PM), 'Time Zone' (UTC-05:00 Eastern Time (US & C...)), and a 'View All Rerun History' toggle. The main area shows a table of pipelines. The first row is for 'CopyParquetToSQL'. It has columns for Pipeline Name (with a 'View Activity Runs' button highlighted with an orange arrow), Run Start (01/28/2019, 9:21:16 PM), Duration (00:06:24), Triggered By (Manual trigger), Status (In Progress...), and Parameters. The 'Monitor' icon in the sidebar is also highlighted with an orange arrow.

You can then drill in even further to an activity run's details by clicking its "Details" icon.

The screenshot shows the Microsoft Azure Data Factory Pipeline Runs page. At the top, there are navigation links: Dashboards, Pipeline Runs (which is selected), Trigger Runs, Integration Runtimes, and Alerts & Metrics. Below the navigation, it says "All Pipeline Runs / CopyParquetToSQL - Activity Runs". There are buttons for Rerun, Rerun from activity, and Refresh. A search bar is present. On the left, there are icons for creating a new pipeline, editing, and deleting. A red arrow points from the left margin to the "Details" icon in the table header. Another red arrow points from the "Details" icon in the table header to the "Details" icon in the row for the "Copy_jfh" activity run. The table has columns: ACTIVITY NAME, ACTIVITY TYPE, ACT (with a dropdown menu), Details, RUN START, DURATION, STATUS, and INTEGRATION RUNTIME. One row is visible for the "Copy_jfh" activity, which is a "Copy" type activity. The status is "In Progress".

ACTIVITY NAME	ACTIVITY TYPE	ACT	Details	RUN START	DURATION	STATUS	INTEGRATION RUNTIME
Copy_jfh	Copy			01/28/2019 9:21 PM	00:09:46		In Progress

Details

⟳ Refresh



Performance tuning tips:

Sink Azure SQL Database: The DTU utilization was high during the copy activity run. To achieve better performance, you are suggested to scale the database to a higher tier than the current 500 DTUs. Refer to this [document](#).

Learn more on copy performance details from here.



Azure Blob Storage

Succeeded



Azure SQL Database

Data read: 483.641 MB
Files read: 8
Rows read: 22,234,960

Data written: 3.146 GB
Rows written: 22,234,960
Throughput: 712.377 KB/s

Copy duration 00:11:03

▶ Azure Blob Storage → Azure SQL Database Queue 00:00:03 | Transfer 00:10:59

When the activity completes, it will show status of “Succeeded”.

Activity Runs

Pipeline Run ID **58e3ab53-736c-49b7-aabf-3d549f9c6302**

All Succeeded In Progress Failed Cancelled

ACTIVITY NAME	ACTIVITY TYPE	ACTIONS	SOURCE	DESTINATION	RUN START	DURATION	STATUS
Copy_jfh	Copy		data/parquet/trips-sm...	[dbo].[trips_all]	01/28/2019 9:21 PM	00:11:10	Succeeded

After the copy activity has succeeded, you can go back to your Resource Group, into your Azure SQL DB, and to the “Query editor” tab. Alternately, you can connect to your database in SQL Server Management Studio. Either way, run a simple query like the following to verify that data has been copied into your database.

```
select count(*) from dbo.trips_all;
```

Dashboard > DiscoveryDay > TaxiDb (discday19sql/TaxiDb) - Query editor (preview)

TaxiDb (discday19sql/TaxiDb) - Query editor (preview)

SQL database

Search (Ctrl+)

Login Edit Data (Preview) New Query Open query Save query Feedback

Overview

Activity log

Tags

Diagnose and solve problems

Quick start

Query editor (preview)

TaxiDb (pelazem)

Showing limited object explorer here.
For full capability please open SSDT.

Tables

Views

Stored Procedures

Run Cancel query

Query 1

1 select count(*) from dbo.trips_all;

Results Messages

Search to filter items...

122115939

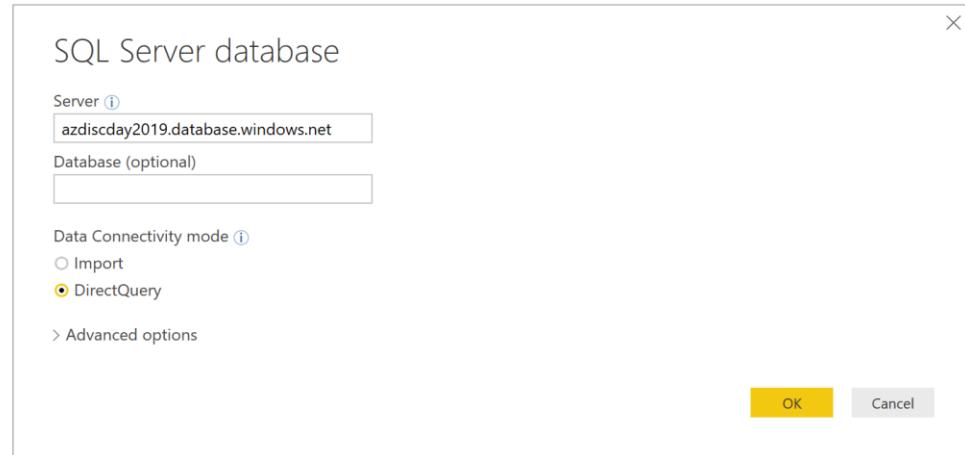
Query succeeded | 11s

Task 3 – Create Power BI Reports

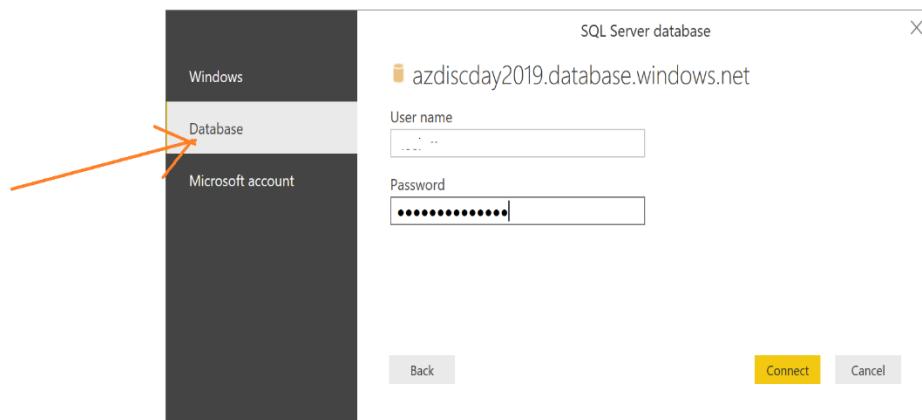
There are two options to create Open Power BI desktop application and connect to Azure sql database.

The screenshot shows two windows side-by-side. On the left is the 'Power BI Desktop' application. In the center is the 'Get Data' dialog box, which is open to the 'Azure' section. An orange arrow points from the 'Azure' section in the 'Get Data' dialog to the 'Azure SQL database' option in the list. On the right is the 'azdiscday2019' Azure SQL database page. An orange arrow points from the 'Server name' field in the main details section to the 'azdiscday2019.database.windows.net' value. Below it, another orange arrow points from the 'Connection strings' link to the 'Show database connection strings' link. At the bottom of the page is a chart titled 'Resource utilization (azdiscday2019)' showing DTU percentage over time, with a sharp drop at approximately 6:15 PM.

Enter Server name and select direct query and click ok



Select Database as the option and enter credentials



Select the view you created (which unions the batch and streaming data tables) and click “Load”.

Navigator

Display Options

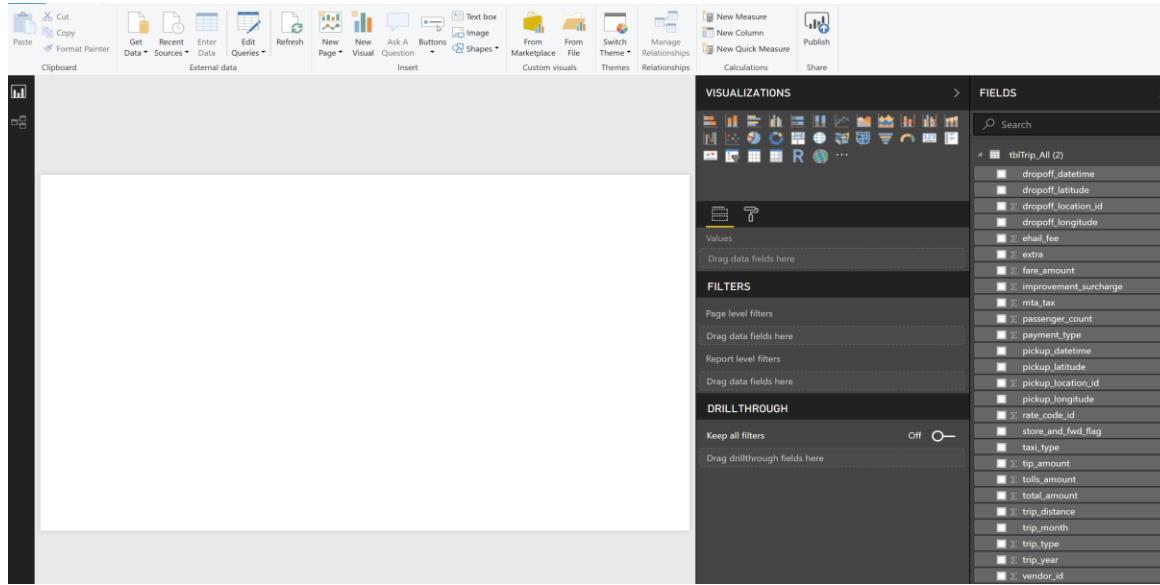
- azdiscday2019.database.windows.net [1]
 - azdiscday2019 [4]
 - sys.database_firewall_rules
 - trips
 - trips_all
 - trips_new



trips

trip_type	trip_year	trip_month	taxis_type	vendor_id	pickup_dat
null	null	null	null	null	
0	2010	01	yellow	1	1/9/2
0	2010	01	yellow	1	1/23/2
0	2010	01	yellow	1	1/7/2
0	2010	01	yellow	1	1/23/2
0	2010	01	yellow	1	1/20/2
0	2010	01	yellow	1	1/29/2
0	2010	01	yellow	1	1/15/2
0	2010	01	yellow	1	1/26/2
0	2010	01	yellow	1	1/11/2
0	2010	01	yellow	1	1/26/2
0	2010	01	yellow	1	1/28/2
0	2010	01	yellow	1	1/18/2
0	2010	01	yellow	1	1/17/2
0	2010	01	yellow	1	1/18/2
0	2010	01	yellow	1	1/4/2
0	2010	01	yellow	1	1/10/2

i The data in the preview has been truncated due to size limits.



Create a Power BI Chart → Total passengers by year.

Drag year into Axis and passengers count in values.

The screenshot shows the Power BI desktop interface. On the left, a bar chart titled "passenger_count by trip_year" displays a single teal bar for the year 2010, reaching approximately 60K on the y-axis. The y-axis is labeled with "OK", "10K", "20K", "30K", "40K", "50K", "60K", and "70K". The x-axis is labeled "2010".

The top ribbon menu includes: Clipboard, External data, Insert, Custom visuals, Themes, Relationships, Calculations, and Share.

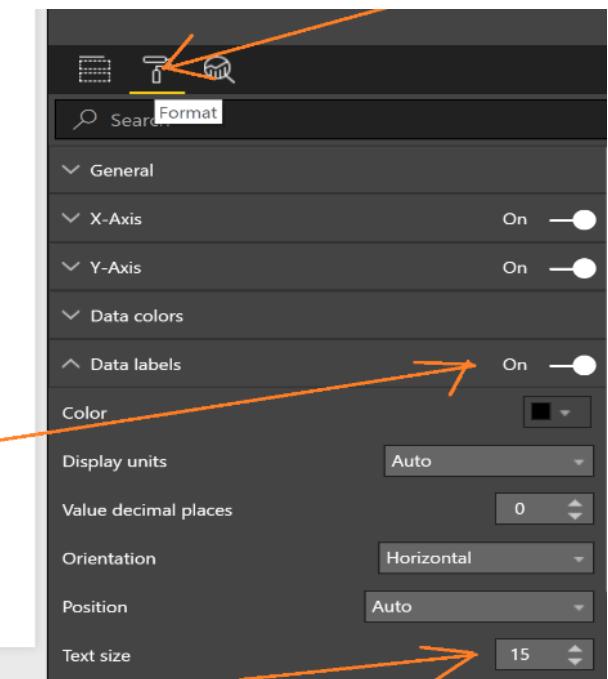
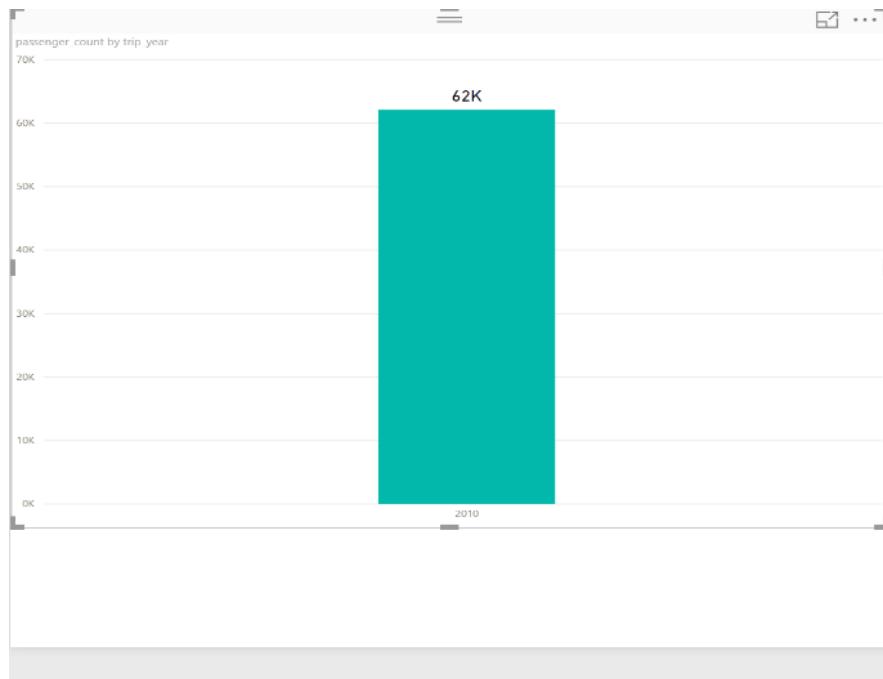
The Visualizations pane on the right lists various chart types, with "Line and stacked column chart" selected and highlighted with a yellow box and an orange arrow pointing to it from the top-left.

The Fields pane on the right shows the data source "vwTripData" with the following fields listed:

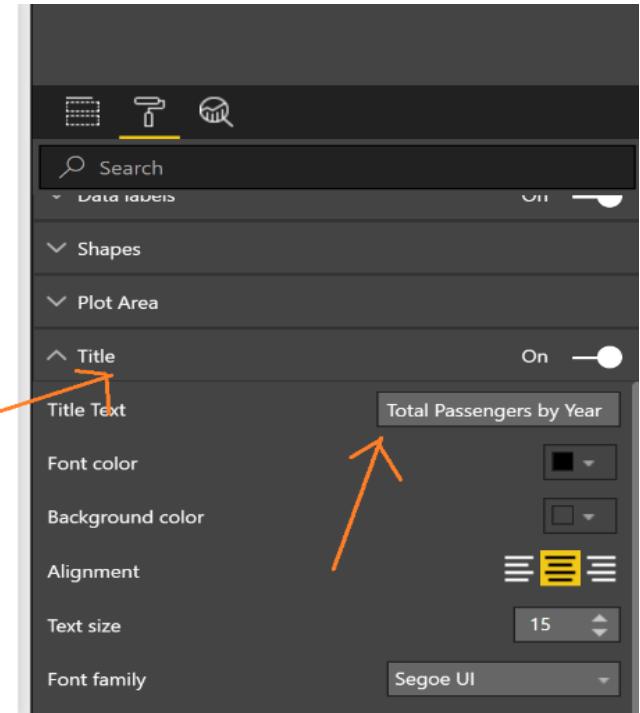
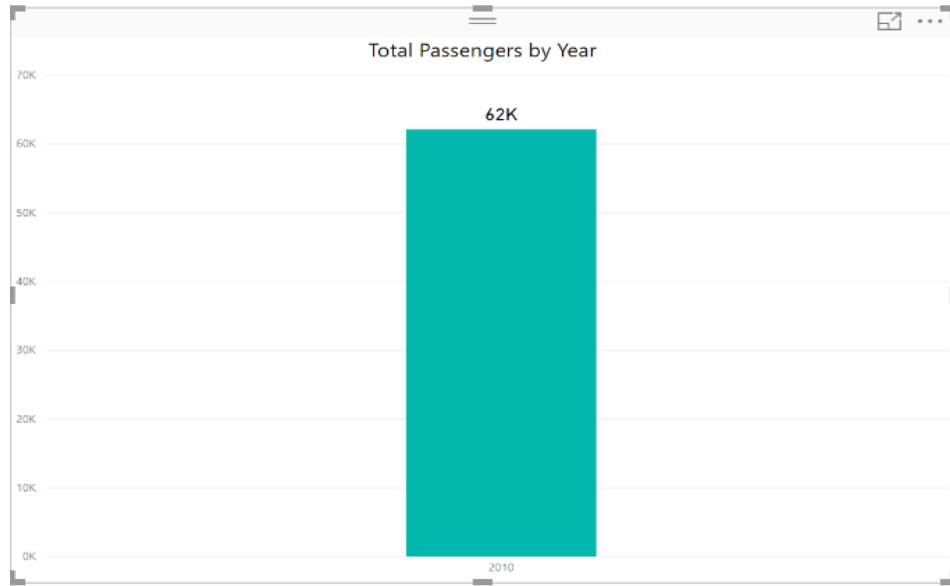
- checkbox dropoff_datetime
- checkbox dropoff_latitude
- checkbox dropoff_location_id
- checkbox dropoff_longitude
- checkbox ehail_fee
- checkbox extra
- checkbox fare_amount
- checkbox improvement_surcharge
- checkbox mta_tax
- checkbox passenger_count** (highlighted with a yellow box and an orange arrow)
- checkbox payment_type
- checkbox pickup_datetime
- checkbox pickup_latitude
- checkbox pickup_location_id
- checkbox pickup_longitude
- checkbox rate_code_id
- checkbox store_and_fwd_flag
- checkbox taxi_type
- checkbox tip_amount
- checkbox tolls_amount
- checkbox total_amount

The "Visualizations" pane also contains sections for "Shared axis", "Column series", "Column values", "Line values", and "Toolips", each with a "Drag data fields here" placeholder.

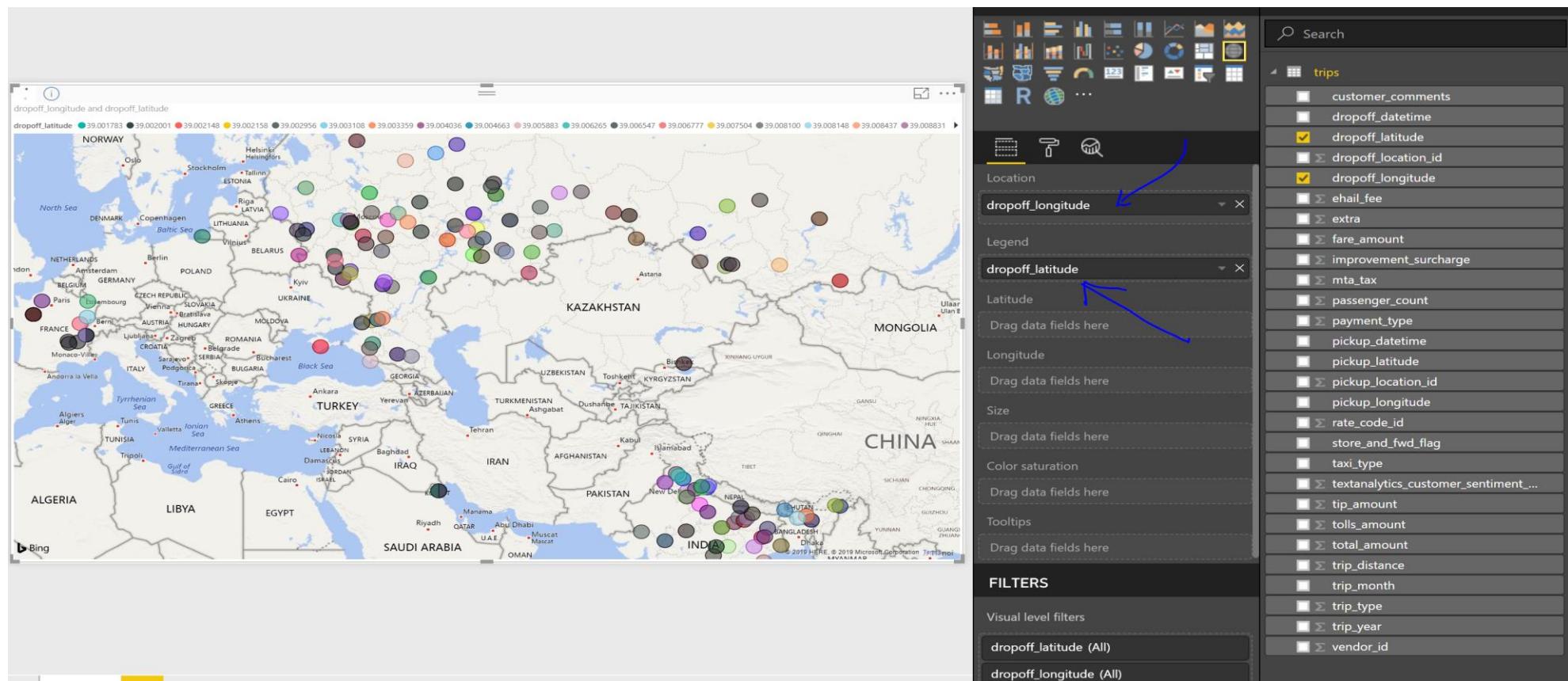
Please format labels using below options –



Rename chart title –

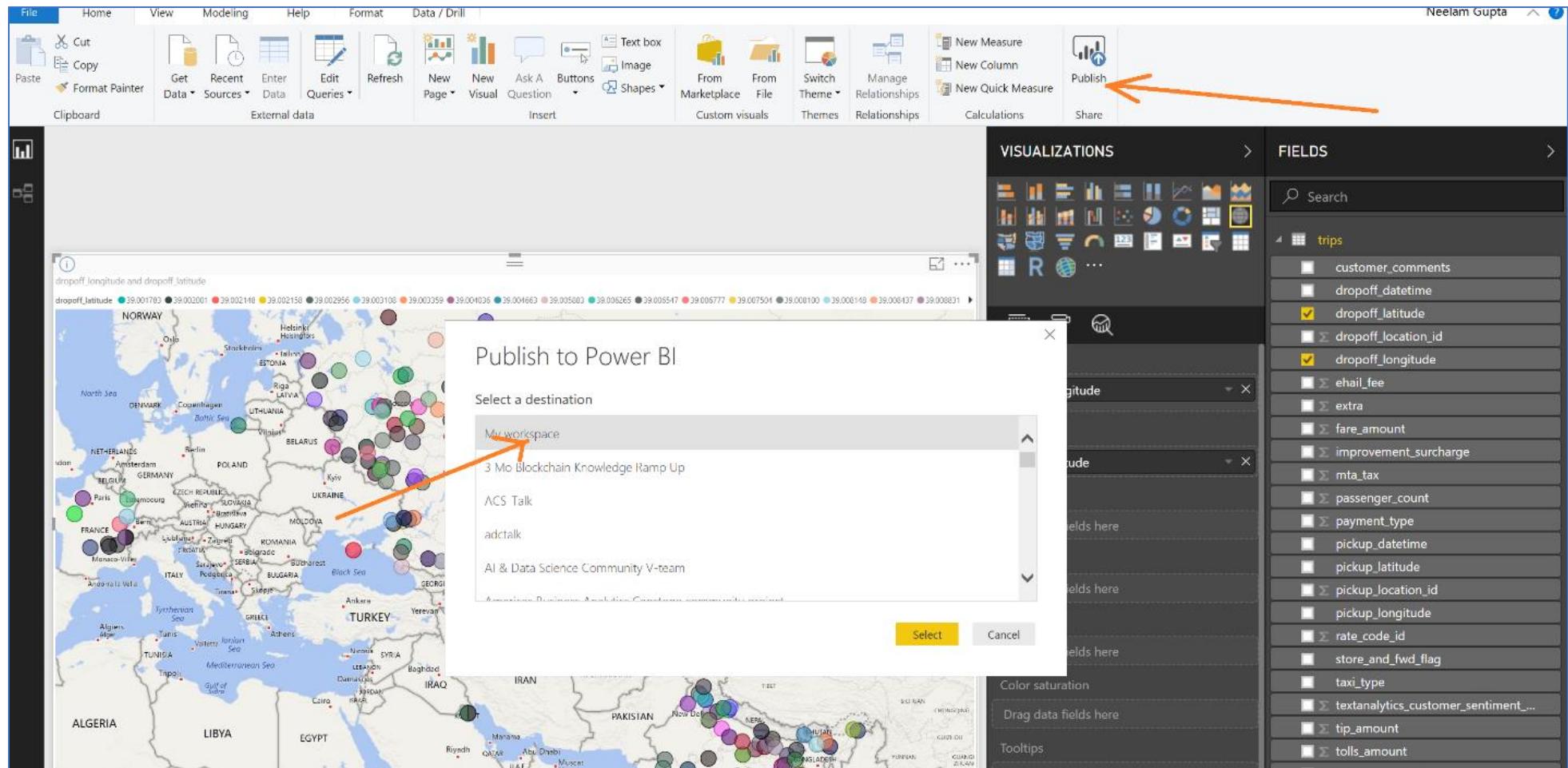


Create Map chart.



After creating all reports, deploy to App.PowerBI.com

Click Publish on the ribbon and then select workspace under which you want to deploy reports.



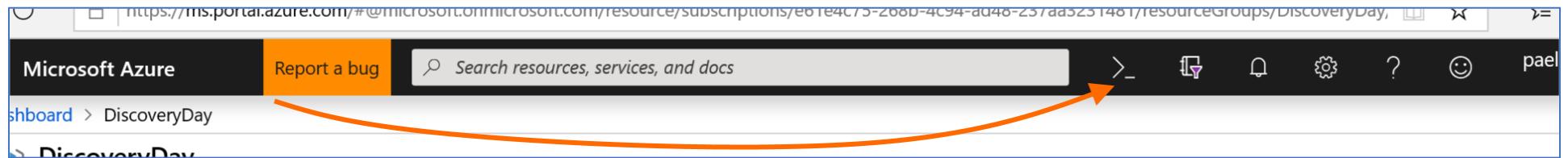
Conclusion

You have now completed lab 2. Great job! Now, it's time to switch to the streaming data path in labs 3 and 4.

Appendix

In case you encounter difficulties copying the Parquet data into Azure SQL DB, this is an alternate approach to provisioning your Azure SQL DB. We have provided a .bacpac export of a small database (2018 data only).

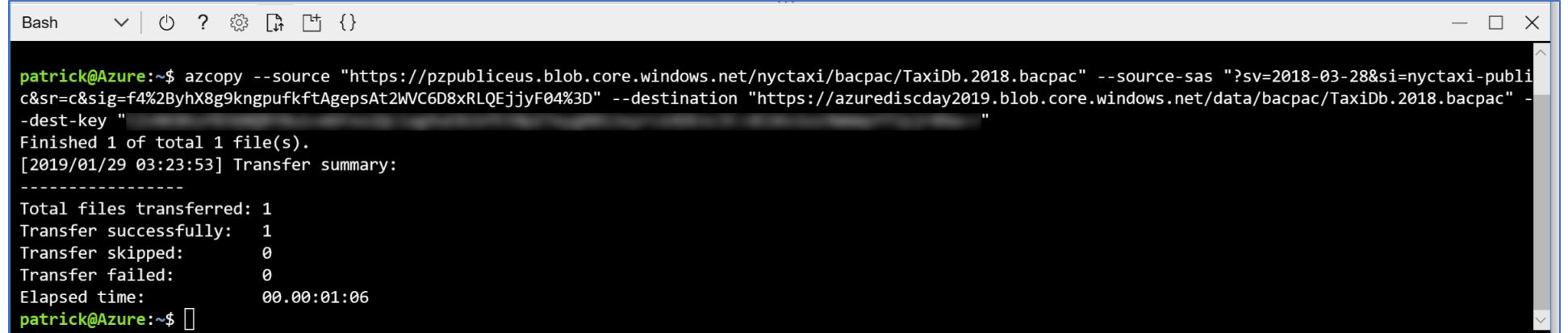
In your Resource Group, click the Cloud Shell icon in the top bar.



In the following command, substitute your Azure storage account name and key for the two tokens “{YOUR STORAGE ACCOUNT NAME}” and “{YOUR STORAGE ACCOUNT KEY}”. Substitute your storage container name for the token “{YOUR CONTAINER NAME}”.

```
azcopy --source "https://pzpubliceus.blob.core.windows.net/nyctaxi/bacpac/TaxiDb.2018.bacpac" --source-sas "?sv=2018-03-28&si=nyctaxi-public&sr=c&sig=f4%2Byhx8g9kngpufkftAgepsAt2WVC6D8xRLQEjjyF04%3D" --destination "https://{{YOUR STORAGE ACCOUNT NAME}}.blob.core.windows.net/{{YOUR CONTAINER NAME}}/bacpac/TaxiDb.2018.bacpac" --dest-key "{{YOUR STORAGE ACCOUNT KEY}}"
```

Copy the updated command and paste it onto the cloud shell prompt. Hit Enter and wait for the copy to complete.



```
Bash ▾ | ⌂ ? ⌘ ⌚ ⌙ { }
```

```
patrick@Azure:~$ azcopy --source "https://pzpublicus.blob.core.windows.net/nyctaxi/bacpac/TaxiDb.2018.bacpac" --source-sas "?sv=2018-03-28&si=nyctaxi-publi  
c&sr=c&sig=f4%2ByhX8g9kngpufkftAeppsAt2WVC6D8xRLQEjjyF04%3D" --destination "https://azurediscday2019.blob.core.windows.net/data/bacpac/TaxiDb.2018.bacpac" -  
-dest-key "  
Finished 1 of total 1 file(s).  
[2019/01/29 03:23:53] Transfer summary:  
-----  
Total files transferred: 1  
Transfer successfully: 1  
Transfer skipped: 0  
Transfer failed: 0  
Elapsed time: 00.00:01:06  
patrick@Azure:~$
```

You now have a .bacpac file in your storage container, in a new bacpac folder.

Next, return to your Resource Group. Click on the SQL Server resource that was created when you created an Azure SQL DB. In its “Overview” view click “Import database”.

The screenshot shows the Azure portal interface for managing an Azure SQL Server. The left sidebar lists several options: Overview (selected), Activity log, Access control (IAM), Tags, Diagnose and solve problems, Settings (with Quick start, Failover groups, and Manage Backups), Notifications (0), Features (6), and tabs for All, Security (4), Performance (1), and Recovery (1). The main content area displays the Overview of the 'discday19sql' SQL server. It includes details such as Resource group (DiscoveryDay), Status (Available), Location (East US), Subscription (Azure-FTE-201609), Subscription ID (redacted), and Tags (Click here to add tags). At the top right, there are buttons for New database, New pool, New data warehouse, Import database (highlighted with a red arrow), Reset password, Move, Delete, and Feedback. Below the main content are two cards: 'Active Directory admin' and 'Advanced Data Security'.

On the Import Database view, select your subscription. Set an appropriate performance level for the database; since you'll be querying it in Power BI but will not be doing bulk import, P2 should be enough (and you can always scale Azure SQL DB down or up without downtime).

After you click on your storage account, you will need to further navigate into the container and to the folder where you copied the .bacpac file above, then click on the .bacpac file and click “Select” in order to complete this Import database screen.

The screenshot shows two overlapping windows from the Azure portal. The left window is titled 'Containers' and lists blobs in the 'azurediscday2019' container. The right window is titled 'bacpac' and shows a list of bacpac files in a folder. Orange arrows highlight the 'data' blob in the container list and point to the 'TaxiDb.2018.bacpac' file in the bacpac list, indicating they are being selected for import.

Containers
azurediscday2019

+ Container Refresh

Search containers by prefix

NAME	LAST MODIFIED	PUBLIC ACCESS L...	LEASE STATE	...
azure-webjobs-eventhub	1/16/2019, 8:55:29 AM	Private	Available	...
azure-webjobs-hosts	1/16/2019, 8:50:57 AM	Private	Available	...
azure-webjobs-secrets	1/16/2019, 8:51:16 AM	Private	Available	...
data	1/7/2019, 4:14:09 PM	Private	Available	...
eh2capture	1/15/2019, 5:19:15 PM	Private	Available	...
insights-logs-operationallogs	1/16/2019, 9:13:14 AM	Private	Available	...
insights-metrics-pt1m	1/16/2019, 8:40:31 AM	Private	Available	...

bacpac
Folder

Upload Refresh

Location: data / bacpac

Search blobs by prefix (case-sensitive)

NAME
[...]
TaxiDb.2018.bacpac

Select

Click “OK” and wait until the import is complete (see Notifications). You should now be able to query the imported database in the Azure portal query editor or in SQL Server Management Studio, as shown previously.