Getting data from the data warehouse

# 1. Getting data from the data warehouse#

In this part of the workflow, you will learn how to get data by making HTTP requests with the HTTP Request node.

After completing this section, your workflow will look like this:

First, let's set the scene for building Nathan's workflow.

## Create new workflow#

Open your Editor UI and create a new workflow with one of the two possible commands:

• Select Ctrl+Alt+N or Cmd+Option+N on your keyboard.

• Open the left menu, navigate to Workflows, and select Add workflow.

Name this new workflow "Nathan's workflow."

The first thing you need to do is get data from ABCorp's old data warehouse.

In a previous chapter, you used an action node designed for a specific service (Hacker News). But not all apps or services have dedicated nodes, like the legacy data warehouse from Nathan's company.

Though we can't directly export the data, Nathan told us that the data warehouse has a couple of API endpoints. That's all we need to access the data using the HTTP Request node in n8n.

No node for that service?

The HTTP Request node is one of the most versatile nodes, allowing you to make HTTP requests to query data from apps and services. You can use it to access data from apps or services that don't have a dedicated node in n8n.

## Add an HTTP Request node#

Now, in your Editor UI, add an HTTP Request node like you learned in the lesson Adding nodes. The node window will open, where you need to configure some parameters.

This node will use credentials.

Credentials

Credentials are unique pieces of information that identify a user or a service and allow them to access apps or services (in our case, represented as n8n nodes). A common form of credentials is a username and a password, but they can take other forms depending on the service.

In this case, you'll need the credentials for the ABCorp data warehouse API included in the email from n8n you received when you signed up for this course. If you haven't signed up yet, sign up here.

In the Parameters of the HTTP Request node, make the following adjustments:

• Method: This should default to GET. Make sure it's set to GET.

• URL: Add the Dataset URL you received in the email when you signed up for this course.

• Send Headers: Toggle this control to true. In Specify Headers, ensure Using Fields Below is selected.  
Header Parameters > Name: Enter unique\_id.  
Header Parameters > Value: The Unique ID you received in the email when you signed up for this course.

• Header Parameters > Name: Enter unique\_id.

unique\_id

• Header Parameters > Value: The Unique ID you received in the email when you signed up for this course.

• Authentication: Select Generic Credential Type. This option requires credentials before allowing you to access the data.  
Generic Auth Type: Select Header Auth. (This field will appear after you select the Generic Credential Type for the Authentication.)  
Credential for Header Auth: To add your credentials, select + Create new credential. This will open the Credentials window.  
In the Credentials window, set Name to be the Header Auth name you received in the email when you signed up for this course.  
In the Credentials window, set Value to be the Header Auth value you received in the email when you signed up for this course.  
Select the Save button in the Credentials window to save your credentials. Your Credentials Connection window should look like this:  
HTTP Request node credentials

• Generic Auth Type: Select Header Auth. (This field will appear after you select the Generic Credential Type for the Authentication.)

• Credential for Header Auth: To add your credentials, select + Create new credential. This will open the Credentials window.

• In the Credentials window, set Name to be the Header Auth name you received in the email when you signed up for this course.

• In the Credentials window, set Value to be the Header Auth value you received in the email when you signed up for this course.

• Select the Save button in the Credentials window to save your credentials. Your Credentials Connection window should look like this:  
HTTP Request node credentials

Credentials naming

New credential names follow the " account" format by default. You can rename the credentials by clicking on the name, similarly to renaming nodes. It's good practice to give them names that identify the app/service, type, and purpose of the credential. A naming convention makes it easier to keep track of and identify your credentials.

Once you save, exit out of the Credentials window to return to the HTTP Request node.

## Get the data#

Select the Test step button in the HTTP Request node window. The table view of the HTTP request results should look like this:

This view should be familiar to you from the Building a mini-workflow page.

This is the data from ABCorp's data warehouse that Nathan needs to work with. This data set includes sales information from 30 customers with five columns:

• orderID: The unique id of each order.

orderID

• customerID: The unique id of each customer.

customerID

• employeeName: The name of Nathan's colleague responsible for the customer.

employeeName

• orderPrice: The total price of the customer's order.

orderPrice

• orderStatus: Whether the customer's order status is booked or still in processing.

orderStatus

booked

processing

## What's next?#

Nathan 🙋: This is great! You already automated an important part of my job with only one node. Now instead of manually accessing the data every time I need it, I can use the HTTP Request Node to automatically get the information.

You 👩‍🔧: Exactly! In the next step, I'll help you one step further and insert the data you retrieved into Airtable.