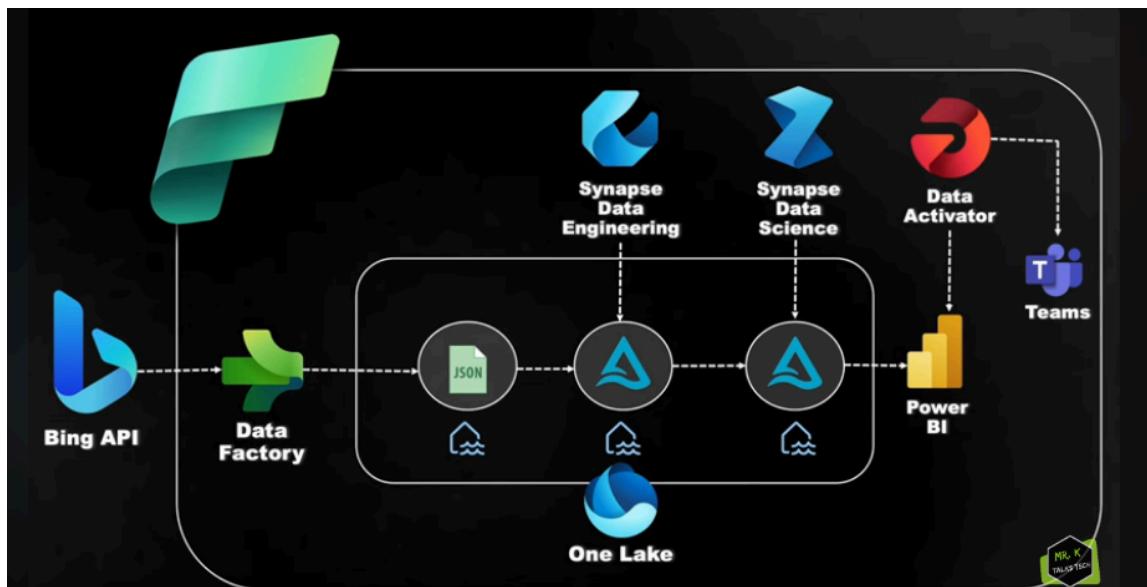


## Project Overview / Architecture

1. We'll be using Bing search data to build BI dashboards using Fabric
2. Our data source will be Bing API
3. Here in this proj..we inject data from Bing API via Synapse Data factory and store it in One lake
4. Initially the data would be in Json format and we clean the data using synapse data engineering and store them in lake DB in the delta table format
5. We use this clean data and perform NLP sentiment analysis using Synapse Data Science and store them in Lake DB
6. This data will be the final data to make PowerBI Reports...and we use data activator for analytic purpose ..

In Azure Fabric, the **Data Activator** is a feature used to trigger workflows, automation, and event-driven actions based on data changes or specific conditions in a data store. It allows you to monitor data streams and react to changes, essentially acting as a bridge between your data and the applications that need to react to it.

7. Overall Architecture





# Agenda

8.

- Environment Setup

- Data Ingestion

- Data Transformation  
(Incremental Load)

- Sentiment Analysis  
(Incremental Load)

- Data Reporting

- Building Pipelines

- Setting up alerts  
(Data Activator)

- End to End Testing

## Environment Setup

1. We create a Resource group and after that we create a data resource for Bing API

2. In marketplace search for bing and select bing v7

The screenshot shows the 'Create a Bing search resource' page in the Microsoft Azure Marketplace. The 'Basics' tab is selected. The 'Subscription' dropdown is set to 'rg-bing-data-analytics'. The 'Resource group' dropdown is also set to 'rg-bing-data-analytics'. The 'Name' field contains 'bing-news-api'. The 'Region' field is set to 'Global'. A note indicates that this is a global resource. The 'Pricing tier' dropdown is set to 'F1 (3 Calls per second, 1k Calls per month)'. At the bottom, there are 'Previous', 'Next', and 'Review + create' buttons.

### Bing Autosuggest

Bing Autosuggest API helps narrow the search quickly by allowing your users to see suggestions for popular search terms. It can correct perceived mistakes and returns detailed contextual suggestions according to other searches people have found useful.

## Bing Search API v7 Pricing Details

INSTANCE	TRANSACTIONS PER SECOND (TPS)	FEATURES	ALL MARKETS
Free	3 TPS	Bing Image Search Bing News Search Bing Video Search Bing Visual Search Bing Web Search Bing Entity Search Bing Autosuggest Bing Spell Check	1,000 transactions free per month for all markets
S1	250 TPS	Bing Web Search Bing Image Search Bing News Search Bing Video Search Bing Entity Search Bing Autosuggest* Bing Spell Check*	\$25 per 1,000 transactions \$25 per 25,000 transactions*

3. Here we have created the bing search API
4. We can connect to this API via endpoints and ingest the news data
5. And we have successfully created a resource and bing search API

## 6. Our current fabric workspace

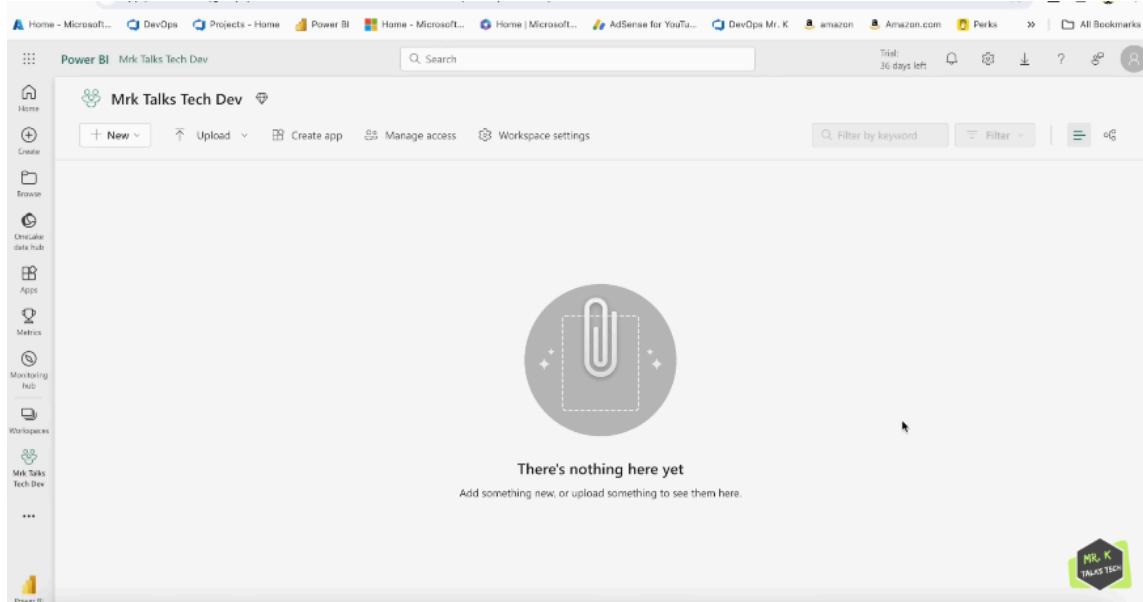
The screenshot shows the Power BI Home page. On the left, there's a sidebar with various icons for Home, Create, Browse, Data Factory, Data Activator, Industry Solutions, Synapse, Data Engineering, Data Science, Data Warehouse, and Real-Time Analytics. The 'Power BI' icon is highlighted. In the main area, there's a 'Recommended' section with cards for 'You frequently open this' (My workspace), 'Getting started with Power BI' (Explore basic Power BI concepts), 'Explore this data story' (Explore the 100 most useful productivity tips), 'Explore this data story' (Cancer statistics in the USA), and 'Getting st...' (Intro—What is PC...). Below this is a table showing recent workspaces:

Name	Type	Opened	Location	Endorsement	Sensitivity
My workspace	Workspace	33 minutes ago	Workspaces	—	—
My workspace	Workspace	33 minutes ago	Workspaces	—	—

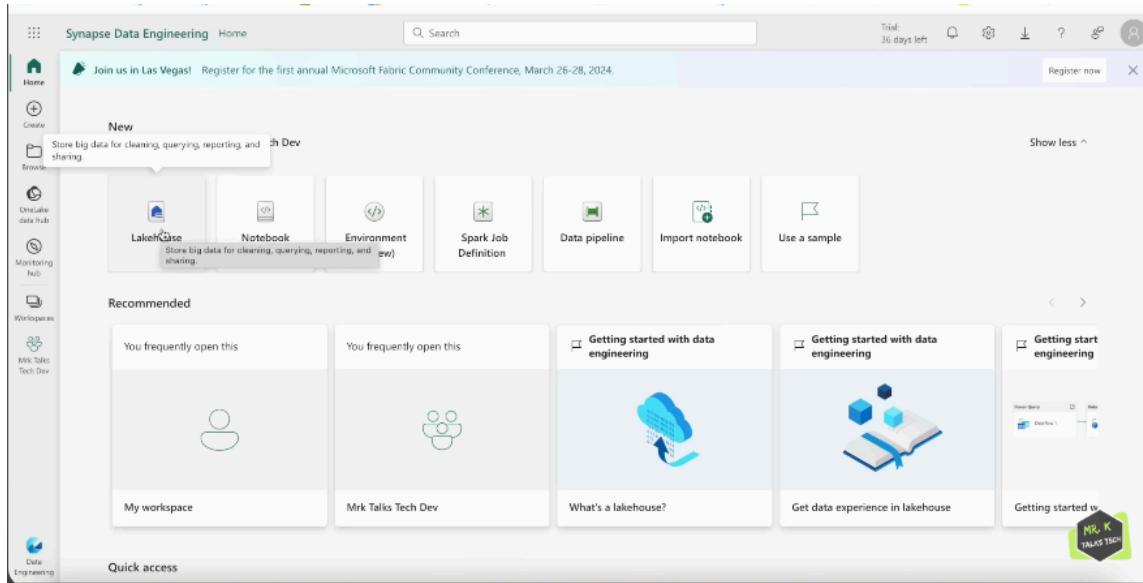
## 7. Inside the fabric we need to create a new workspace for our project and enable fabric access

The screenshot shows the Power BI Home page with a 'Create a workspace' overlay on the right. The sidebar and recommended cards are the same as in the previous screenshot. The overlay has fields for 'Workspace image' (Upload or Reset), 'Contact list' (kumarsr (Owner) and a text input for 'Enter users and groups'), and 'License mode'. There are three options: 'Trial' (selected), 'Pro' (radio button), and 'Premium per-user' (radio button). The 'Trial' option includes a note about a 60-day trial for new features. The 'Pro' option includes a note about Pro licenses. The 'Premium per-user' option includes a note about Premium per-user licenses. At the bottom are 'Apply' and 'Cancel' buttons.

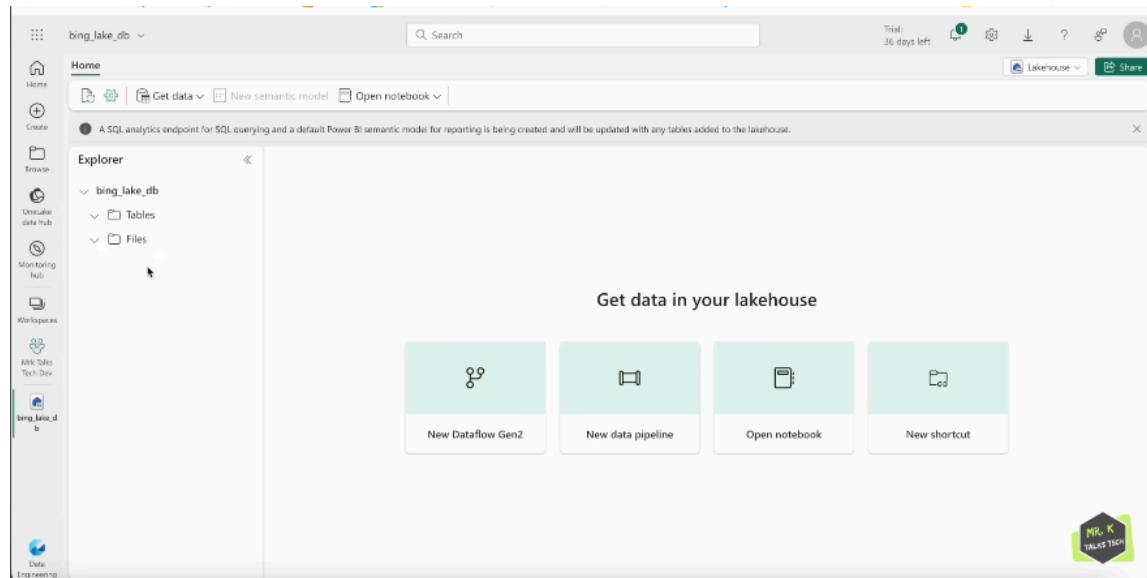
8. This is the workspace that we will be using for fabric project we can get access to fabric features



9. Go to Synapse data engineering comp and create a lakehouse



## 10. After creating the lakehouse



## Data Ingestion

1. Here we will be using ADF for ingesting our data
2. Before that we to get the data we use the endpoints from our API

A screenshot of the Microsoft Azure portal. The URL is 'rg-bing-data-analytics > bing-news-api'. The left sidebar has sections for Overview, Activity log, Access control (IAM), Tags, Diagnose and solve problems, RESOURCE MANAGEMENT (with 'Keys and Endpoint' selected), Pricing tier, Billing By Subscription, Properties, Locks, and Monitoring. The main content area shows the 'bing-news-api | Keys and Endpoint' blade. It includes a note about keys, two text input fields for 'Key 1' and 'Key 2' (both redacted), an 'Endpoint' field with the value 'https://api.bing.microsoft.com/', and a 'Location' field set to 'global'. A watermark for 'MR K TALKS TECH' is in the bottom right.

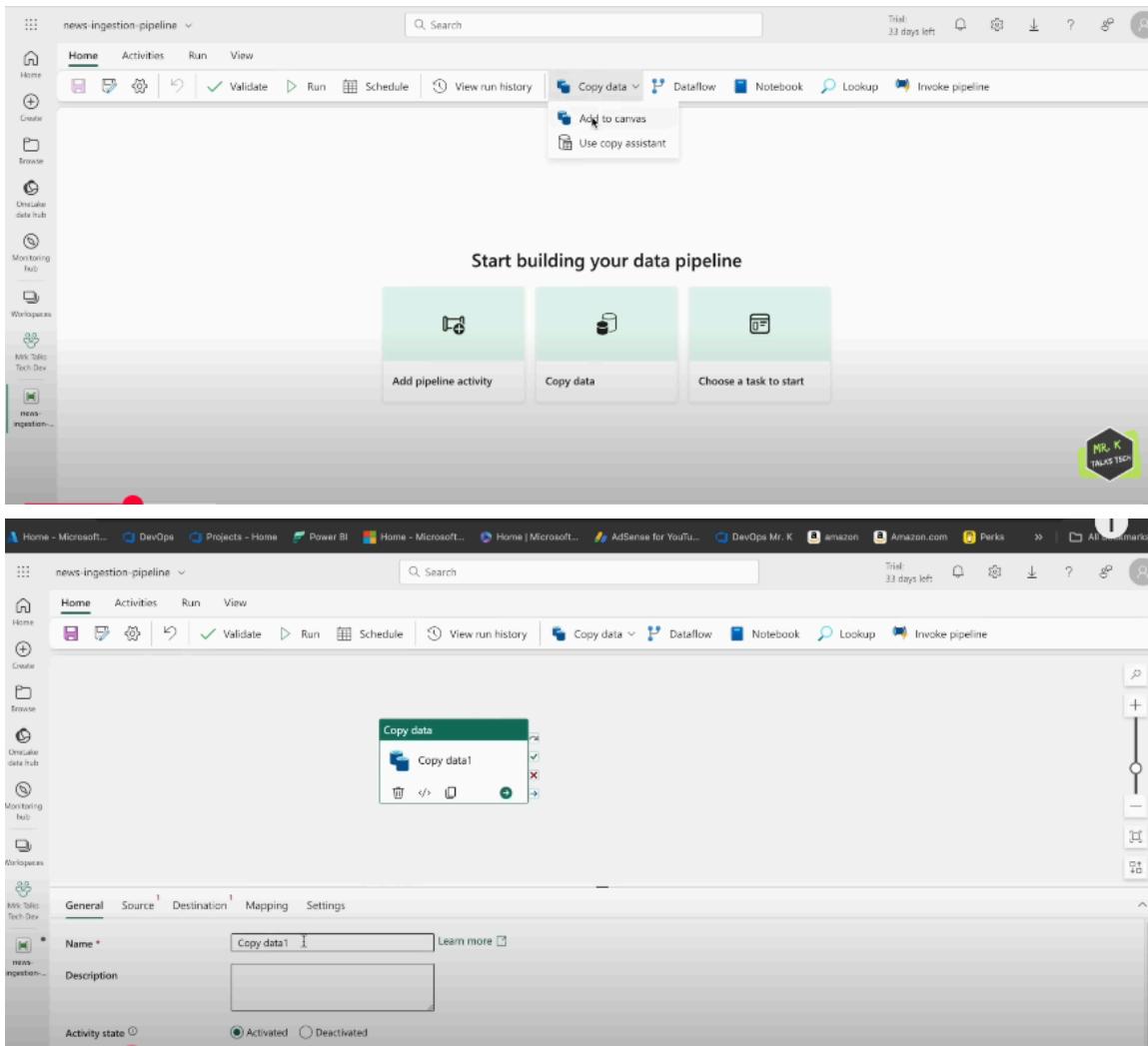
3. Go to the lakehouse DB and from bottom left menu choose data factory



4. Click on data pipeline and create it a name news\_ingestion

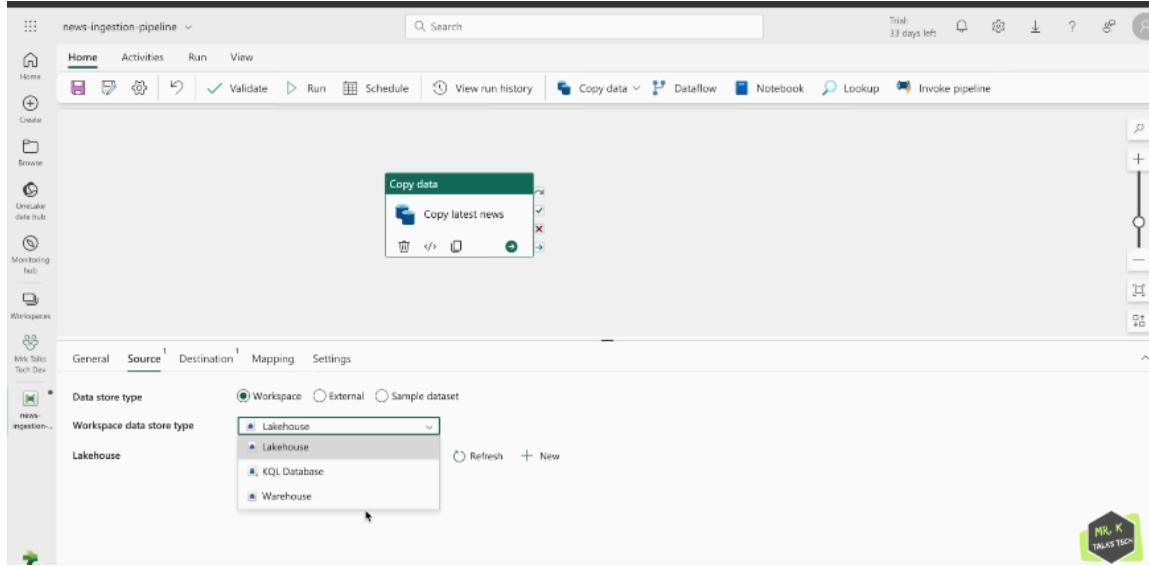
The image consists of two vertically stacked screenshots of the Microsoft Data Factory interface. The top screenshot shows the 'New' workspace creation screen. It has a sidebar with 'Data Factory' selected. The main area shows a 'Current workspace: Mrk Talks Tech Dev' message and two options: 'Dataflow Gen2' and 'Data pipeline'. Below this is a 'Recommended' section with five items: 'My workspace', 'Mrk Talks Tech Dev', 'Learn to use Data Factory', 'Create your first dataflow', and 'Create your first pipeline'. The bottom screenshot shows the 'news-ingestion-pipeline' workspace home screen. It has a similar sidebar and navigation bar. The main area features a central message 'Start building your data pipeline' above three cards: 'Add pipeline activity', 'Copy data', and 'Choose a task to start'. A green watermark with 'MR. K TALKS TECH' is in the bottom right corner of both screenshots.

5. Now click on the copy data activity

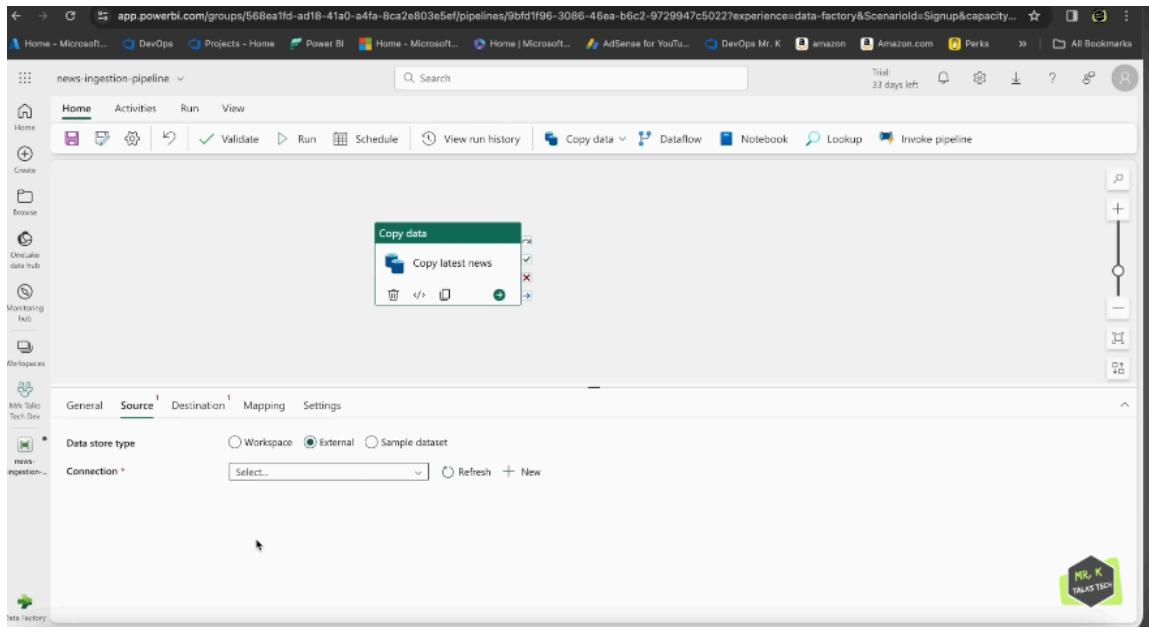


6. Give a name to our copy activity and provide the source

7. Here if the source is “fabric lakehouse or KQL or warehouse “ then we can choose workspace

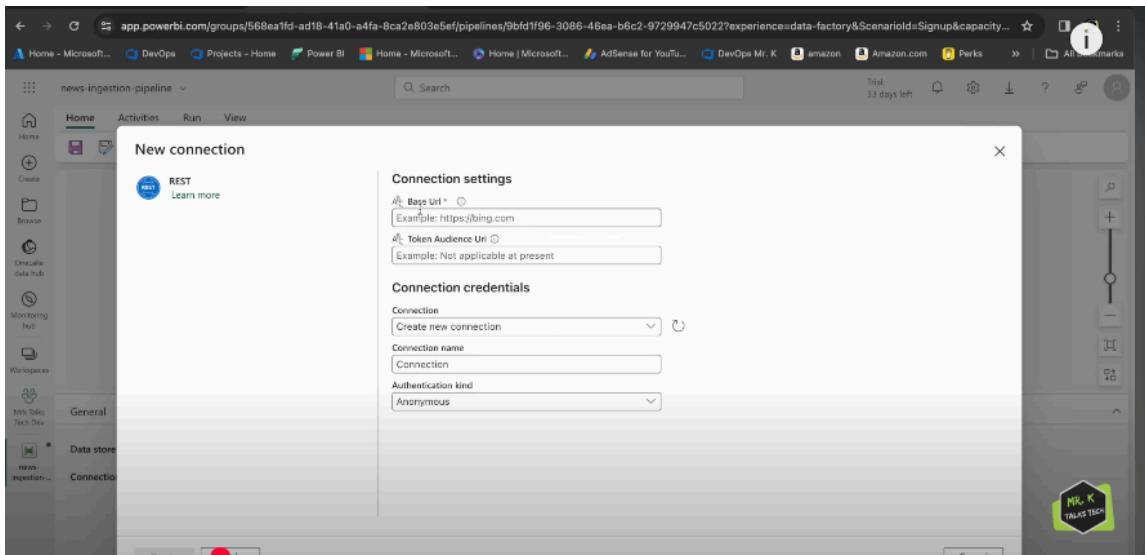


8. Since our data is coming via Azure API we need to choose as external source



Next we need to configure the connection

## 9. Create a new connection and select REST



now we need to give url of API

## 10. To get the URL link...go to our bing API and in the overview→Tutorial go to news seach API

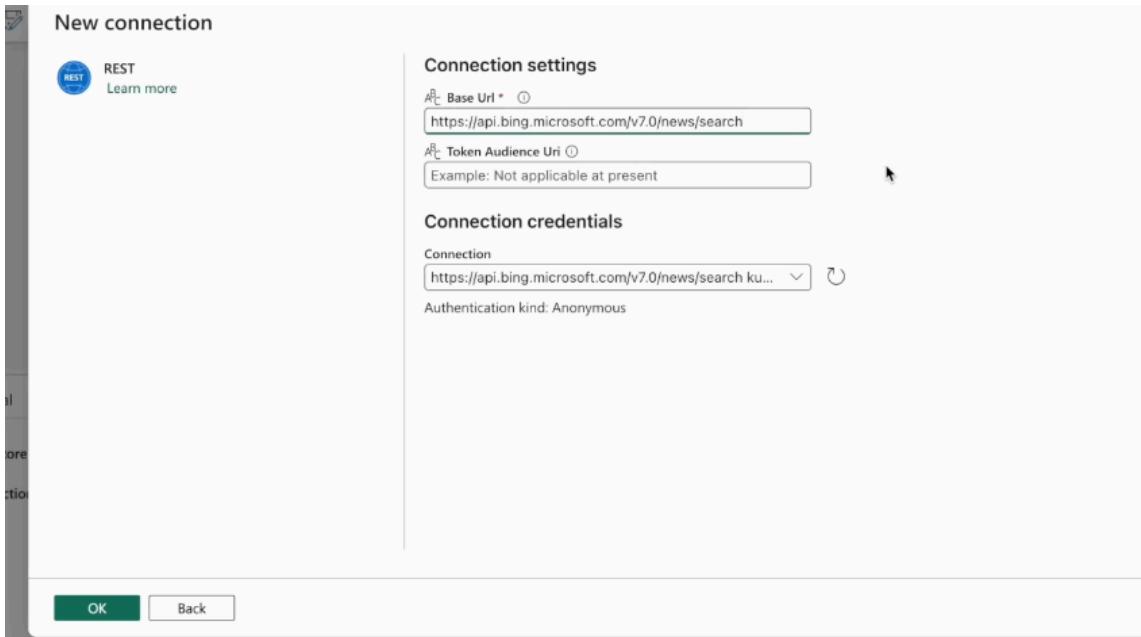
The screenshot shows the Microsoft Learn Bing News Search documentation page. The left sidebar has a 'Query parameters' section highlighted. The main content area is titled 'News Search APIs v7 query parameters' and discusses required query parameters like 'cc'. A table details these parameters.

Name	Value	Type	Required
cc	A 2-character country code of the country where the results come from. For a list of possible values, see <a href="#">Market codes</a> .	String	No

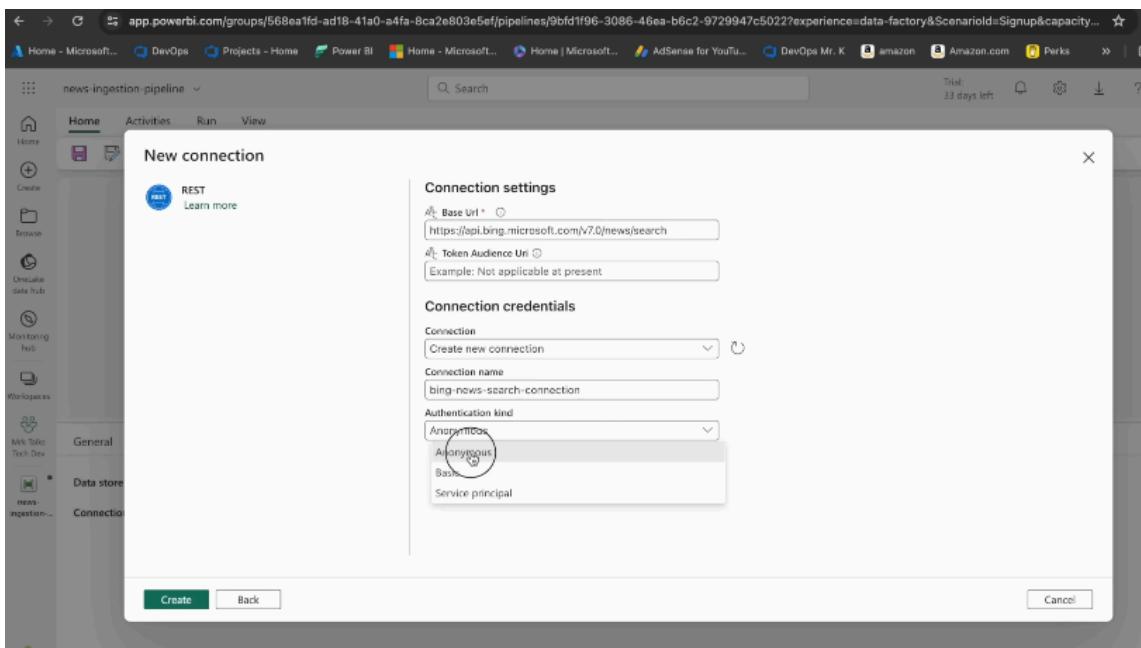
The screenshot shows the Microsoft Learn Bing News Search documentation page. The left sidebar has an 'Endpoints' section highlighted. The main content area is titled 'Endpoints' and lists three API endpoints: 'https://api.bing.microsoft.com/v7.0/news', 'https://api.bing.microsoft.com/v7.0/news/search', and 'https://api.bing.microsoft.com/v7.0/news/trendingtopics'.

Endpoint	Description
<a href="https://api.bing.microsoft.com/v7.0/news">https://api.bing.microsoft.com/v7.0/news</a>	Returns the top news articles by category. For example, you can request the top sports or entertainment articles. For information about specifying categories, see the <a href="#">category query parameter</a> .
<a href="https://api.bing.microsoft.com/v7.0/news/search">https://api.bing.microsoft.com/v7.0/news/search</a>	Returns news articles based on the user's search query. If the search query is empty, the call returns the top news articles.
<a href="https://api.bing.microsoft.com/v7.0/news/trendingtopics">https://api.bing.microsoft.com/v7.0/news/trendingtopics</a>	Returns news topics that are currently trending on social networks. For a list of markets that support trending news, see <a href="#">Supported Trending News markets</a> .

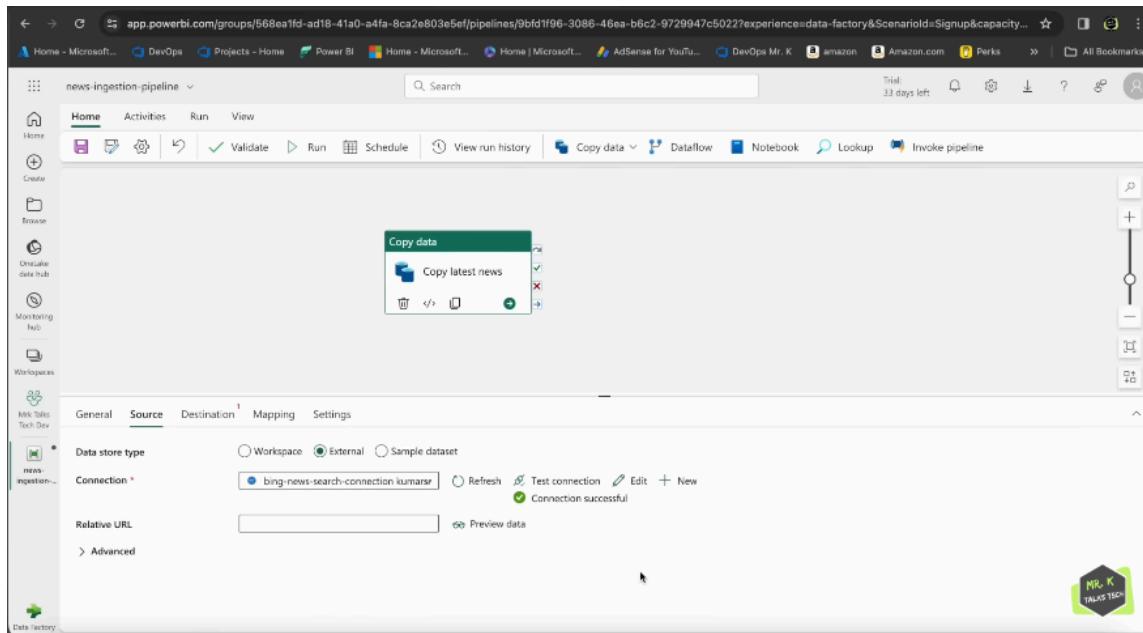
Copy the end point which is our base URL



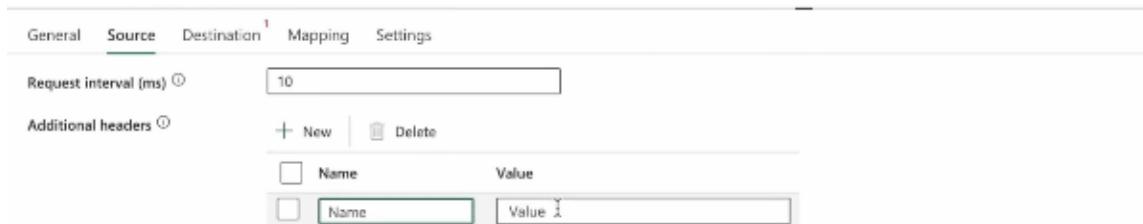
next we have to provide the connection details of this connection



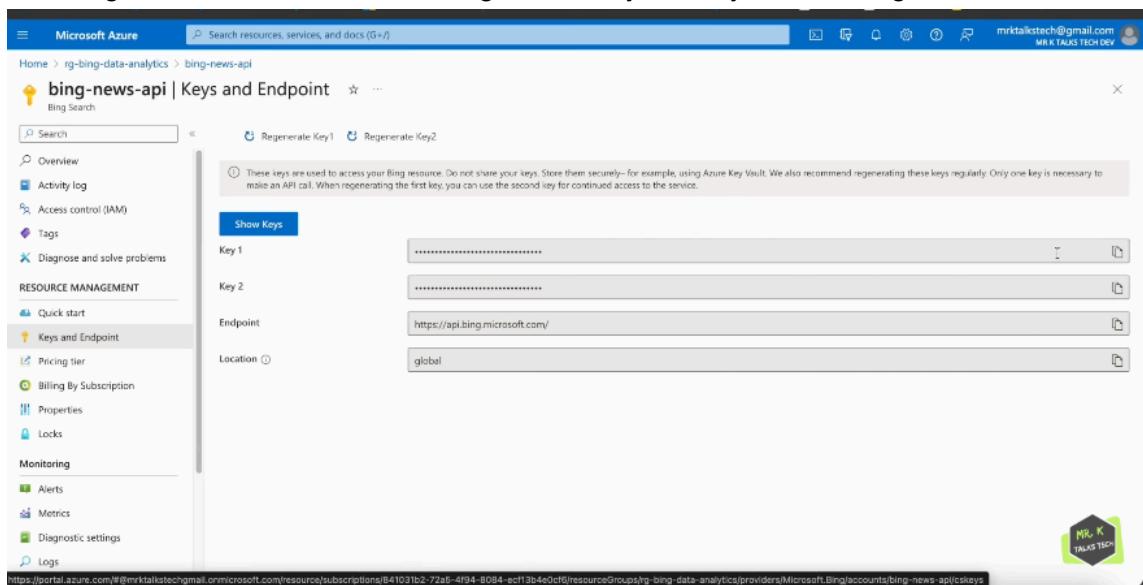
and click on create



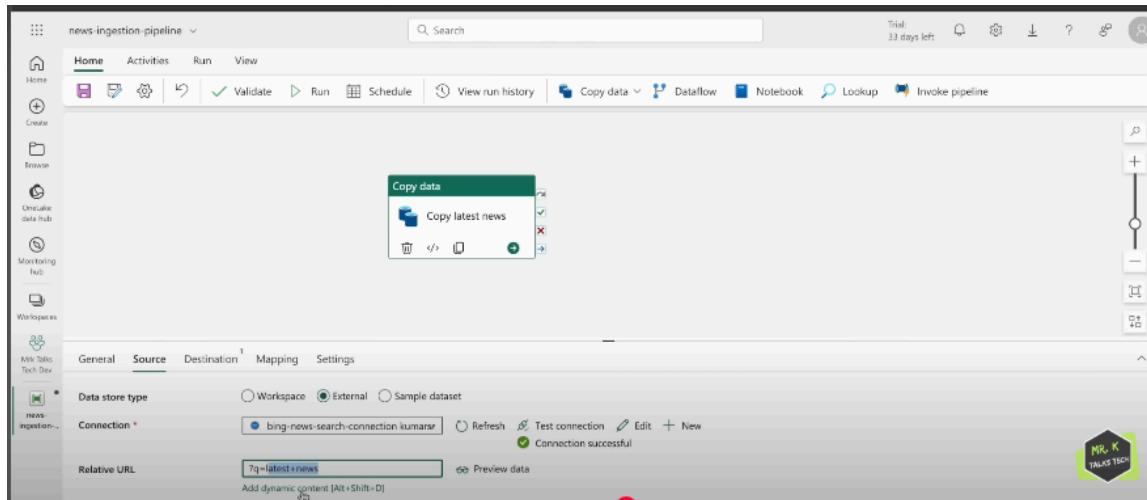
11. Next we need to give access to the bing Api data for Fabric
12. We will authenticate this API using headers...which contains the key of the API
13. For that in the source tab→Advanced...click on additional headers



we can give a name and we have to give this key1 or key2 of our bing API as the value



14. Here we need to only get the latest news from our API... for this ...we need to pass the url keyword in the relative url



here we gave our search as latest news...so this gives the latest news

**Preview data**

```

    "value": [
        {
            "name": "Africa Live: New gas leak feared days after deadly Kenyan blast",
            "url": "https://www.bbc.com/news/live/world-africa-67977518?pinne...
            "description": "Authorities evacuate the same part of Nairobi where an earlier explosi...
            "about": [
                {
                    "readLink": "https://api.bing.microsoft.com/api/v7/entities/d8c12054-a38a-14f5-8f2...
                    "name": "BBC News"
                },
                {
                    "readLink": "https://api.bing.microsoft.com/api/v7/entities/5e22fc18-eb26-41e5-8fa...
                    "name": "Africa"
                },
                {
                    "readLink": "https://api.bing.microsoft.com/api/v7/entities/8ee43333-b344-289c-d8b...
                    "name": "Kenya"
                }
            ],
            "provider": [
                {
                    "_type": "Organization",
                    "name": "BBC",
                    "image": {
                        "thumbnail": {
                            "contentUrl": "https://www.bing.com/th?id=00F.k3N40qazNjgK376rAnFmsA&pid=news"
                        }
                    }
                }
            ]
        }
    ]
}

```

## 15. Here we can see the query parameters for our Bing API

The screenshot shows a web browser displaying the Microsoft Learn Bing API documentation at [learn.microsoft.com/en-us/bing/search-apis/bing-news-search/reference/query-parameters](https://learn.microsoft.com/en-us/bing/search-apis/bing-news-search/reference/query-parameters). The left sidebar is collapsed, and the main content area is titled 'Query parameters'. It contains two sections: 'category' and 'count'. The 'category' section describes how to filter news articles by category like Sports or Entertainment. The 'count' section explains how to specify the number of articles to return, with a note about using it with the offset parameter for paginated results.

and we use count function to get 100 articles

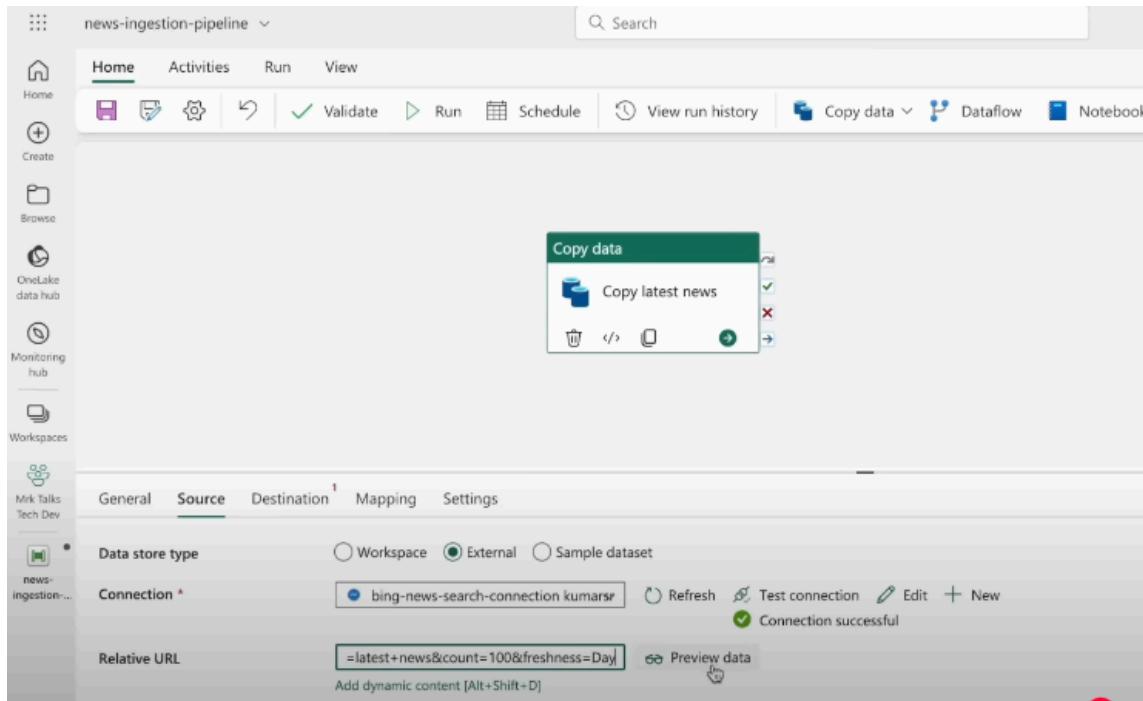
The screenshot shows the Azure Data Factory pipeline editor for a pipeline named 'news-ingestion-pipeline'. On the left, there's a navigation bar with 'Home', 'Activities', 'Run', and 'View'. Below it are buttons for 'Create', 'Browse', 'Omnichannel data hub', and 'Monitoring hub'. Under 'Workspaces', there's a workspace named 'Mr. K Talks Tech Dev'. In the center, the pipeline canvas shows a 'Copy data' activity. The 'Source' tab is selected, showing a connection named 'bing-news-search-connection-kumarsh' (selected via a radio button) and a relative URL of '?q=latest+news&count=100'. The 'Destination' tab is also visible. On the right side of the canvas, there are various pipeline components like 'Dataflow', 'Notebook', 'Lookup', and 'Invoke pipeline'.

so this is how we use the query parameters

## 16. Next we will freshness parameter

<div style="background-color: #f0f0f0; padding: 10px;"> <p><input type="text"/> Filter by title</p> <p>Bing News Search documentation</p> <ul style="list-style-type: none"> <li>&gt; Overview</li> <li>&gt; Quickstarts</li> <li>&gt; Tutorials</li> <li>Samples</li> <li>&gt; How-to guides</li> <li>&gt; Reference</li> <li>&gt; REST</li> <li>Endpoints</li> <li>Response objects</li> <li><b>Query parameters</b></li> <li>Headers</li> <li>Market and language codes</li> <li>Error codes</li> </ul> <p>&gt; Resources</p> <p>Use and display requirements</p> <p>Use and display requirements for your LLM</p> <p>Release notes</p>   <p><input type="button"/> Download PDF</p> </div>	<p>published in the last 24 hours from any category.</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%;">count</td><td>The number of news articles to return in the response. The actual number delivered may be less than requested. The default is 10 and the maximum is 100.</td><td style="width: 10%;">UnsignedShort</td><td style="width: 10%;">No</td></tr> <tr> <td colspan="4"> <p>You may use this parameter along with the <code>offset</code> parameter to page results. For example, if your user interface presents 20 articles per page, set <code>count</code> to 20 and <code>offset</code> to 0 to get the first page of results. For each subsequent page, increment <code>offset</code> by 20 (for example, 0, 20, 40). It is possible for multiple pages to include some overlap in results.</p> <p>Use this parameter only when calling the <code>/news/search</code> endpoint. Do not specify this parameter when calling the <code>/news</code> or <code>/news/trendingtopics</code> endpoint.</p> </td></tr> <tr> <td style="font-weight: bold;">freshness</td><td>Filter news articles by the following age values:</td><td style="font-weight: bold;">String</td><td style="font-weight: bold;">No</td></tr> <tr> <td></td><td> <ul style="list-style-type: none"> <li>• Day — Return news articles that Bing discovered within the last 24 hours.</li> <li>• Week — Return news articles that Bing discovered within the last 7 days.</li> <li>• Month — Return news articles that Bing discovered within the last 30 days.</li> </ul> </td><td></td><td></td></tr> <tr> <td colspan="4"> <p>Use this parameter only when calling the <code>/news/search</code> endpoint. Do not specify this parameter when calling the <code>/news</code> or <code>/news/trendingtopics</code> endpoint.</p> </td></tr> <tr> <td style="font-weight: bold;">mkt</td><td>The market where the results come from. Typically, <code>mkt</code> is the country where the user is making the request from. However, it could be a different country if the user is not located in a country where Bing delivers results. The market must be in the form <code>&lt;language&gt;-&lt;country/region&gt;</code>. For example, <code>en-US</code>. The string is case insensitive. For a list of possible market values, <a href="#">see Market codes</a>.</td><td style="font-weight: bold;">String</td><td style="font-weight: bold;">No</td></tr> </table>	count	The number of news articles to return in the response. The actual number delivered may be less than requested. The default is 10 and the maximum is 100.	UnsignedShort	No	<p>You may use this parameter along with the <code>offset</code> parameter to page results. For example, if your user interface presents 20 articles per page, set <code>count</code> to 20 and <code>offset</code> to 0 to get the first page of results. For each subsequent page, increment <code>offset</code> by 20 (for example, 0, 20, 40). It is possible for multiple pages to include some overlap in results.</p> <p>Use this parameter only when calling the <code>/news/search</code> endpoint. Do not specify this parameter when calling the <code>/news</code> or <code>/news/trendingtopics</code> endpoint.</p>				freshness	Filter news articles by the following age values:	String	No		<ul style="list-style-type: none"> <li>• Day — Return news articles that Bing discovered within the last 24 hours.</li> <li>• Week — Return news articles that Bing discovered within the last 7 days.</li> <li>• Month — Return news articles that Bing discovered within the last 30 days.</li> </ul>			<p>Use this parameter only when calling the <code>/news/search</code> endpoint. Do not specify this parameter when calling the <code>/news</code> or <code>/news/trendingtopics</code> endpoint.</p>				mkt	The market where the results come from. Typically, <code>mkt</code> is the country where the user is making the request from. However, it could be a different country if the user is not located in a country where Bing delivers results. The market must be in the form <code>&lt;language&gt;-&lt;country/region&gt;</code> . For example, <code>en-US</code> . The string is case insensitive. For a list of possible market values, <a href="#">see Market codes</a> .	String	No
count	The number of news articles to return in the response. The actual number delivered may be less than requested. The default is 10 and the maximum is 100.	UnsignedShort	No																						
<p>You may use this parameter along with the <code>offset</code> parameter to page results. For example, if your user interface presents 20 articles per page, set <code>count</code> to 20 and <code>offset</code> to 0 to get the first page of results. For each subsequent page, increment <code>offset</code> by 20 (for example, 0, 20, 40). It is possible for multiple pages to include some overlap in results.</p> <p>Use this parameter only when calling the <code>/news/search</code> endpoint. Do not specify this parameter when calling the <code>/news</code> or <code>/news/trendingtopics</code> endpoint.</p>																									
freshness	Filter news articles by the following age values:	String	No																						
	<ul style="list-style-type: none"> <li>• Day — Return news articles that Bing discovered within the last 24 hours.</li> <li>• Week — Return news articles that Bing discovered within the last 7 days.</li> <li>• Month — Return news articles that Bing discovered within the last 30 days.</li> </ul>																								
<p>Use this parameter only when calling the <code>/news/search</code> endpoint. Do not specify this parameter when calling the <code>/news</code> or <code>/news/trendingtopics</code> endpoint.</p>																									
mkt	The market where the results come from. Typically, <code>mkt</code> is the country where the user is making the request from. However, it could be a different country if the user is not located in a country where Bing delivers results. The market must be in the form <code>&lt;language&gt;-&lt;country/region&gt;</code> . For example, <code>en-US</code> . The string is case insensitive. For a list of possible market values, <a href="#">see Market codes</a> .	String	No																						

we will use day to get latest news



The screenshot shows the Azure Data Factory pipeline editor. On the left, there's a sidebar with navigation links like Home, Activities, Run, View, Create, Browse, OneLake data hub, Monitoring hub, Workspaces, and a workspace named 'Mrk Talks Tech Dev'. The main area has tabs for General, Source, Destination, Mapping, and Settings. Under the Source tab, the 'Data store type' is set to 'External' (bing-news-search-connection kumarsr), and the 'Connection' dropdown also shows 'bing-news-search-connection kumarsr'. The 'Relative URL' field contains the value '=latest+news&count=100&freshness=Day'. A 'Copy data' dialog box is open over the source settings, showing a green checkmark next to the 'Copy latest news' option. Other buttons in the dialog include a trash can, a refresh icon, a test connection button (which says 'Connection successful'), and a preview data button.

## 17. Next we have mkt parameter

<p><input type="checkbox"/> Filter by title</p> <p>Bing News Search documentation</p> <ul style="list-style-type: none"> <li>&gt; Overview</li> <li>&gt; Quickstarts</li> <li>&gt; Tutorials</li> <li>Samples</li> <li>&gt; How-to guides</li> <li>✓ Reference           <ul style="list-style-type: none"> <li>✗ REST               <ul style="list-style-type: none"> <li>Endpoints</li> <li>Response objects</li> <li>Query parameters</li> </ul> </li> <li>Headers</li> <li>Market and language codes</li> <li>Error codes</li> </ul> </li> <li>&gt; Resources</li> <li>Use and display requirements</li> <li>Use and display requirements for your LLM</li> <li>Release notes</li> </ul> <hr/> <p><input type="checkbox"/> Download PDF</p>	<ul style="list-style-type: none"> <li>• Day — Return news articles that Bing discovered within the last 24 hours.</li> <li>• Week — Return news articles that Bing discovered within the last 7 days.</li> <li>• Month — Return news articles that Bing discovered within the last 30 days.</li> </ul> <p>Use this parameter only when calling the <code>/news/search</code> endpoint. Do not specify this parameter when calling the <code>/news</code> or <code>/news/trendingtopics</code> endpoint.</p>
	<p><b>mkt</b></p> <p>The market where the results come from. Typically, <code>mkt</code> is the country where the user is making the request from. However, it could be a different country if the user is not located in a country where Bing delivers results. The market must be in the form <code>&lt;language&gt;-&lt;country/region&gt;</code>. For example, <code>en-US</code>. The string is case insensitive. For a list of possible market values, see Market codes.</p> <p><b>NOTE:</b> If known, you are encouraged to always specify the market. Specifying the market helps Bing route the request and return an appropriate and optimal response. If you specify a market that is not listed in Market codes, Bing uses a best fit market code based on an internal mapping that is subject to change.</p> <p>To know which market Bing used, get the <code>BingAPIs-Market</code> header in the response.</p> <p>This parameter and the <code>cc</code> query parameter are mutually exclusive — do not specify both.</p>
	<p><b>offset</b></p> <p>The zero-based offset that indicates the number of news articles to skip before returning results. The default is 0. The offset should be less than <code>(totalEstimatedMatches - count)</code>.</p> <p>Unsigned Short No</p>

## Market codes

Country/Region			Language	Market code
Argentina			Spanish	es-AR
Australia			English	en-AU
Austria			German	de-AT
Belgium			Dutch	nl-BE
Belgium			French	fr-BE
Brazil			Portuguese	pt-BR
Canada			English	en-CA
Canada			French	fr-CA
Chile			Spanish	es-CL
Denmark			Danish	da-DK
Finland			Finnish	fi-FI
France			French	fr-FR
Germany			German	de-DE
Hong Kong SAR			Traditional Chinese	zh-HK
India			English	en-IN

## Lets use this param

The screenshot shows the Azure Data Factory interface. On the left, there's a sidebar with options like Home, Create, Browse, OneLake data hub, Monitoring hub, and Workspaces. The main area shows a pipeline named 'news-ingestion-pipeline'. A 'Copy data' step is selected, with a sub-menu open showing the step name 'Copy latest news'. Below this, the 'Source' tab is active, showing settings for a 'Data store type' (External), a connection named 'bing-news-search-connection kumars' (status: Connection successful), and a 'Relative URL' input field containing 's&count=100&freshness=Day&mkt=en-IN'. There's also a 'Preview data' button.

## 18. To see more parameters you see here

The screenshot shows the Bing News Search documentation page. The left sidebar has sections like Overview, Quickstarts, Tutorials, Samples, How-to guides, Reference, REST (Endpoints, Response objects, Query parameters), Headers, Market and language codes, Error codes, Resources, Use and display requirements, and Release notes. The 'Query parameters' section is currently selected. The main content area describes the 'safeSearch' parameter, which is used to filter news articles for adult content. It lists three possible values: Off (Return news articles with adult text, images, or videos), Moderate (Return news articles with adult text but not adult images or videos), and Strict (Do not return news articles with adult text, images, or videos). The default is Moderate. A note states that if the request comes from a market that Bing's adult policy requires safeSearch to be set to Strict, Bing ignores the safeSearch value and uses Strict. The 'setLang' parameter is also described, which is used to specify the language for user interface strings. It notes that for a list of supported language codes, see the Bing supported languages page. Bing loads localized strings if setLang contains a valid 2-letter neutral culture code (fr) or a valid 4-letter specific language code.

19. Next we need to config a destination..as our dest is fabric lakehouwe we choose workspace

The screenshot shows the 'news-ingestion-pipeline' in the Azure Data Factory interface. On the left, there's a sidebar with 'Home', 'Activities', 'Run', 'View', 'Create', 'Browse', 'Unscale data hub', 'Monitoring hub', and 'Workspaces'. Under 'Workspaces', there's a 'news-ingestion...' item. The main area has a 'Copy data' activity named 'copy latest news'. The 'Destination' tab is selected. The configuration shows:

- Data store type: Workspace (radio button selected)
- Workspace data store type: Lakehouse (dropdown selected)
- Lakehouse: bing\_lake\_db (dropdown selected)
- Root folder: Files (radio button selected)
- File path: Directory / File name (Directory is selected)
- File format: DelimitedText (dropdown selected)

We have given our bing\_lake\_db as our dest and file format as JSON as our API gives json data

The screenshot shows the same 'news-ingestion-pipeline' interface. The 'Destination' tab is selected for the 'copy latest news' activity. The configuration has been updated:

- Data store type: Workspace (radio button selected)
- Workspace data store type: Lakehouse (dropdown selected)
- Lakehouse: bing\_lake\_db (dropdown selected)
- Root folder: Tables (radio button selected)
- File path: Directory / bing-latest-news.json (Directory is selected)
- File format: JSON (dropdown selected)

20. Next we run our pipeline...then our copy activity copies the data from API and stores in bing\_lake\_db..

## 21. Here we can see ingested file

The screenshot shows the Power BI Home page for the 'bing\_lake\_db' lakehouse. On the left, there's a sidebar with icons for Home, Create, Browse, Data hub, Metrics, Monitoring hub, Workspaces, and Mrk Talks Tech Dev. The main area has a search bar at the top. Below it, there are buttons for 'Get data', 'New semantic model', and 'Open notebook'. A message states: 'A SQL analytics endpoint for SQL querying and a default Power BI semantic model for faster reporting were created and will be updated with any tables added to the lakehouse.' The 'Explorer' section on the left shows 'bing\_lake\_db' expanded, with 'Tables' and 'Files' listed under it. The 'Files' section contains a table with one item:

Name	Date modified	Type	Size
bing-latest-news.json	2/6/2024 1:03:57 A...	JSON	27 KB

## Data Transformation

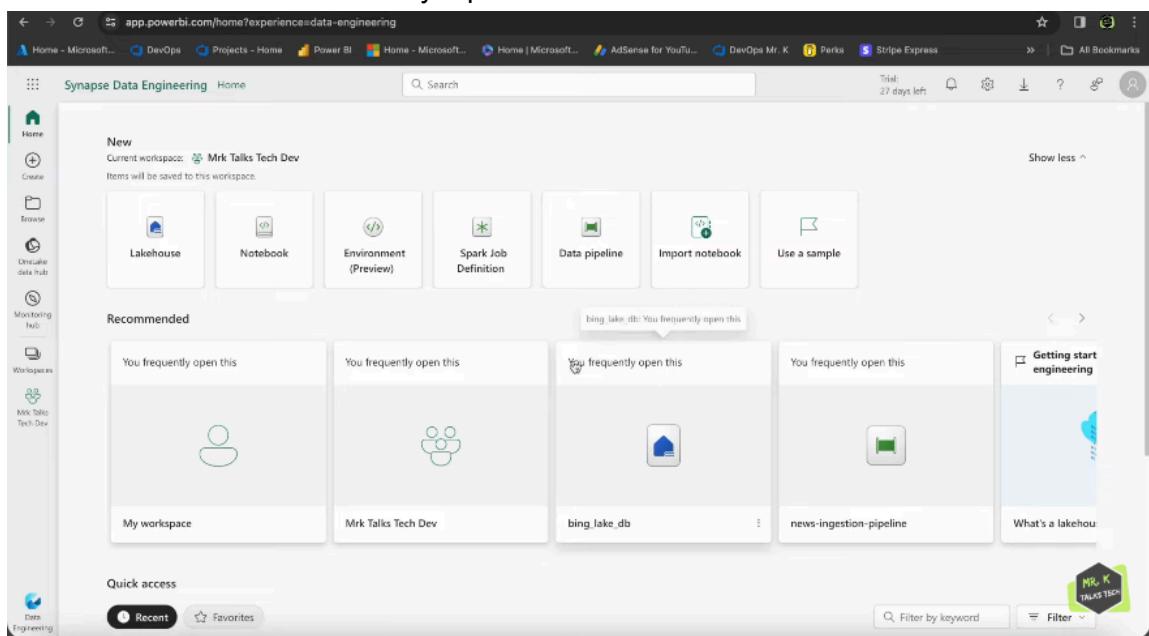
### 1. We use Raw json file that we ingested and clean it

The screenshot shows the Power BI Home page for the 'Mrk Talks Tech Dev' workspace. The sidebar and top navigation are similar to the previous screenshot. The main area displays a table of resources:

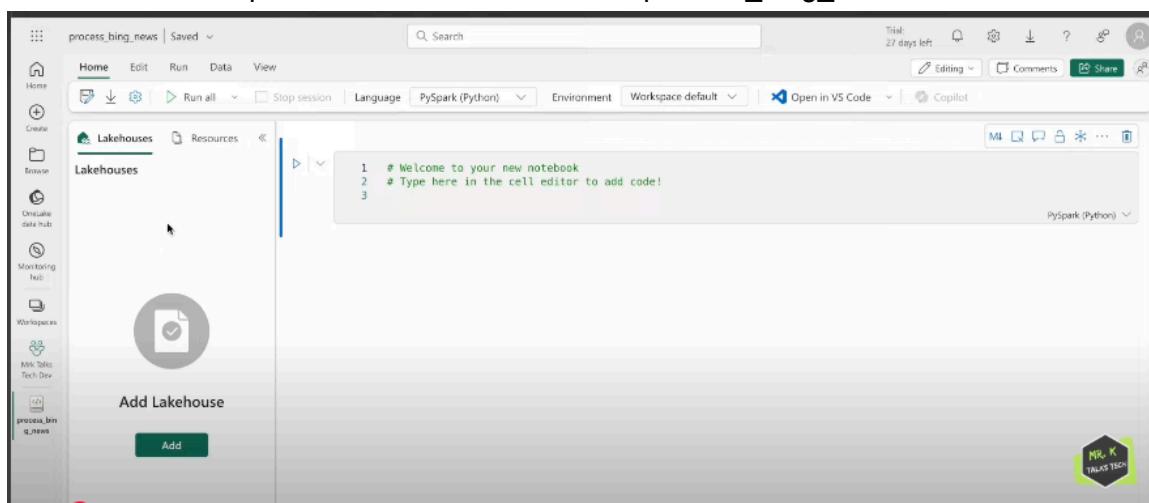
Name	Type	Owner	Refreshed	Next refresh	Endorsement	Sensitivity
bing_lake_db	Lakehouse	kumarsr	—	—	—	—
bing_lake_db	Semantic model (...)	Mrk Talks Tech Dev	2/3/24, 8:13:33 PM	N/A	—	—
bing_lake_db	SQL analytics end...	Mrk Talks Tech Dev	2/11/24, 4:40:15 PM	N/A	—	—
news-ingestion-pipeline	Data pipeline	kumarsr	—	—	—	—

here in our home we have our lake house db and the data pipeline that we have created

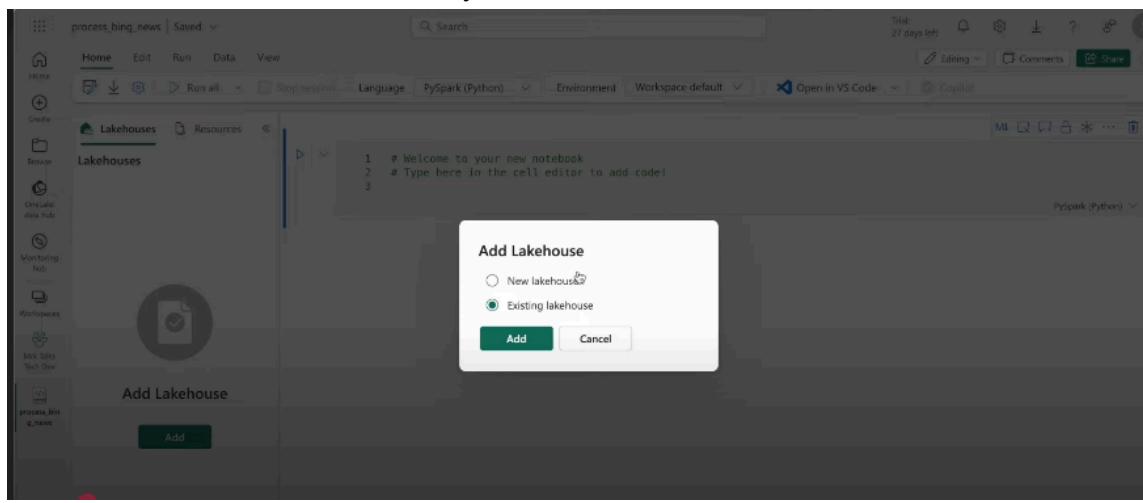
## 2. For this transformation we use synapse DE



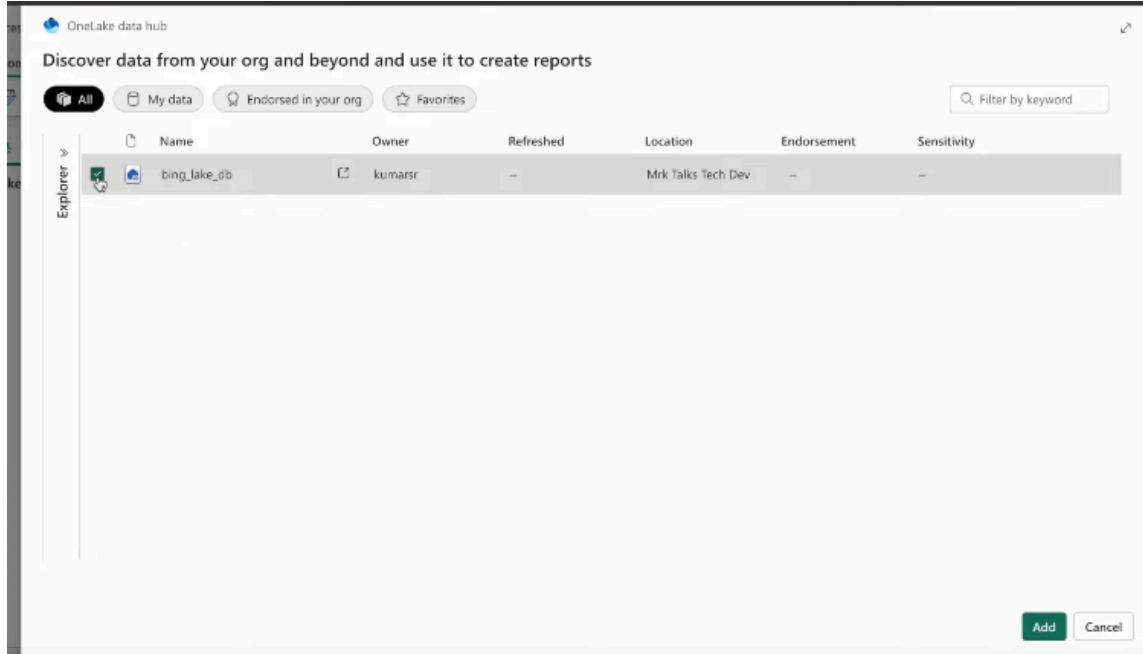
## 3. Here we create a spark notebook and name it as process\_bing\_news



4. Inside the notebook ..we can directly access the lakehouse db

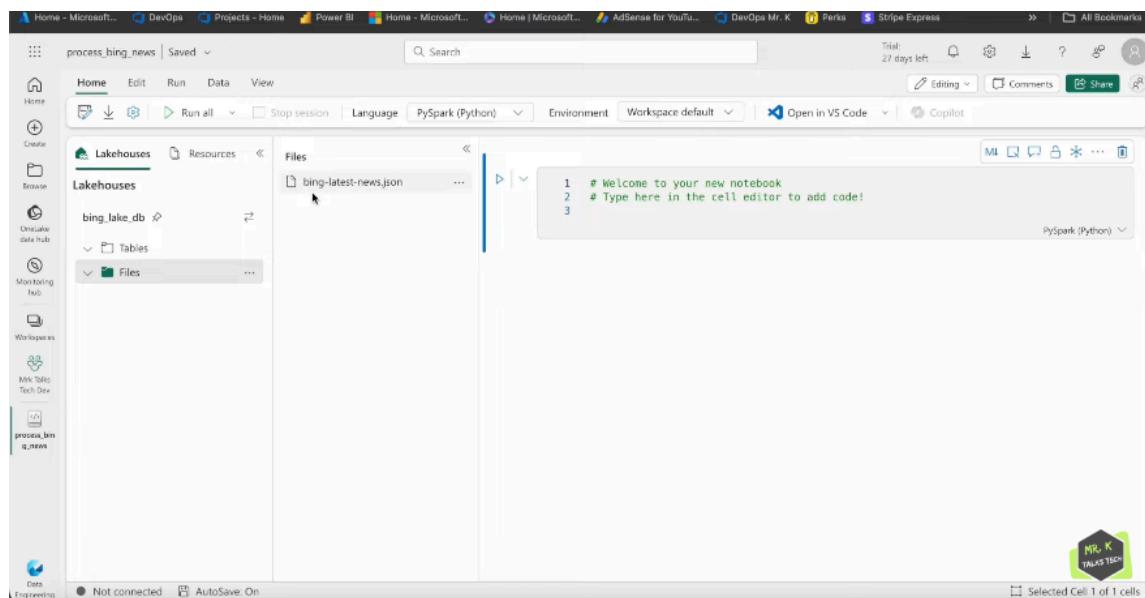


The screenshot shows a Jupyter Notebook interface with a "process\_bing\_news" workspace. A modal dialog box titled "Add Lakehouse" is open in the foreground. It contains two radio button options: "New lakehouse" and "Existing lakehouse", with "Existing lakehouse" selected. Below the radio buttons are "Add" and "Cancel" buttons. In the background, the notebook cell editor shows three lines of code: "# Welcome to your new notebook", "# Type here in the cell editor to add code!", and a third line starting with "3".

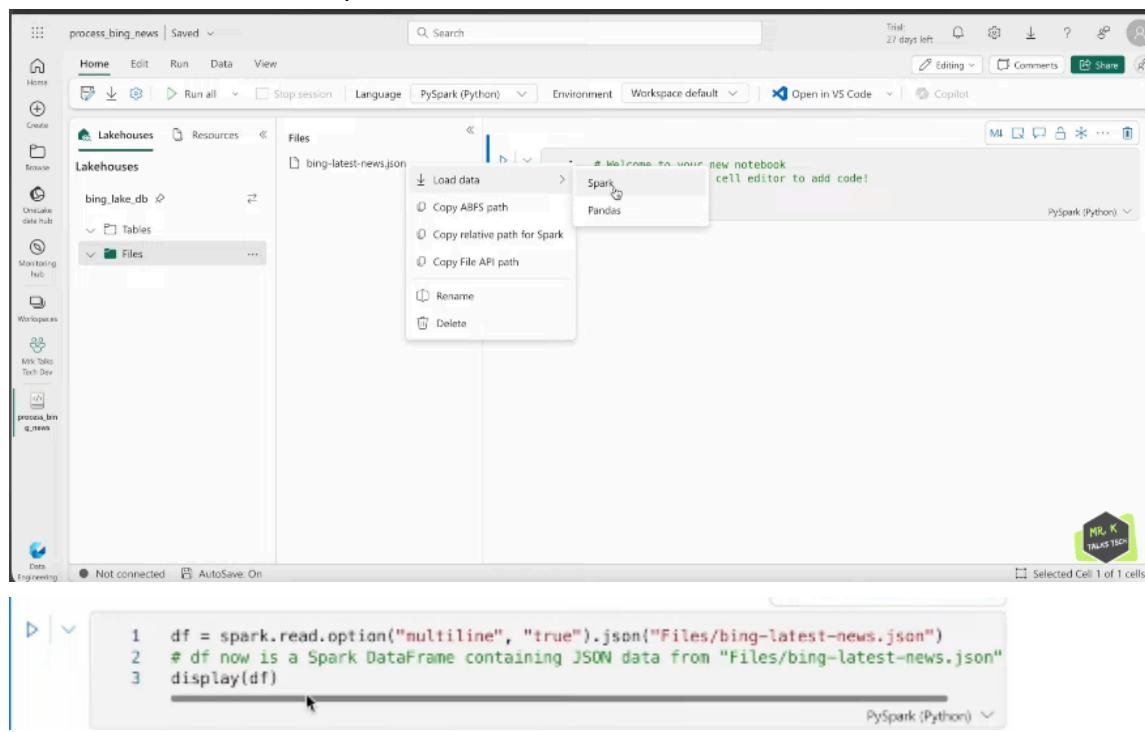
  


The screenshot shows the OneLake data hub interface. At the top, it says "Discover data from your org and beyond and use it to create reports". Below this is a navigation bar with tabs: "All" (selected), "My data", "Endorsed in your org", and "Favorites". There is also a search bar labeled "Filter by keyword". The main area is titled "Explorer" and displays a table of lakehouses. The columns are: Name, Owner, Refreshed, Location, Endorsement, and Sensitivity. One row is visible, showing "bing\_lake\_db" as the Name, "kumarsr" as the Owner, and "Mrk Talks Tech Dev" as the Location.

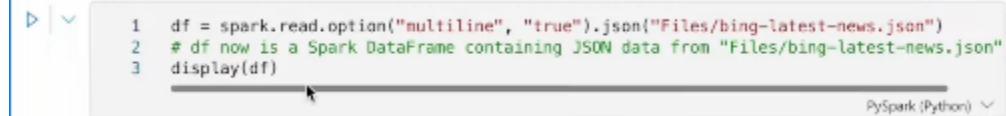
5. If we add our lakehouse to notebook..then we can see our files inside the notebook



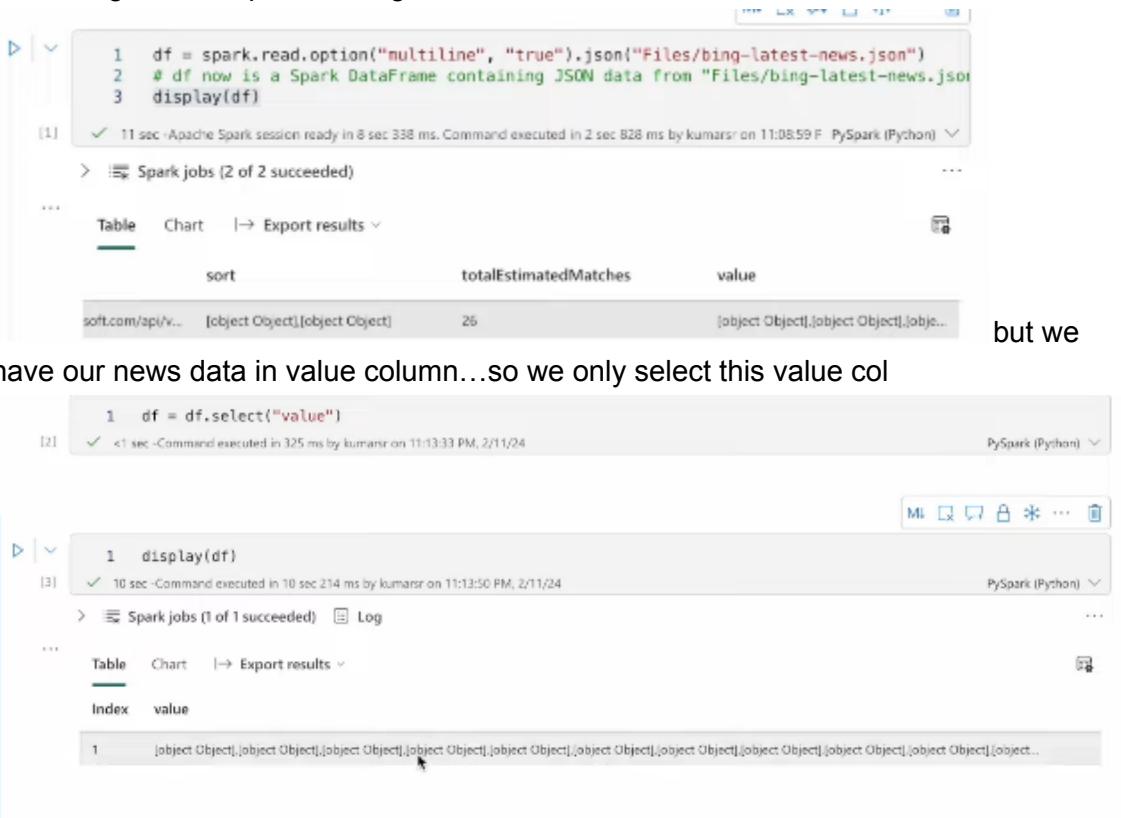
6. To read our data choose spark



7. If our json file has multiple lines then need to provide option in spark.read



## 8. Here we get our output in a single row



The screenshot shows a PySpark session interface. In the top panel, code is run to read a JSON file into a DataFrame:

```
1 df = spark.read.option("multiline", "true").json("Files/bing-latest-news.json")
2 # df now is a Spark DataFrame containing JSON data from "Files/bing-latest-news.json"
3 display(df)
```

The status bar indicates the command was executed in 11 seconds. Below this, the session shows two succeeded Spark jobs.

In the main area, a table view is selected. The DataFrame has three columns: sort, totalEstimatedMatches, and value. The value column contains complex JSON objects.

Text overlay: "but we have our news data in value column...so we only select this value col"

Another code run shows selecting the 'value' column:

```
1 df = df.select("value")
```

The status bar indicates the command was executed in less than a second. The session shows one succeeded job.

Finally, the DataFrame is displayed again, this time with only the 'value' column visible.

## 9. Next we need to process this value col..to see all the rows of values column...we use explode function

In Apache Spark, the `explode` function is used to transform a column containing arrays or maps into multiple rows, where each element in the array or each key-value pair in the map results in a new row. This is especially useful when working with data in a nested format (like JSON) or when dealing with arrays or lists within your dataset, and you want to "flatten" these structures into individual rows for further analysis.

```
python
from pyspark.sql.functions import explode
df_exploded = df.select(explode(df["column_name"]).alias("new_column"))
```

Suppose we have a DataFrame with a column of arrays, and we want to explode that array into separate rows. Here's an example:

#### Input DataFrame:

Let's assume we have the following DataFrame:

id	items
1	["apple", "banana", "cherry"]
2	["orange", "grape"]
3	["kiwi"]

#### Using `explode`:

We can use the `explode` function to turn the `items` column into separate rows:

```
python Copy
from pyspark.sql.functions import explode

df_exploded = df.select("id", explode(df["items"]).alias("item"))
df_exploded.show()
```

#### Output:

id	item
1	apple
1	banana
1	cherry
2	orange
2	grape
3	kiwi

#### In this case:

- For `id = 1`, the `items` array had 3 elements, so 3 new rows were created.
- For `id = 2`, the `items` array had 2 elements, so 2 new rows were created.
- For `id = 3`, the `items` array had 1 element, so 1 new row was created.

## Example: Exploding a Map

If the column contains a map (key-value pairs), `explode` can be used to create multiple rows, each representing a key-value pair.

### Input DataFrame:

Assume the DataFrame looks like this:

id	attributes
1	{"color": "red", "size": "M"}
2	{"color": "blue", "size": "L"}
3	{"color": "green", "size": "S"}

Using `explode`:

```
python Copy
from pyspark.sql.functions import explode
df_exploded = df.select("id", explode(df["attributes"]).alias("attribute", "value"))
df_exploded.show()
```

**Output:**

<b>id</b>	<b>attribute</b>	<b>value</b>
1	color	red
1	size	M
2	color	blue
2	size	L
3	color	green
3	size	S

In this case:

- For `id = 1`, the map had two key-value pairs, so two new rows are created.
- For `id = 2`, the map had two key-value pairs as well, so two new rows are created.
- For `id = 3`, the map had two key-value pairs, so two new rows are created.

**Summary:**

- `explode` is useful for flattening arrays or maps into individual rows.
- It helps in breaking down complex nested data into more manageable, row-based formats, which is particularly useful for analysis and transformations.

10. We use explode for our value col

```
1   from pyspark.sql.functions import explode
2
3   df_exploded = df.select(explode(df["value"]).alias("json_object"))
[5] ✓ <1 sec -Command executed in 315 ms by kumarsr on 11:18:20 PM, 2/11/24
...
[6] ✓ 1 sec -Command executed in 917 ms by kumarsr on 11:19:04 PM, 2/11/24
> ⚡ Spark jobs (1 of 1 succeeded)

Table    Chart    |→ Export results ▾
Index    json_object

1 [object Object]
2 [object Object]
3 [object Object]
4 [object Object]
5 [object Object]
```

here all these objects are news articles

11. Next we use toJSON function to see our objects in json string format

The `to_json` function in Apache Spark is used to convert a column containing complex data types (like structs, arrays, or maps) into a JSON string representation. This is particularly useful when working with structured data (like structs or maps) and you need to serialize it into a JSON format for storage or transfer.

```
python Copy
from pyspark.sql.functions import to_json
df_json = df.select(to_json(df["column_name"]).alias("json_column"))
```

## Example: Converting a Struct to JSON

### Input DataFrame:

Let's assume you have the following DataFrame with a column that contains **structs**:

id	name	address
1	Alice	{"street": "123 Elm St", "city": "NYC"}
2	Bob	{"street": "456 Oak St", "city": "Boston"}

In this case, the **address** column is a **struct** with two fields: `street` and `city`.

### Using `to_json`:

You can use the `to_json` function to convert the **address** column into a JSON string.

```
python Copy
from pyspark.sql.functions import to_json

df_json = df.select("id", "name", to_json(df["address"]).alias("address_json"))
df_json.show(truncate=False)
```

### Output:

id	name	address_json
1	Alice	{"street": "123 Elm St", "city": "NYC"}
2	Bob	{"street": "456 Oak St", "city": "Boston"}

In this example:

- The **address** struct is converted into a JSON string representation: `{"street": "123 Elm St", "city": "NYC"}` for Alice and `{"street": "456 Oak St", "city": "Boston"}` for Bob.

## 12. Our json structure would look like this

```
1 json_list = df_exploded.toJSON().collect()
✓ 4 sec -Command executed in 3 sec 670 ms by kumarsr on 11:28:05 PM, 2/11/24

> ⏷ Spark jobs (1 of 1 succeeded) ⏷ Log

1 print(json_list)
✓ <1 sec -Command executed in 322 ms by kumarsr on 11:28:18 PM, 2/11/24

-----
lawmakers could lead to a new water resource in the eastern part of the state. On Friday, Fe
Senate introduced Senate Bill 497 which seeks to establish the Pike Reservoir Project Distri
consider creation of new lake, dam in Bourbon County","provider":[{"_type":"Organization","i
{"contentUrl":"https://www.bing.com/th?
id=ODF.UM0amhg3WA0bCEGqc4w_Cw&pid=news"}],"name":"Yahoo"]],"url":"https://www.yahoo.com/news
lake-dam-193131924.html"}}, {"json_object":{"about":[{"name":"Super
Bowl","readLink":"https://api.bing.microsoft.com/api/v7/entities/celfece8-34c4-6249-a1aa-
2e779294760e"}]}, "datePublished":"2024-02-10T19:23:20.000000Z","description":"Each year, fan
Gatorade will end up dousing the winning head coach at the end of the Super Bowl. See what t
are.", "image":{"thumbnail":{"contentUrl":"https://www.bing.com/th?
id=OVFT.aA6lMboTb200n0v8ES7V1S&pid=News","height":393,"width":700}}, "mentions":[{"name":"Sup
shower"}, {"name":"Big Game"}], "name":"What color will the Gatorade bath be at the end of 202
odds", "provider":[{"_type":"Organization","image":{"thumbnail":{"contentUrl":"https://www.bi
id=ODF.m1iod50DNiyyKu23kGIllQ&pid=news"}}, "name":"USA Today on MSN.com"}], "url":"https://www
us/sports/nfl/what-color-will-the-gatorade-bath-be-at-the-end-of-2024-super-bowl-see-the-lat
{"json_object":{"about":[{"name":"United Kingdom","readLink":"https://api.bing.microsoft.co
6bb2-4646-8f7c-3e6b3a53c831"}], "name":"Wales", "readLink":"https://api.bing.microsoft.com/api
d525-d360-f2eb5bf3410b"}, {"name":"BBC", "readLink":"https://api.bing.microsoft.com/api/v7/ent
78a579d79f5b"}]}, "category":"Sports", "datePublished":"2024-02-10T16:44:00.000000Z", "descript
news, BBC coverage, standings and statistics for Saturday's Six Nations match between Engla
Twickenham.", "image":{"thumbnail":{"contentUrl":"https://www.bing.com/th?
id=OVFT.qCQwjb_G0SzAlZjuKfuiPy&pid=News", "height":351, "width":624}}, "name":"Six Nations 2024
preview, team news, kick-off time & BBC coverage", "provider":[{"_type":"Organization", "image
{"contentUrl":"https://www.bing.com/th?
id=ODF.yhngt24TSWuyw3ur0Pt3WQ&pid=news"}}, {"name":"BBC"}], "url":"https://www.bbc.co.uk/sport/
{"json_object":{"about":[{"name":"NASDAQ", "readLink":"https://api.bing.microsoft.com/api/v7
3bb6-29450420b38e"}]}, "category":"ScienceAndTechnology", "datePublished":"2024-02-
10T12:42:00.000000Z", "description":"Celebrations may be in order for Vanda Pharmaceuticals
shareholders, with the covering analyst delivering a significant upgrade to their statutory
consensus statutory numbers for both revenue and earnings per share (EPS). " "image":{}}
```

```
1 print(json_list[0])
✓ <1 sec -Command executed in 304 ms by kumarsr on 11:28:36 PM, 2/11/24
PySpark (Python) ▾

{"json_object":{"about":[{"name":"New Delhi", "readLink":"https://api.bing.microsoft.com/api/v7/entities/b474d3c7-
a39a-d5ba-7426-18e0042f03e"}, {"name":"Russia", "readLink":"https://api.bing.microsoft.com/api/v7/entities/ed4fce79-
8ad4-352b-e4db36c49bbe"}, {"name":"Threat", "readLink":"https://api.bing.microsoft.com/api/v7/entities/9ccdc704-
0b71-57e0-a4f1-c3036ff79bd4"}], "datePublished":"2024-02-10T23:41:00.000000Z", "description":"The US is trying to
threaten the relationship between New Delhi and Moscow with sanctions, according to the Russian envoy. Read more
about the growing bilateral ties and the call for urgent reforms of the United Nations.", "image":{"thumbnail":
{"contentUrl":"https://www.bing.com/th?
id=OVFT.G2Ludo3yUmFT2bcx6r5Vo&pid=News", "height":379, "width":700}}, "mentions":[{"name":"New Delhi"}, {"name":"Russia"}, {"name":"Threat"}], "name":"US threatening sanctions to tear New Delhi away from Moscow: Russian
Envoy", "provider":[{"_type":"Organization", "image":{"thumbnail":{"contentUrl":"https://www.bing.com/th?
id=ODF.VBzhmoy4z9NL3FDronyM1&pid=news"}}, {"name":"Indiatimes on MSN.com"}], "url":"https://www.msn.com/en-
in/news/India/us-threatening-sanctions-to-tear-new-delhi-away-from-moscow-russian-envoy/ar-B811518t"})
```

## 13. Next we convert this Json string to Json dictionary to retrieve the data(like name,description etc) clearly and easily

### What are JSON loads () in Python?

The `json.loads()` method can be used to parse a valid JSON string and convert it into a [Python Dictionary](#). It is mainly used for deserializing native string, byte, or byte array which consists of JSON data into Python Dictionary.

#### 14. Lets use this on our last article and see the output



```
1 import json
2
3 news_json = json.loads(json_list[25])
```

PySpark

```
1 print(news_json)
✓ <1 sec -Command executed in 317 ms by kumarsr on 11:34:26 PM, 2/11/24
```

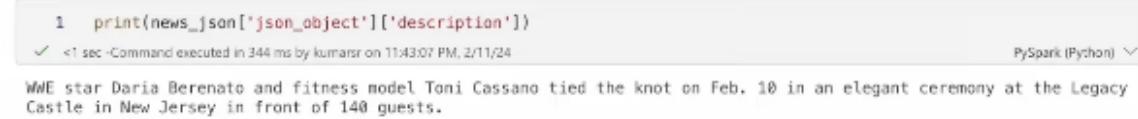
PySpark (Python) ↴

```
{'json_object': {'about': [{"name": "New Jersey", "readLink": "https://api.bing.microsoft.com/api/v7/entities/05277898-b62b-4878-8632-09d29756a2ff"}, {"name": "WWE", "readLink": "https://api.bing.microsoft.com/api/v7/entities/413f922e-0c5f-f9d2-0bd2-f5970294a12e"}], 'category': 'LifeStyle', 'datePublished': '2024-02-11T08:46:35.000000Z', 'description': 'WWE star Daria Berenato and fitness model Toni Cassano tied the knot on Feb. 10 in an elegant ceremony at the Legacy Castle in New Jersey in front of 140 guests.', 'image': {'thumbnail': {'contentUrl': 'https://www.bing.com/th?id=OVFT.Uf1PpdBMv2QSYmMztUz_BC&pid=News', 'height': 466, 'width': 700}}, 'mentions': [{"name": "New Jersey", "name": "Marriage"}, {"name": "WWE Star Daria Berenato Marries Toni Cassano in New Jersey Wedding Officiated by Maria Menounos (Exclusive)", "provider": {"_type": "Organization", "image": {"thumbnail": {"contentUrl": "https://www.bing.com/th?id=00F.kridahMliz5AdgcUGG5eB0&pid=news"}, "name": "People on MSN.com"}, "url": "https://www.msn.com/en-us/tv/celebrity/wwe-star-daria-berenato-marries-toni-cassano-in-new-jersey-wedding-officiated-by-maria-menounos-exclusive/ar-B81i5MS8'}}]}
```

MR K  
TALKS TECH

now we can use this json dict and easily process the data.

Like if we want to see the description of news article then we use

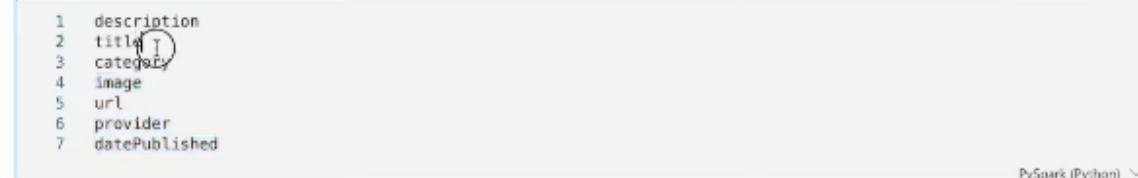


```
1 print(news_json["json_object"]['description'])
✓ <1 sec -Command executed in 344 ms by kumarsr on 11:43:07 PM, 2/11/24
```

PySpark (Python) ↴

```
WWE star Daria Berenato and fitness model Toni Cassano tied the knot on Feb. 10 in an elegant ceremony at the Legacy Castle in New Jersey in front of 140 guests.
```

#### 15. In our output we need this columns



```
1 description
2 title
3 category
4 image
5 url
6 provider
7 datePublished
```

PySpark (Python) ↴

and this would be our schema

## 16. To see our json data clearly ..we can use json parser

The screenshot shows a JSON parser interface. On the left, there is a large block of raw JSON code. On the right, the same JSON is displayed in a more readable, hierarchical tree format. The tree structure includes nodes for "about", "category", "datePublished", "description", "image", "mentions", and "contentUrl". The "mentions" node further branches into "name" nodes for "New Jersey", "WWER", and "Marriage". The "image" node contains a "thumbnail" URL.

```

{
  "json_object": {
    "about": [
      {
        "name": "New Jersey",
        "readLink": "https://api.bing.microsoft.com/api/v7/entities/05277898-b62b-4678-8632-09d29756a2ff",
        "name": "WWER",
        "readLink": "https://api.bing.microsoft.com/api/v7/entities/413f922e-0c5f-f9d2-0bd2-f5970294a12e"
      }
    ],
    "category": "LifeStyle",
    "datePublished": "2024-02-11T00:46:35.000000Z",
    "description": "WWER star Daria Berenato and fitness model Toni Cassano tied the knot on Feb. 10 in an elegant ceremony at the Legacy Castle in New Jersey in front of 140 guests.",
    "image": {
      "contentUrl": "https://www.bing.com/th?id=OVID.Uf1PpdBmV2QSYmMztUz_BC&pid=News",
      "height": 466,
      "width": 700
    },
    "mentions": [
      {
        "name": "New Jersey"
      },
      {
        "name": "WWER"
      },
      {
        "name": "Marriage"
      }
    ],
    "contentUrl": "https://www.bing.com/th?id=OVID.kridaNmIIz5AdgcUGG5eBQ&pid=news"
  }
}
  
```

## 17. We can use this code to get the required data from our JSON file

```

1 print(news_json["json_object"]["name"])
2 print(news_json["json_object"]["description"])
3 print(news_json["json_object"]["category"])
4 print(news_json["json_object"]["url"])
5 print(news_json["json_object"]["image"]["thumbnail"]["contentUrl"])
6 print(news_json["json_object"]["provider"][0]["name"])
7 print(news_json["json_object"]["datePublished"])
  
```

## 18. Now we need to write a code to extract all 26 articles

## 19. we can use a for loop to iterate all the 26 articles

The screenshot shows a Jupyter Notebook in a Databricks workspace. The sidebar on the left shows a "Lakehouses" folder containing a "bing\_lake\_db" database with tables and files. The main notebook area displays a Python script. The script initializes several empty lists: title, description, category, url, image, provider, and datePublished. It then processes each JSON object in a list named "json\_list" by loading the JSON string into a dictionary, extracting information from it, and appending the extracted values to their respective lists. Finally, it prints an error message if an exception occurs during processing.

```

title = []
description = []
category = []
url = []
image = []
provider = []
datePublished = []

# Process each JSON object in the list
for json_str in json_list:
    try:
        # Parse the JSON string into a dictionary
        article = json.loads(json_str)

        # Extract information from the dictionary
        title.append(article["json_object"]["name"])
        description.append(article["json_object"]["description"])
        category.append(article["json_object"]["category"])
        url.append(article["json_object"]["url"])
        image.append(article["json_object"]["image"]["thumbnail"]["contentUrl"])
        provider.append(article["json_object"]["provider"][0]["name"])
        datePublished.append(article["json_object"]["datePublished"])

    except Exception as e:
        print(f"Error processing JSON object: {e}")
  
```

here in this code ...json list has all the articles from our input file

20. Now if we run the above code we get this output

```
File "C:\Users\DELL\PycharmProjects\Python\news\main.py", line 10, in <module>
    for json_str in json_list:
          ^~~~~~
TypeError: 'NoneType' object is not iterable
```

```
Error processing JSON object: 'category'
Error processing JSON object: 'category'
Error processing JSON object: 'category'
Error processing JSON object: 'image'
Error processing JSON object: 'category'
Error processing JSON object: 'image'
Error processing JSON object: 'image'
Error processing JSON object: 'image'
```

21. We are getting this error bcz ...there are some null values in our json file...so we have used try except to find any errors ..and we got this error

22. So to solve this...we only process the news article which has all the values(no nulls)

```
# Process each JSON object in the list
for json_str in json_list:
    try:
        # Parse the JSON string into a dictionary
        article = json.loads(json_str)

        if article["json_object"].get("category") and article["json_object"].get("image", {}).get("thumbnail", {}):
            #Extract information from the dictionary
            title.append(article["json_object"]["name"])
            description.append(article["json_object"]["description"])
            category.append(article["json_object"]["category"])
            url.append(article["json_object"]["url"])
            image.append(article["json_object"]["image"]["thumbnail"]["contentUrl"])
            provider.append(article["json_object"]["provider"][0]['name'])
            datePublished.append(article["json_object"]["datePublished"])

    except Exception as e:
        print(f"Error processing JSON object: {e}")
```

here we used a if condition to check whether the all values are present...if yes then we extract the data from the JSON file...if false then we ignore it

23. Another way to solve this is by using a default value

## 24. Next we will combine all this data and create a df called df\_cleaned

```
1  from pyspark.sql.types import StructType, StructField, StringType
2
3
4  # Combine the lists
5  data = list(zip(title,description,category,url,image,provider,datePublished))
6      [
7  # Define schema
8  schema = StructType([
9      StructField("title", StringType(), True),
10     StructField("description", StringType(), True),
11     StructField("category", StringType(), True),
12     StructField("url", StringType(), True),
13     StructField("image", StringType(), True),
14     StructField("provider", StringType(), True),
15     StructField("datePublished", StringType(), True)
16   ])
17
18 # Create DataFrame
19 df_cleaned = spark.createDataFrame(data, schema=schema)
```

1 display(df\_cleaned)

✓ 1 sec Command executed in 889 ms by kumars on 12:49:44 AM, 2/12/24

PySpark (Python) ▾

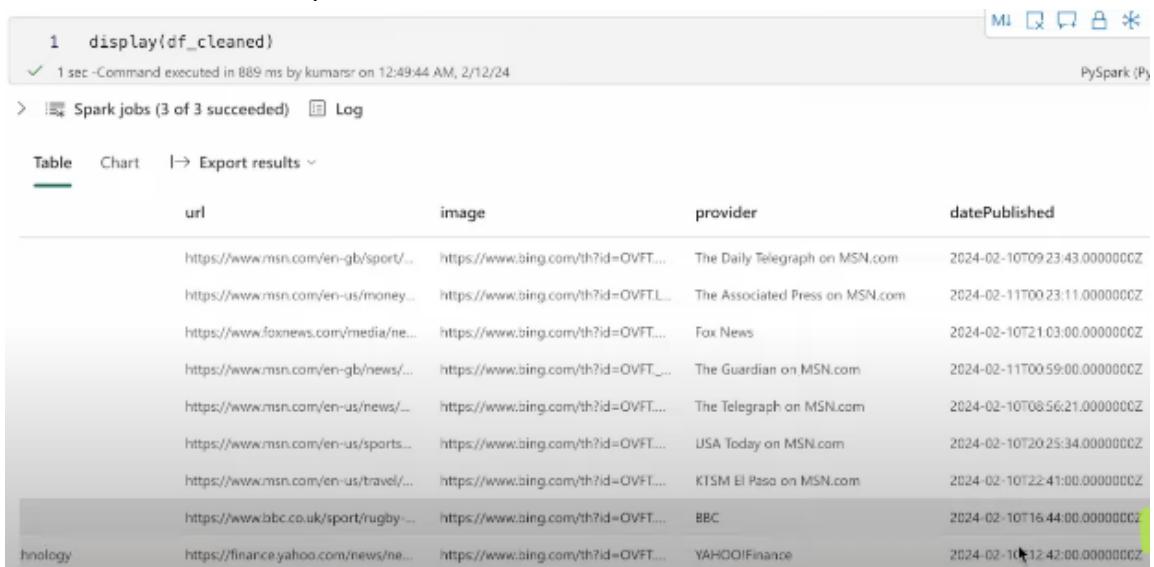
> ⚙ Spark jobs (3 of 3 succeeded) Log

Table Chart ↗ Export results

Index	title	description	category	url	image
1	Ireland v Italy, Six Nations 2024: Kick...	Ireland and Italy face each other on ...	Sports	<a href="https://www.msn.com/en-gb/sport/">https://www.msn.com/en-gb/sport/...</a>	<a href="https://www.msn.com/en-gb/sport/">https://www.msn.com/en-gb/sport/...</a>
2	New Mexico Budget Bill Would Fou...	New Mexico's strategy for spending...	Politics	<a href="https://www.msn.com/en-us/money/">https://www.msn.com/en-us/money...</a>	<a href="https://www.msn.com/en-us/money/">https://www.msn.com/en-us/money...</a>
3	New York Times editorial board de...	The New York Times editorial board ...	Politics	<a href="https://www.nytimes.com/media/me...">https://www.nytimes.com/media/me...</a>	<a href="https://www.nytimes.com/media/me...">https://www.nytimes.com/media/me...</a>
4	Australia news live: Coalition will op...	Follow the day's news live	LifeStyle	<a href="https://www.msn.com/en-gb/news/...">https://www.msn.com/en-gb/news/...</a>	<a href="https://www.msn.com/en-gb/news/...">https://www.msn.com/en-gb/news/...</a>
5	Scotland v France, Six Nations 2024:...	France head to Edinburgh on the ba...	Sports	<a href="https://www.msn.com/en-us/news/...">https://www.msn.com/en-us/news/...</a>	<a href="https://www.msn.com/en-us/news/...">https://www.msn.com/en-us/news/...</a>
6	What is the Super Bowl spread? Lat...	Super Bowl in just over 24 hours, be...	Sports	<a href="https://www.msn.com/en-us/sports/...">https://www.msn.com/en-us/sports/...</a>	<a href="https://www.msn.com/en-us/sports/...">https://www.msn.com/en-us/sports/...</a>
7	New vintage boutique opens in Do...	The El Paso Downtown Management...	LifeStyle	<a href="https://www.msn.com/en-us/travel/...">https://www.msn.com/en-us/travel/...</a>	<a href="https://www.msn.com/en-us/travel/...">https://www.msn.com/en-us/travel/...</a>
8	Six Nations 2024: England vs Wales ...	Match preview, team news, BBC cov...	Sports	<a href="https://www.bbc.co.uk/sport/rugby-...">https://www.bbc.co.uk/sport/rugby-...</a>	<a href="https://www.bbc.co.uk/sport/rugby-...">https://www.bbc.co.uk/sport/rugby-...</a>
9	News Flash: Analysts Just Made A Si...	Celebrations may be in order for Va...	ScienceAndTechnology	<a href="https://finance.yahoo.com/news/ne...">https://finance.yahoo.com/news/ne...</a>	<a href="https://finance.yahoo.com/news/ne...">https://finance.yahoo.com/news/ne...</a>
10	The latest on the Israel-Hamas war	Israeli Prime Minister Benjamin Net...	World	<a href="https://edition.cnn.com/middleeast/...">https://edition.cnn.com/middleeast/...</a>	<a href="https://edition.cnn.com/middleeast/...">https://edition.cnn.com/middleeast/...</a>

MR. K  
TALKS TECH

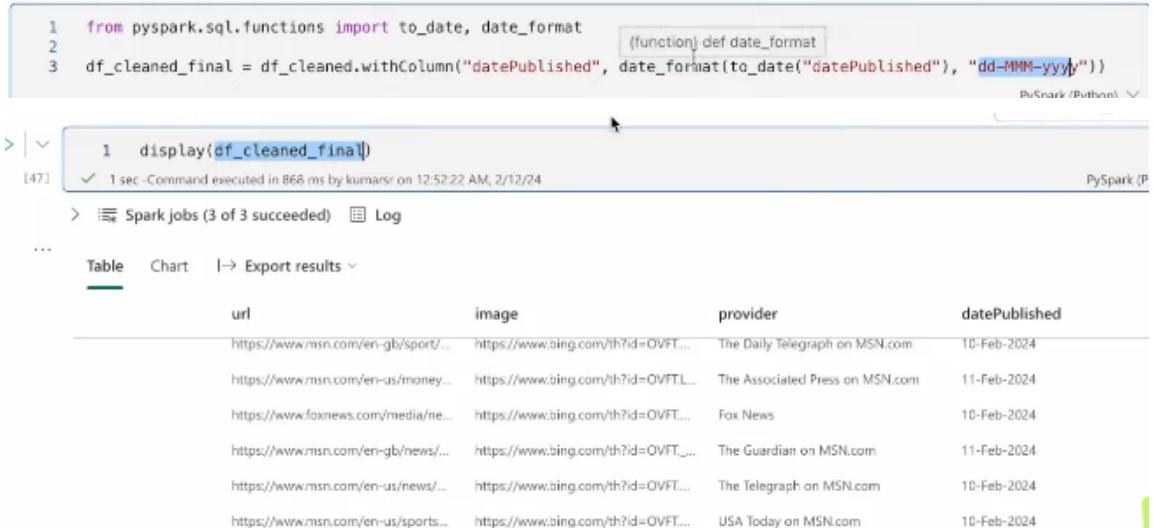
## 25. Next we format the datepublished col



```

1 display(df_cleaned)
✓ 1 sec -Command executed in 889 ms by kumarsr on 12:49:44 AM, 2/12/24
> Spark jobs (3 of 3 succeeded) Log
Table Chart I→ Export results
url image provider datePublished
https://www.msn.com/en-gb/sport/... https://www.bing.com/th?id=OVFT... The Daily Telegraph on MSN.com 2024-02-10T09:23:43.000000Z
https://www.msn.com/en-us/money... https://www.bing.com/th?id=OVFT... The Associated Press on MSN.com 2024-02-11T00:23:11.000000Z
https://www.foxnews.com/media/ne... https://www.bing.com/th?id=OVFT... Fox News 2024-02-10T21:03:00.000000Z
https://www.msn.com/en-gb/news/... https://www.bing.com/th?id=OVFT... The Guardian on MSN.com 2024-02-11T00:59:00.000000Z
https://www.msn.com/en-us/news/... https://www.bing.com/th?id=OVFT... The Telegraph on MSN.com 2024-02-10T08:56:21.000000Z
https://www.msn.com/en-us/sports... https://www.bing.com/th?id=OVFT... USA Today on MSN.com 2024-02-10T20:25:34.000000Z
https://www.msn.com/en-us/travel/... https://www.bing.com/th?id=OVFT... KTSM El Paso on MSN.com 2024-02-10T22:41:00.000000Z
https://www.bbc.co.uk/sport/rugby-... https://www.bing.com/th?id=OVFT... BBC 2024-02-10T16:44:00.000000Z
hnology https://finance.yahoo.com/news/ne... https://www.bing.com/th?id=OVFT... YAHOO!Finance 2024-02-10T12:42:00.000000Z

```

```

1 from pyspark.sql.functions import to_date, date_format
2
3 df_cleaned_final = df_cleaned.withColumn("datePublished", date_format(to_date("datePublished"), "dd-MMM-yyyy"))

```

```

1 display(df_cleaned_final)
[47] ✓ 1 sec -Command executed in 868 ms by kumarsr on 12:52:22 AM, 2/12/24
> Spark jobs (3 of 3 succeeded) Log
Table Chart I→ Export results
url image provider datePublished
https://www.msn.com/en-gb/sport/... https://www.bing.com/th?id=OVFT... The Daily Telegraph on MSN.com 10-Feb-2024
https://www.msn.com/en-us/money... https://www.bing.com/th?id=OVFT... The Associated Press on MSN.com 11-Feb-2024
https://www.foxnews.com/media/ne... https://www.bing.com/th?id=OVFT... Fox News 10-Feb-2024
https://www.msn.com/en-gb/news/... https://www.bing.com/th?id=OVFT... The Guardian on MSN.com 11-Feb-2024
https://www.msn.com/en-us/news/... https://www.bing.com/th?id=OVFT... The Telegraph on MSN.com 10-Feb-2024
https://www.msn.com/en-us/sports... https://www.bing.com/th?id=OVFT... USA Today on MSN.com 10-Feb-2024

```

## 26. Finally we save our cleaned data in delta format

Writing the Final Dataframe to the Lakehouse DB in a Delta format



```

1 df_cleaned_final.write.format("delta").saveAsTable("bing_lake_db.tbl_latest_news")

```

## 27. Now we can see our data in tables section

	abc_title	abc_description	abc_category	abc_url	abc_image	abc_provider	abc_datePublished
1	Ireland v Italy, Six Nations ...	Ireland and Italy face each...	Sports	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	The Daily Telegraph o...	10-Feb-2024
2	New Mexico Budget Bill W...	New Mexico's strategy for ...	Politics	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	The Associated Press ...	11-Feb-2024
3	New York Times editorial b...	The New York Times editori...	Politics	<a href="https://www.nytimes.com/...">https://www.nytimes.com/...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	Fox News	10-Feb-2024
4	Australia news live: Collin...	Follow the day's news live	LifeStyle	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	The Guardian on MS...	11-Feb-2024
5	Scotland v France, Six Nati...	France head to Edinburgh ...	Sports	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	The Telegraph on MS...	10-Feb-2024
6	What is the Super Bowl sp...	Super Bowl in just over 24 ...	Sport	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	USA Today on MSN.C...	10-Feb-2024
7	New vintage boutique ope...	The El Paso Downtown Ma...	LifeStyle	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	KTSM El Paso on MS...	10-Feb-2024
8	Six Nations 2024: England ...	Match preview, team news...	Sports	<a href="https://www.bbc.co.uk/spo...">https://www.bbc.co.uk/spo...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	BBC	10-Feb-2024
9	News Flash: Analysts Just...	Celebrations may be in or...	ScienceAndTechnology	<a href="https://finance.yahoo.com/...">https://finance.yahoo.com/...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	YAHOO!Finance	10-Feb-2024
10	The latest on the Israel-Ho...	Israel Prime Minister Beny...	World	<a href="https://edition.cnn.com/m...">https://edition.cnn.com/m...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	CNN	10-Feb-2024
11	King Charles cancer – latest...	Prince William has returne...	Entertainment	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	The Independent on ...	10-Feb-2024
12	It's a brand new route on L...	In travel news this week, a...	LifeStyle	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	CNN on MSN.com	10-Feb-2024
13	European soccer news: Ma...	Real Madrid send a statem...	Sports	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	ESPN on MSN.com	11-Feb-2024
14	The best new popular ficti...	Lena's gift forces her dad, ...	Entertainment	<a href="https://www.thetimes.co.u...">https://www.thetimes.co.u...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	The Times	10-Feb-2024
15	WWE Star Daria Berenato ...	WWE star Daria Berenato ...	LifeStyle	<a href="https://www.msn.com/en-...">https://www.msn.com/en-...</a>	<a href="https://www.bing.com/thi...">https://www.bing.com/thi...</a>	People on MSN.com	11-Feb-2024

## Incremental Load - Type1

- Previously we have saved our table in our one lake...and if we again try to save it..we get an error 'already exists'

```

1 df_cleaned_final.write.format("delta").saveAsTable('bing_lake_db.tbl_latest_news')
! 1 sec -Command executed in 1 sec 567 ms by kumarsr on 12:53:54 AM, 2/24/24
> Diagnostics ①

AnalysisException                                     Traceback (most recent call last)
Cell In[299], line 1
----> 1 df_cleaned_final.write.format("delta").saveAsTable('bing_lake_db.tbl_latest_news')

File /opt/spark/python/lib/pyspark.zip/pyspark/sql/readwriter.py:1521, in DataFrameWriter.saveAsTable(self, name, mode, partitionBy, **options)
  1519 if format is not None:
  1520     self.format(format)
-> 1521 self._jwrite.saveAsTable(name)

File ~/cluster-env/trident_env/lib/python3.10/site-packages/py4j/java_gateway.py:1322, in JavaMember.__call__(self, *args)
  1316 command = proto.CALL_COMMAND_NAME +\
  1317     self.command_header +\
  1318     args_command +\
  1319     proto.END_COMMAND_PART
-> 1321 answer = self.gateway_client.send_command(command)
-> 1322 return_value = answer.get_return_value()

```

- So to overcome this we can use mode

```

1 df_cleaned_final.write.format("delta").mode('overwrite').saveAsTable('bing_lake_db.tbl_latest_news')
✓ 5 sec -Command executed in 4 sec 806 ms by kumarsr on 12:54:55 AM, 2/24/24
> Spark jobs (7 of 7 succeeded) Log

```

ANd this is not the best way to do it...if we overwrite the same data ..it costs us additional time and compute power...and the other reason is we can lose the data

### 3. Next way to solve is by using append mode

```
1 df_cleaned_final.write.format("delta").mode('append').saveAsTable('bing_lake_db.tbl_latest_news')
✓ 5 sec -Command executed in 4 sec 866 ms by kumarsr on 12:54:55 AM, 2/24/24
> 7 of 7 succeeded  Log
```

PySpark (Python) ▾

if we use this mode..the same table data will be appended to the old table...and this also is not a best way to deal

### 4. Third way is to use incremental load.and this will be best way

Here if the data already exists in table..then we don't add the new data...

This can be done with the help of SQL Merge Data Warehousing

In the context of data warehousing, the SQL `MERGE` statement is a powerful tool that allows you to synchronize data between two tables (typically a target table and a source table). It combines `INSERT`, `UPDATE`, and `DELETE` operations in a single query. This operation is often used for tasks like data consolidation, updating dimensions, and managing slowly changing dimensions (SCDs).

#### What does the `MERGE` statement do?

- **Insert:** Adds new records to the target table if they don't already exist.
- **Update:** Modifies existing records in the target table if they have changed.
- **Delete:** Removes records from the target table if they no longer exist in the source.

Let's consider a data warehouse where we have a `sales_fact` table (target) and a `new_sales` table (source) with the same structure. We want to update the `sales_fact` table with new sales data from `new_sales`, add any new records, and delete records that no longer exist in the source.

#### Tables:

- `sales_fact` :
  - `sale_id`
  - `product_id`
  - `sale_date`
  - `quantity_sold`
  - `total_amount`

- `new_sales`:
  - `sale_id`
  - `product_id`
  - `sale_date`
  - `quantity_sold`
  - `total_amount`

**Scenario:**

1. If a sale exists in both `sales_fact` and `new_sales` with the same `sale_id`, update the record in `sales_fact`.
2. If a sale exists in `new_sales` but not in `sales_fact`, insert it into `sales_fact`.
3. If a sale exists in `sales_fact` but not in `new_sales`, delete it from `sales_fact`.

SQL | MERGE | Query:

```
sql
Copy

MERGE INTO sales_fact AS SF
USING new_sales AS NS
ON SF.sale_id = NS.sale_id
WHEN MATCHED THEN
    UPDATE SET SF.quantity_sold = NS.quantity_sold, SF.total_amount = NS.total_amount
WHEN NOT MATCHED THEN
    INSERT (sale_id, product_id, sale_date, quantity_sold, total_amount)
    VALUES (NS.sale_id, NS.product_id, NS.sale_date, NS.quantity_sold, NS.total_amount)
WHEN NOT MATCHED BY SOURCE THEN
    DELETE;
```

**Explanation:**

- **Matching rows** (when `sale_id` is the same in both `sales_fact` and `new_sales`) will be updated with the latest `quantity_sold` and `total_amount`.
- **Non-matching rows** in `new_sales` will be inserted into `sales_fact` to reflect new sales.
- **Rows in** `sales_fact` **that do not exist in** `new_sales` **will be deleted** (which may represent outdated data, like canceled sales).

5. We have two types

# TYPE 1 VS TYPE 2

6. IN type 1...if we have new unique primary key ..then it will add a new row

## TYPE 1

Record 1: 333, Alex, alex@gmail.com

Record 2: 222, Mike, mike123@gmail.com

Record 3: 111, John, john@gmail.com

ID	NAME	Email_ID
111	John	john@gmail.com
222	Mike	mike123@gmail.com
333	Alex	alex@gmail.com

If the key is already present and if there any changes it will update it...

If it will ignore the duplicates

## TYPE 1

Record 1: 333, Alex, alex@gmail.com

Record 2: 222, Mike, mike123@gmail.com

Record 3: 111, John, john@gmail.com

- 1. Overwritten
- 2. No History

ID	NAME	Email_ID
111	John	john@gmail.com
222	Mike	mike123@gmail.com
333	Alex	alex@gmail.com

7. It is similar to SCD types

The diagram illustrates Type 2 Slowly Changing Dimension (SCD) with three records and a history table.

**TYPE 2**

Record 1: 333, Alex, alex@gmail.com  
 Record 2: 222, Mike, mike123@gmail.com  
 Record 3: 111, John, john@gmail.com

**1. New Row**  
**2. History**

ID	NAME	Email_ID	FLAG
111	John	john@gmail.com	Y
222	Mike	mike@gmail.com	N
333	Alex	alex@gmail.com	Y
222	Mike	mike123@gmail.com	Y

MR. K TALKS TECH

8. We will use type 1 in our proj

```

1  from pyspark.sql.utils import AnalysisException
2
3  try:
4
5      table_name = 'bing_lake_db.tbl_latest_news'
6
7      df_cleaned_final.write.format("delta").saveAsTable(table_name)
8
9  except AnalysisException:
10
11     print("Table Already Exists")
12
13     df_cleaned_final.createOrReplaceTempView("vw_df_cleaned_final")
14
15     spark.sql(f"""
16         MERGE INTO {table_name} target_table
17             USING vw_df_cleaned_final source_view
18
19                 ON source_view.url = target_table.url
20
21                 WHEN MATCHED AND
22                     source_view.title <> target_table.title OR
23                     source_view.description <> target_table.description OR
24                     source_view.category <> target_table.category OR
25                     source_view.image <> target_table.image OR
26                     source_view.provider <> target_table.provider OR
27                     source_view.datePublished <> target_table.datePublished
28
29                 THEN UPDATE SET *
30
31         """)
```

9. Here in our code we use URL as join col as each article has unique URL...if when they are matched.. then we check if any cols has updated or not....if not updated then we leave as it is...cuz this is a duplicate...if there any changes ...then we update the entire row again

10. Similarly if there are no matching URL then it is a new item and we insert entire row

```
except AnalysisException:

    print("Table Already Exists")

    df_cleaned_final.createOrReplaceTempView("vw_df_cleaned_final")

    spark.sql(f"""
        MERGE INTO {table_name} target_table
        USING vw_df_cleaned_final source_view
        ON source_view.url = target_table.url

        WHEN MATCHED AND
        source_view.title <> target_table.title OR
        source_view.description <> target_table.description OR
        source_view.category <> target_table.category OR
        source_view.image <> target_table.image OR
        source_view.provider <> target_table.provider OR
        source_view.datePublished <> target_table.datePublished

        THEN UPDATE SET *

        WHEN NOT MATCHED THEN INSERT *
    """)

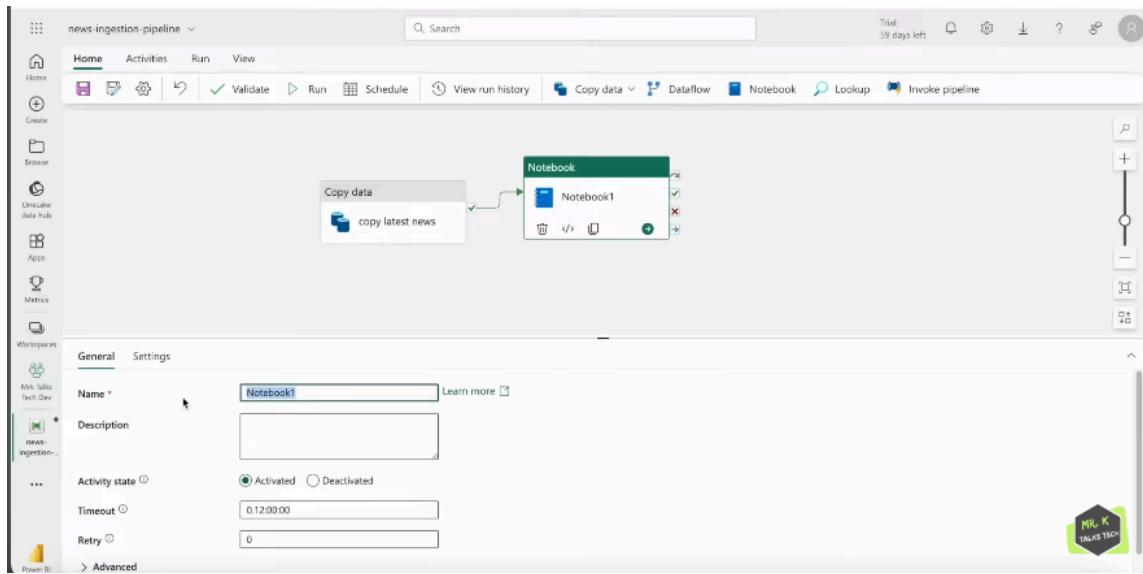
    ....)
```

## Orchestrating the entire project

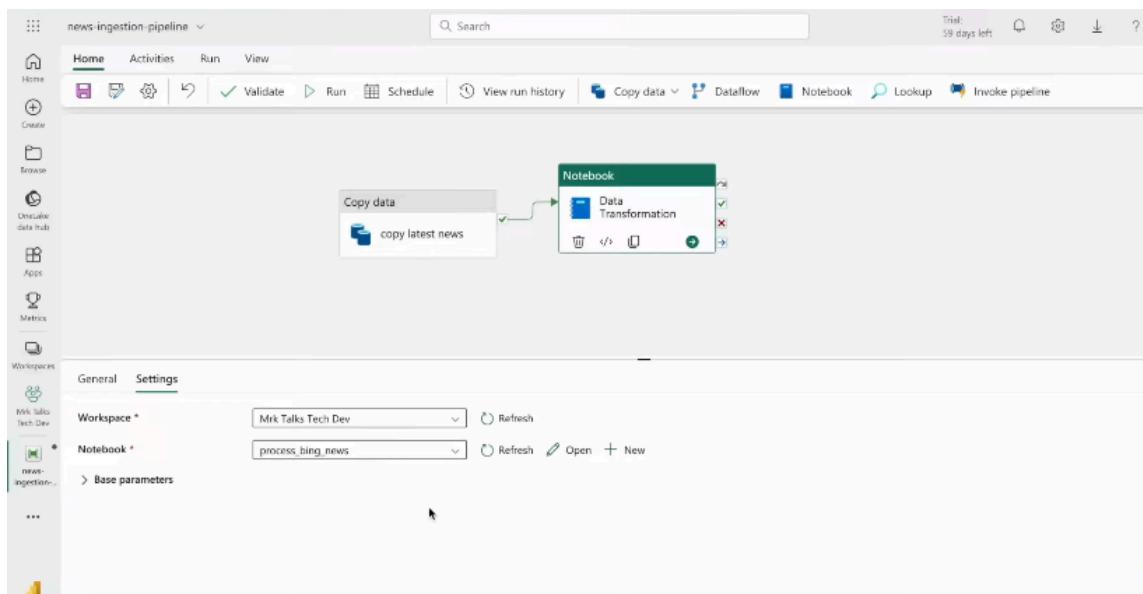
1. Now we will build this entire end to end pipeline

2. First we go to our ingestion pipeline..where we ingest data from Bing API

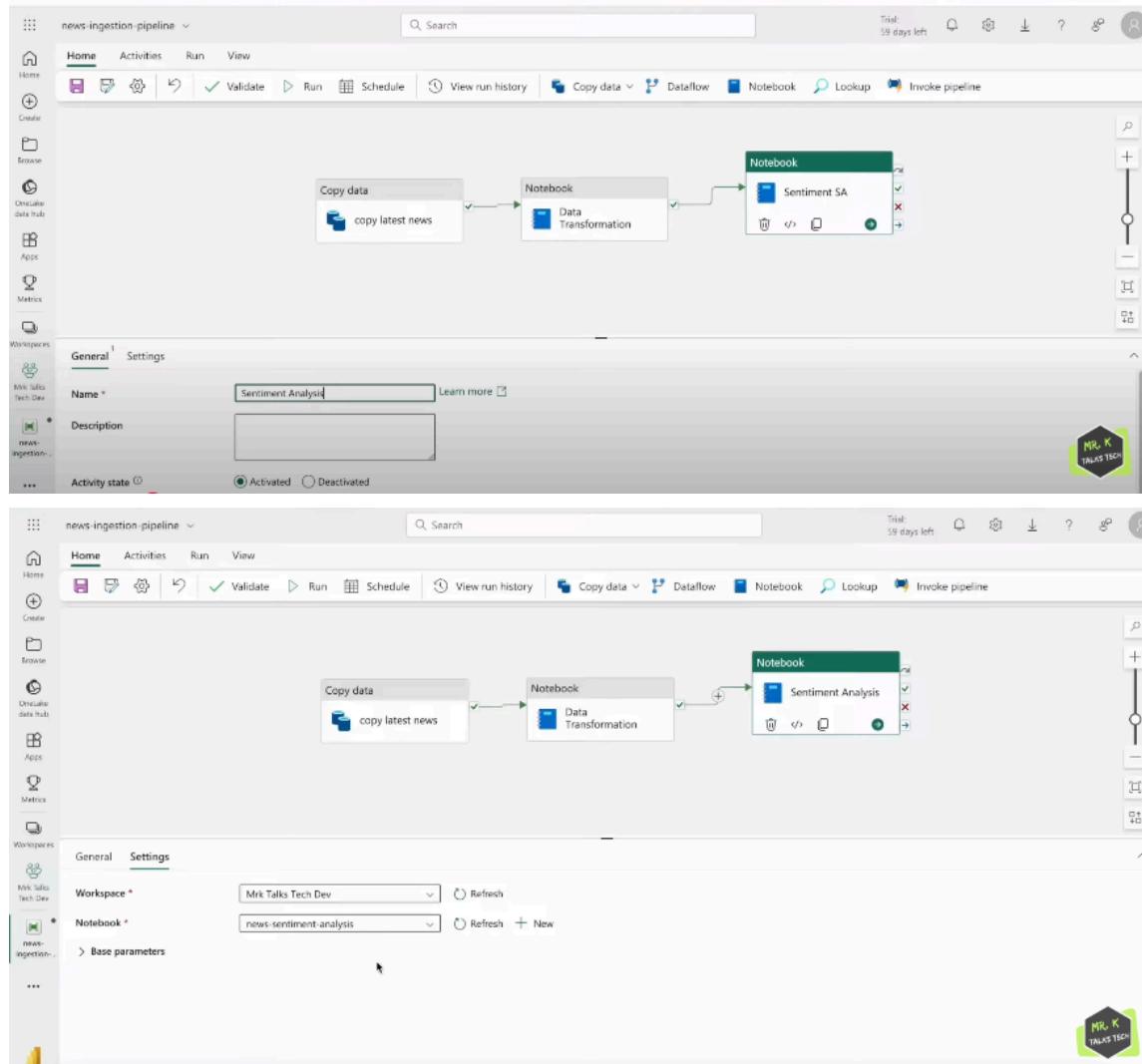
### 3. Now we will add transformation notebook to this activity



When the copy data is successful then it goes to notebook activity and then we provide the notebook that we have created

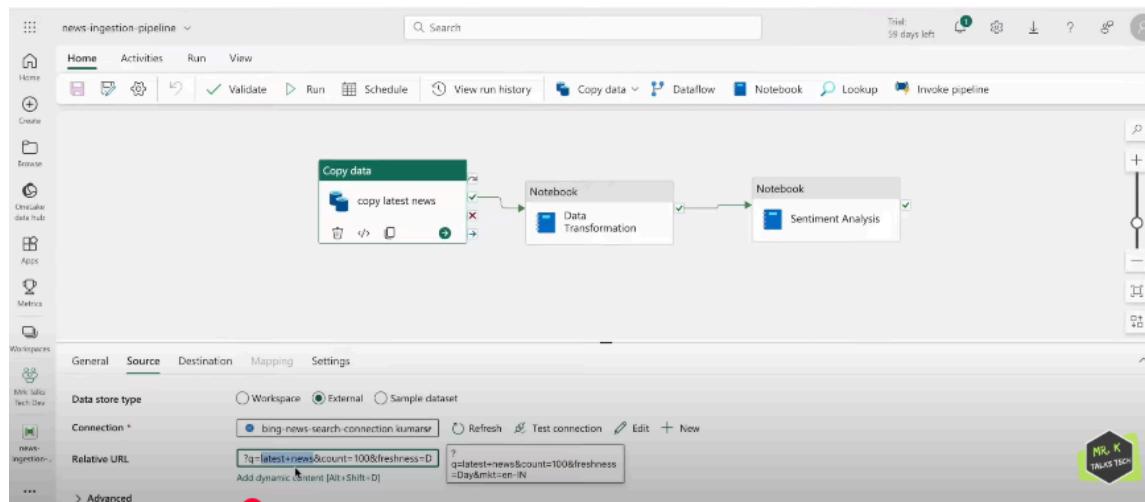


#### 4. Next we will add a sentiment analysis notebook activity



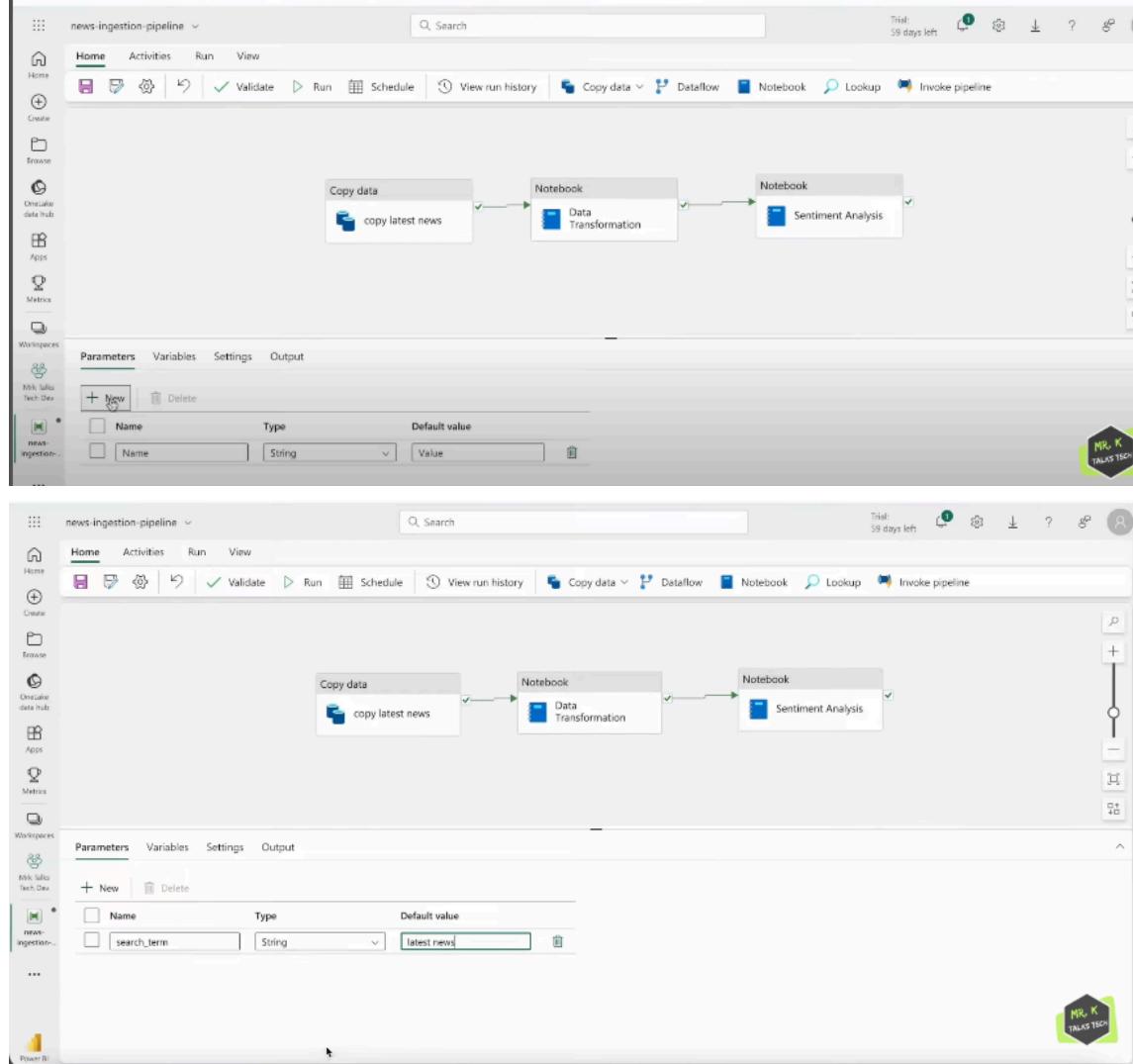
#### 5. This is our entire pipeline

#### 6. Now in the copy activity we have manually encoded the latest\_news keyword for bingAPI



ANd we'd like it to be dynamic..by giving a parameter and our freshness argument will gives us the latest news

For that ...click on white space and click parameters



Next we use this parameter in our relative URL...

Now go to Add dynamic content and select the param we created...then we'll get the expression..which we can use in url

The screenshot shows the 'Pipeline expression builder' interface. In the main area, the expression `@pipeline().parameters.search_term` is entered. Below the input field, there are tabs for 'Parameters', 'System variables', 'Functions', and 'Variables'. A search bar and a '+' button are also present. A watermark for 'MR. K TALKS TECH' is visible in the bottom right corner.

Relative URL :

Now if we preview the data...we can type what we need as value

The screenshot shows the 'Copy data' activity in the pipeline preview screen. The 'Source' tab is selected. In the 'Relative URL' field, the expression `?q=@{pipeline().parameters.search_term}` is entered. A tooltip or validation message 'Please provide actual value of the parameters to preview data' is displayed above the URL field. To the right, a table titled 'Please provide actual value of the parameters for pipeline news-ingestion-pipeline.' shows a row for 'search\_term' with a value 'latest news'. A watermark for 'MR. K TALKS TECH' is visible in the bottom right corner.

7. Now we will schedule this pipeline to run every mrgn at 6am

