# Get started with Real-Time Intelligence in Microsoft Fabric

Microsoft Fabric provides a runtime that you can use to store and query data by using Kusto Query Language (KQL). Kusto is optimized for data that includes a time series component, such as real-time data from log files or IoT devices.

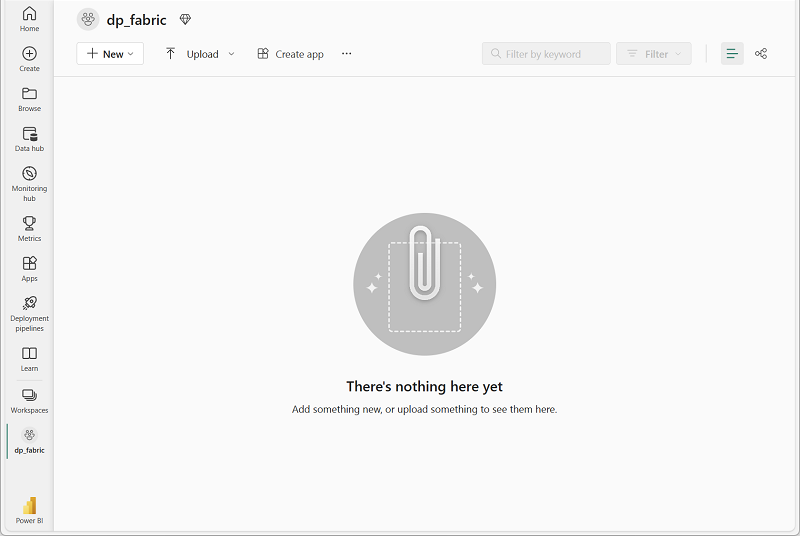
This lab takes approximately **30** minutes to complete.

**Note**: You need a [Microsoft Fabric trial](https://learn.microsoft.com/fabric/get-started/fabric-trial) to complete this exercise.

Create a workspace

Before working with data in Fabric, create a workspace with the Fabric trial enabled.

1. On the [Microsoft Fabric home page](https://app.fabric.microsoft.com/home?experience=fabric) at https://app.fabric.microsoft.com/home?experience=fabric, select **Real-Time Intelligence**.
2. In the menu bar on the left, select **Workspaces** (the icon looks similar to 🗇).
3. Create a new workspace with a name of your choice, selecting a licensing mode that includes Fabric capacity (*Trial*, *Premium*, or *Fabric*).
4. When your new workspace opens, it should be empty.

[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/new-workspace.png)

Download file for KQL database

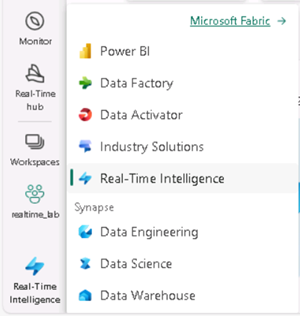
Now that you have a workspace, it’s time to download the data file you’re going to analyze.

1. Download the data file for this exercise from <https://raw.githubusercontent.com/MicrosoftLearning/dp-data/main/sales.csv>, saving it as **sales.csv** on your local computer (or lab VM if applicable)
2. Return to the browser window with the **Microsoft Fabric** Experience.

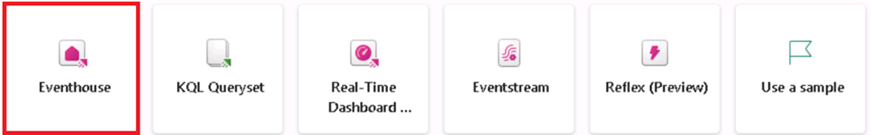
Create a KQL database

Kusto query language (KQL) is used to query static or streaming data in a table that is defined in a KQL database. To analyze the sales data, you must create a table in a KQL database and ingest the data from the file.

1. At the bottom left of the portal, switch to the Real-Time Intelligence experience.

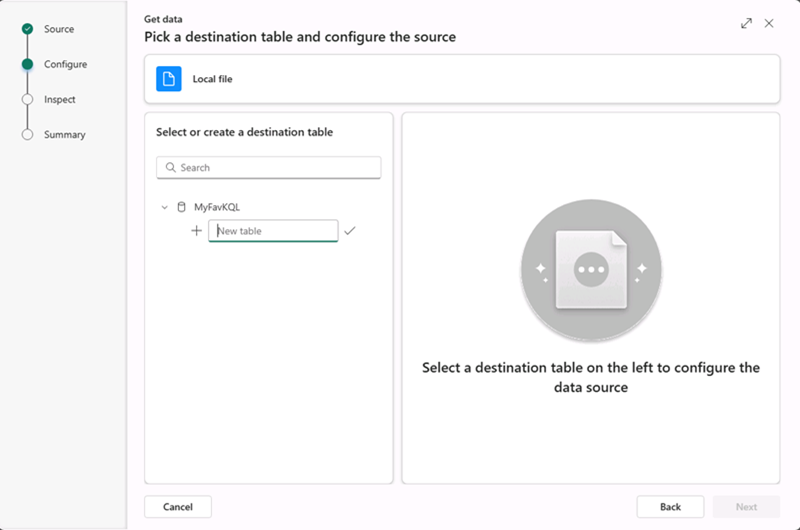
[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/fabric-real-time.png)

1. In the Real-Time Intelligence home page, create a new **Eventhouse** with a name of your choice.

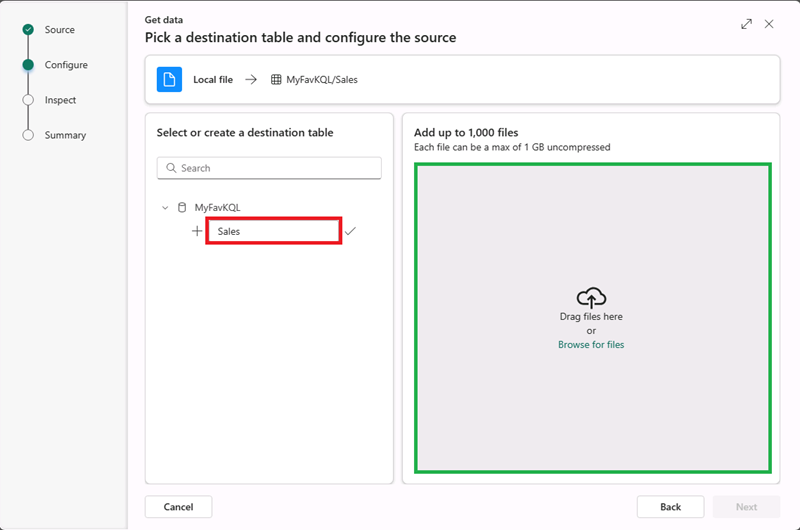
[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/create-kql-db.png)

The Eventhouse is used to group and manage your databases across projects. An empty KQL database is automatically created with the eventhouse’s name.

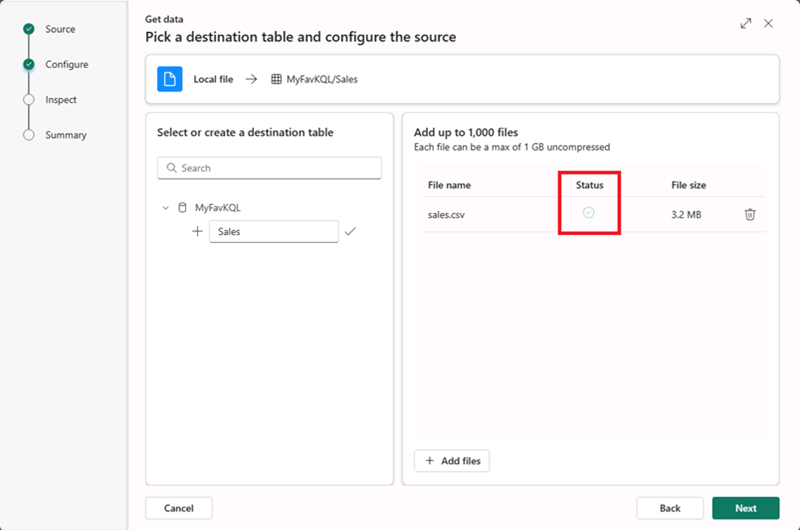
1. Once the new database has been created, select it from the list on the left under KQL Databases. Then select the option to get data from **Local File**. Use the wizard to import the data into a new table by selecting the following options:
   * **Destination**:
     + **Database**: *The database you created is already selected*
     + **Table**: *Create a new table named* **sales** by clicking on the + sign to the left of ***New table***

[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/import-wizard-local-file-1.png?raw=true)

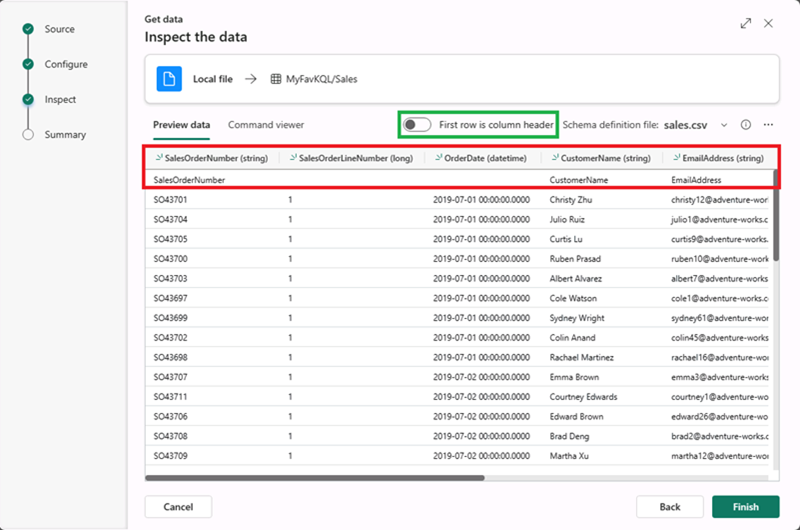
* + - Youll now see the **Drag files here or a Browse for files** hyperlink appear in the same window.

[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/import-wizard-local-file-2.png?raw=true)

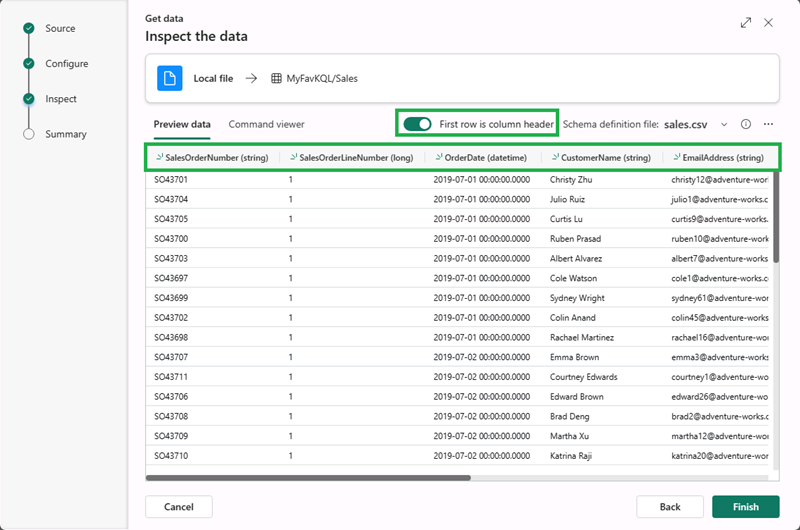
* + - browse or drag your **sales.csv** onto the screen and wait for the Status box to change to a green check box and then select **Next**

[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/import-wizard-local-file-3.png?raw=true)

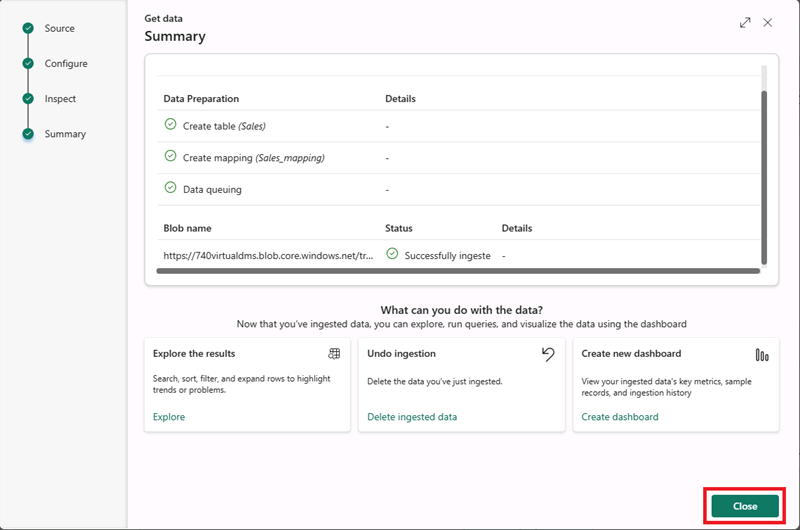
* + - In this screen you’ll see that your column headings are in the first row although the system detected them, we still need to move the slider above these lines **First row is column header** in order to get keep from getting any errors.

[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/import-wizard-local-file-4.png?raw=true)

* + - Once you select this slider you will see everything looks good to go, select the **Finish** button on the bottom right of the panel.

[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/import-wizard-local-file-5.png?raw=true)

* + - Wait for the steps in the summary screen to complete which include:
      * Create table (sales)
      * create mapping (sales\_mapping)
      * Data queuing
      * Ingestion
    - Select the **Close** button

[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/import-wizard-local-file-6.png?raw=true)

**Note**: In this example, you imported a very small amount of static data from a file, which is fine for the purposes of this exercise. In reality, you can use Kusto to analyze much larger volumes of data; including real-time data from a streaming source such as Azure Event Hubs.

Use KQL to query the sales table

Now that you have a table of data in your database, you can use KQL code to query it.

1. Make sure you have the **sales** table highlighted. From the menu bar, select the **Query table** drop-down, and from there select **Show any 100 records** .
2. A new pane will open with the query and its result.
3. Modify the query as follows:

codeCopy

sales

| where Item == 'Road-250 Black, 48'

1. Run the query. Then review the results, which should contain only the rows for sales orders for the *Road-250 Black, 48* product.
2. Modify the query as follows:

codeCopy

sales

| where Item == 'Road-250 Black, 48'

| where datetime\_part('year', OrderDate) > 2020

1. Run the query and review the results, which should contain only sales orders for *Road-250 Black, 48* made after 2020.
2. Modify the query as follows:

codeCopy

sales

| where OrderDate between (datetime(2020-01-01 00:00:00) .. datetime(2020-12-31 23:59:59))

| summarize TotalNetRevenue = sum(UnitPrice) by Item

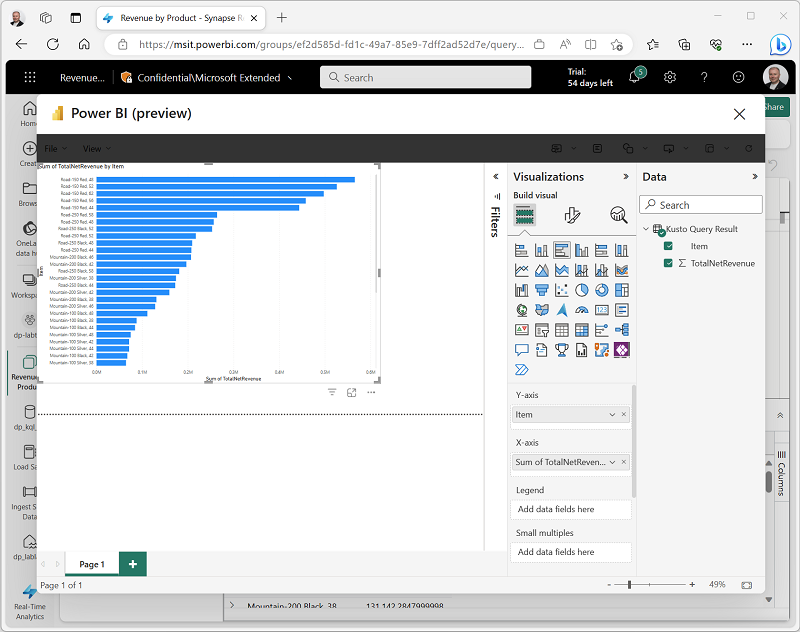
| sort by Item asc

1. Run the query and review the results, which should contain the total net revenue for each product between January 1st and December 31st 2020 in ascending order of product name.
2. Select **Save as KQL queryset** and save the query as **Revenue by Product**.

Create a Power BI report from a KQL Queryset

You can use your KQL Queryset as the basis for a Power BI report.

1. In the query workbench editor for your query set, run the query and wait for the results.
2. Select **Build Power BI report** and wait for the report editor to open.
3. In the report editor, in the **Data** pane, expand **Kusto Query Result** and select the **Item** and **TotalRevenue** fields.
4. On the report design canvas, select the table visualization that has been added and then in the **Visualizations** pane, select **Clustered bar chart**.

[](https://microsoftlearning.github.io/mslearn-fabric/Instructions/Labs/Images/kql-report.png)

1. In the **Power BI** window, in the **File** menu, select **Save**. Then save the report as **Revenue by Item.pbix** in the workspace where your lakehouse and KQL database are defined using a **Non-Business** sensitivity label.
2. Close the **Power BI** window, and in the bar on the left, select the icon for your workspace.

Refresh the Workspace page if necessary to view all of the items it contains.

1. In the list of items in your workspace, note that the **Revenue by Item** report is listed.

Clean up resources

In this exercise, you have created a lakehouse, a KQL database to analyze the data uploaded into the lakehouse. You used KQL to query the data and create a query set, which was then used to create a Power BI report.

If you’ve finished exploring your KQL database, you can delete the workspace you created for this exercise.

1. In the bar on the left, select the icon for your workspace.
2. In the **…** menu on the toolbar, select **Workspace settings**.
3. In the **General** section, select **Remove this workspace**.