

## Create connections with Environment Variables

# Create connections with Environment Variables

## Introduction

There are three ways to define a Connection:

1. In the Airflow metadata database (using the CLI or the UI)
2. In an external Secret Backend (cf: [secret backend](#) module)
3. In an Environment Variables

## Why?

When you create a Connection in the database, each time a task needs this Connection, it requests the database. If you have many tasks, that can drastically increase the workload on your database.

With Connections in Environment Variables, the task doesn't need to request the database. Airflow checks if the corresponding Connection exists and grabs it without accessing the database. Again, at scale, this can help reduce the number of requests on your database.

On top of that, Connections defined in Environment Variables do not show up in the Airflow UI or using `airflow connection list`.

## How?

To define a Connection as an environment variable, you need to use the following naming convention:

```
AIRFLOW__CONN_{CONN_ID}='Your connection'
```

Here `{CONN_ID}` corresponds to the connection id you want to give to your connection such as `snowflake_default`, `postgres_default`, etc.

## URI format

In your terminal, if serializing with Airflow URI, you can export a connection by executing:

```
export AIRFLOW_CONN_MY_PROD_DATABASE='my-conn-type://login:password@host:port/schema?param1=val1&param2=val2'
```

That creates the connection `MY_PROD_DATABASE` with the connection string given.

## JSON format

In your terminal, if serializing with JSON, you can export a connection by executing

```
export AIRFLOW_CONN_MY_PROD_DATABASE='{  
  "conn_type": "my-conn-type",  
  "login": "my-login",  
  "password": "my-password",  
  "host": "my-host",  
  "port": 1234,  
  "schema": "my-schema",  
  "extra": {  
    "param1": "val1",  
    "param2": "val2"  
  }  
}'
```

## Notes

Creating Connections using environment variables reduces requests made to the database and adds a security layer since they do not appear on the UI and CLI. However, it supposes restraint access to your bash environment (and so environment variables) and a mechanism so they get exported in every machine where Airflow is running (Airflow workers).

Was this page helpful?



Want feedback like this? [Try Hotjar](#)